Finite-Time Trajectory Tracking Control of a Class of Nonlinear Discrete-Time Systems

Zhuo Wang, Member, IEEE, Renquan Lu, Member, IEEE, Furong Gao, and Hong Wang, Senior Member, IEEE

Abstract—This paper studies how to control a class of nonlinear discrete-time systems, to completely track any given bounded expected trajectories in finite time. For this problem, we develop some constructive control methods for both the total output case as well as the partial output case. Some of these control methods can design the tracking instant when the trajectory tracking is just accomplished, but cannot guarantee the monotonic decrease of the norm of the tracking error before that instant. The other control methods not only can determine the tracking instant after which the (partial) output trajectory coincides with the expected trajectory, but also can make the norm of the tracking error decrease monotonically before that instant. In the mean time, for the partial output case, the rest part of the system output is bounded for all the time. Then, we give some simulation examples to demonstrate the effectiveness of these control methods. Among these examples, a permanent magnet linear motor system is employed to illustrate the practicability.

Index Terms—Bounded expected trajectories, finite-time trajectory tracking, nonlinear discrete-time systems, partial output trajectory, total output trajectory.

I. INTRODUCTION

In control theory and control engineering, trajectory tracking is one of the most important subjects to be extensively studied for both linear/nonlinear continuous-time systems and discrete-time systems [1], [2], [3]. The objective of these studies is to make the system state/output track a prescribed trajectory. The trajectory tracking has so far been investigated can be divided into two categories: the asymptotic trajectory tracking [4], [5], [6] and the finite-time trajectory tracking [7], [8], [9]. The asymptotic trajectory tracking is generally in the Lyapunov’s sense [5], [6], [10], [11], that the norm of the tracking error decreases monotonically as time increases, and the system state/output will converge to the expected trajectory as time tends to infinity. Besides, it is often required that the system model has smooth dynamic characteristics [12], [13], such as the continuous partial derivatives. However, it is impractical to wait for an infinitely long time, thus people need to introduce a prescribed accuracy (usually referred to as the maximum tolerable tracking error), and consider the trajectory tracking is accomplished when the accuracy requirement is satisfied.

The finite-time trajectory tracking, on the other hand, is quite a different problem. It is concerned with how to design a tracking instant when the trajectory tracking is accomplished, and how to make the tracking error keep zero from that instant on. In this sense, the finite-time trajectory tracking is superior to the asymptotic trajectory tracking regardless of the system requirements, since it has higher tracking precision, faster convergence rate, and can determine the exact time when the trajectory tracking is accomplished. In engineering practice, there are some precise tracking tasks need to be done in finite time, due to the resource limitation or the standard of workmanship, etc. For example, in the areas of shipbuilding and car manufacturing, it has become very common that the industrial robot precisely cuts materials along the preset edge or welds workpieces along the preset welding seam [14], [15]. When launching aircraft, the aircraft is supposed to enter the predesigned trajectory in finite time, and subsequently, flies along this trajectory precisely [16], [17]. In astronomical observation area, the attitude precision and the flying precision along the predesigned trajectory of spacecrafts (such as Hubble Space Telescope), will directly affect the observation of the far-away stars [18], [19]. In this circumstance, the study and investigation of finite-time trajectory tracking problems have important significance in theory and practice.

Up to now, most of the existing works on finite-time trajectory tracking are about the control design problem, and some typical control schemes have been proposed for it. Such as the backstepping methods [10], [19], [20], the sliding mode methods [18], [21], [22], and the neural network methods [23], [24]. Despite the merits of these control methods, they have some shortcomings: they can only deal with some particular systems with specific structures or dynamic characteristics, but cannot control a general class of systems; they can only drive the system state/output to track some particular trajectories, but not to track arbitrary bounded trajectories; they can only accomplish the trajectory tracking within a finite time horizon, but cannot determine the exact tracking instant when the state/output trajectory begins to coincide with the expected trajectory. In this situation, further researches need to be conducted. The study on these problems is necessary to reveal the essential role of finite-time trajectory tracking control.

In this paper, we will conduct research on the finite-time output trajectory tracking control problem for a class of nonlinear discrete-time systems. We will develop control laws in a constructive way, for both the total output case and the partial output case of these systems. Our control methods...
have some advantages: first, they are not limited to particular systems, but can control a general class of nonlinear discrete-time systems; second, they do not require special structure or smooth dynamic characteristics (like continuous partial derivatives) of the system model; third, they are applicable to any bounded expected trajectories; and finally, they can arbitrarily design the tracking instant when the trajectory tracking is accomplished.

The remainder of this paper is organized as follows. Section II studies how to control the system output to completely track the given bounded expected trajectory in finite time, and how to make the norm of the tracking error decrease monotonically before the trajectory tracking is accomplished. Section III studies how to control a part of the system output to completely track the given bounded expected trajectory in finite time while keeping the other part bounded, and also studies the conditions of monotonic decrease of the norm of the tracking error. Section IV presents some simulation examples to demonstrate the effectiveness of these control methods. In particular, a permanent magnet linear motor system is employed to illustrate the practicability in engineering applications. Finally, Section V draws the conclusion of this paper and looks forward to the prospect of future researches.

II. Finite-Time Total Output Trajectory Tracking Control

Consider the following nonlinear discrete-time system:

\[ y(k + 1) = f(y(k)) + B(y(k))u(k), \tag{1} \]

where \( u(k), y(k) \in \mathbb{R}^n \) are the input and the output, respectively. Integer \( k \geq 0 \) is the tracking instant. \( f: \mathbb{R}^n \rightarrow \mathbb{R}^n \) is piecewise continuous with respect to \( y(k) \), and \( f(0) = 0 \). \( B(y(k)) \in \mathbb{R}^{n \times n} \) is the continuous input gain matrix.

In this section, we study how to control the total output of system (1) to completely track any given bounded trajectory in finite time. Define the set of the trajectories to be tracked:

\[ S_1 = \{ y(k) \mid k \geq 0, \; y(k) \in \mathbb{R}^n, \; \| y(k) \| \leq \alpha < \infty \}. \tag{2} \]

In this paper, \( \| \cdot \| \) denotes the spectral norm. \( g(t) \) is continuous with respect to \( t \in \mathbb{R}^+ \) and \( g(k) \) is the discrete value of \( g(t) \) at \( t = kT \), where \( T > 0 \) is the sampling period. In this way, \( 0 < \alpha < \infty \) is the boundary of any \( y(k) \in S_1 \). Moreover, we suppose that each given \( g(k) \in S_1 \) is known for all \( k \geq 0 \).

**Assumption 1:** For system (1), assume that \( \forall y(k) \in \mathbb{R}^n \),

\[ \text{Rank} \left[ B(y(k)) \right] = n. \]

When Assumption 1 is satisfied, system (1) is controllable. To facilitate discussion, we suppose that \( \| y(0) \| \leq \alpha \) for any given \( y(0) \); otherwise, we may first control the output to satisfy this initial condition, and then discuss the tracking problem.

**Theorem 1:** Suppose that system (1) satisfies Assumption 1. Then, for any given \( y(0) \) and \( g(k) \in S_1 \), the output \( y(k) \) of system (1) can completely track \( g(k) \) in finite time.

**Proof:** If \( y(0) = g(0) \), we design the control input as

\[ u(k) = B^{-1}(y(k)) \left[ g(k + 1) - f(y(k)) \right] \quad (k \geq 0). \tag{3} \]

From (1) and (3), we can see that \( y(k) = g(k) \) \( (k \geq 0) \). Thus, the system output \( y(k) \) completely tracks \( g(k) \) instantaneously by using control law (3).

If \( y(0) \neq g(0) \), preset a tracking instant \( M \geq 1 \) when the trajectory tracking is just accomplished. Then, we design the control input as

\[ u(k) = \begin{cases} B^{-1}(y(k)) \left[ g(k + 1) - f(y(k)) \right], & (0 \leq k \leq M - 1) \smallbreak 0, & (k \geq M). \end{cases} \tag{4} \]

From (1) and (4), we can obtain

\[ y(k + 1) = \begin{cases} B^{-1}(y(k)) \left[ g(k + 1) - f(y(k)) \right], & (0 \leq k \leq M - 1) \smallbreak g(k + 1), & (k \geq M). \end{cases} \tag{5} \]

When \( k = M - 1 \), \( y(k + 1) = y(M) = \frac{k + 1}{M} g(k + 1) = g(M) \). When \( k \geq M \), \( y(k + 1) = g(k + 1) \). Therefore, the system output \( y(k) \) completely tracks \( g(k) \) from \( k = M \) on by using control law (4).

In summary, the output \( y(k) \) of system (1) can completely track \( g(k) \) in finite time.

From the above proof, we can see that when \( y(0) = g(0) \),

\[ \| y(k) \| = \| g(k) \| \leq \alpha \quad \text{for all} \quad k \geq 0. \quad \text{When} \quad y(0) \neq g(0), \]

\[ \| y(k) \| = \begin{cases} \frac{k}{M} \| g(k) \|, & (1 \leq k \leq M) \smallbreak \| g(k) \|, & (k \geq M + 1). \end{cases} \]

Since we have supposed that \( \| y(0) \| \leq \alpha \) for any given \( y(0) \), then \( \| y(k) \| \leq \alpha \quad \text{for all} \quad k \geq 0 \). In summary, the system output also has a boundary of \( 0 < \alpha < \infty \).

In the above discussion, though not specifically mentioned, system (1) is usually supposed to be globally asymptotically stable. Otherwise, according to Assumption 1, system (1) is stabilizable, we can stabilize it first and then study the tracking problem. Note that whether control inputs (3) and (4) are stabilizable, we can stabilize it first and then study the tracking problem. Note that whether control inputs (3) and (4) are stabilizable, we can stabilize it first and then study the tracking problem.

**Suppose that** \( \sup_{y(k) \in \mathbb{R}^n} \{ \| B^{-1}(y(k)) \| \} = M_b < \infty \). Then if \( y(0) = g(0) \), from (2) and (3), the control input should satisfy

\[ \| u(k) \| \leq \| B^{-1}(y(k)) \| \| g(k + 1) \| + \| f(y(k)) \| \leq M_b [\alpha + \| f(y(k)) \|]. \]

When system (1) is globally asymptotically stable, it is easy to see that \( \| f(y(k)) \| < \| y(k) \| \). With the above conclusion that \( \| y(k) \| = \| g(k) \| \leq \alpha \quad \text{for all} \quad k \geq 0 \), we can obtain \( \| u(k) \| < M_b [\alpha + \| g(k) \|] \leq 2\alpha M_b < \infty \quad \text{for all} \quad k \geq 0 \), which indicates the boundary of control input (3) for finite-time trajectory tracking. For the case that \( y(0) \neq g(0) \), we can still get the same result, which is omitted here.

It is worth mentioning that although control law (4) can design when the system output \( y(k) \) starts to completely track \( g(k) \) by choosing proper tracking instant \( M \geq 1 \), which is its contribution, we cannot guarantee that the norm of the tracking error decreases for all \( 0 \leq k \leq (M - 1) \). As a
consequence, this control method may cause troubles in some particular situations. To overcome this problem, we give an additional assumption below.

Assumption 2: For system (1), \( \forall y(k), \hat{y}(k) \in \mathbb{R}^n \), assume that \( f(\cdot) \) satisfies
\[
||f(y(k)) - f(\hat{y}(k))|| \leq L_f ||y(k) - \hat{y}(k)||,
\]
where \( 0 < L_f < 1 \) is a constant number.

Before continuing the discussion, we introduce a constant number \( 0 < \epsilon < \infty \) that can be designed to adjust the instant when the trajectory tracking is accomplished.

**Theorem 2:** Suppose that system (1) satisfies Assumptions 1 and 2. Then, for any given \( y(0) \) and \( g(k) \in S_1 \), the output \( y(k) \) of system (1) can completely track \( g(k) \) in finite time.

In addition, for given constant number \( 0 < \epsilon < \infty \), when \( ||y(0) - g(0)|| > \epsilon \), there must exist a control law which can make \( ||y(k) - g(k)|| \) decrease monotonically before the trajectory tracking is accomplished.

**Proof:** For the case of \( ||y(0) - g(0)|| \leq \epsilon \), if Assumption 1 is satisfied, we can still use control law (3), which easily leads to \( y(k) = g(k) (k \geq 1) \). In particular, when \( y(0) = g(0) \), the system output \( y(k) \) completely tracks \( g(k) \) instantaneously.

When \( ||y(0) - g(0)|| > \epsilon \), let
\[
\eta = \log_{L_f} \left( \frac{\epsilon}{||y(0) - g(0)||} \right), \quad M_{\eta} = [\eta] + 1, \quad (6)
\]
where \([\eta]\) denotes the largest integer less than or equal to \( \eta \). Then, we design the control law:
\[
u(k) = \begin{cases} 
B^{-1}(y(k)) \left[ g(k+1) - f(y(k)) \right], & (0 \leq k \leq M_{\eta} - 1); \\
B^{-1}(y(k)) \left[ g(k+1) - f(y(k)) \right], & (k \geq M_{\eta}).
\end{cases} \tag{7}
\]
From (1) and (7), for \( 0 \leq k \leq (M_{\eta} - 1) \), the system output is
\[
y(k+1) = g(k+1) + f(y(k)) - f(g(k)).
\]
By repeatedly using the inequality in Assumption 2, we have
\[
||y(k+1) - g(k+1)|| = ||f(y(k)) - f(g(k))|| \\
\leq L_f ||y(k) - g(k)|| \\
\leq L_f ||f(y(k-1)) - f(g(k-1))|| \\
\leq L_f^2 ||g(k-1) - g(k-1)|| \\
\leq \cdots \\
\leq L_f^{k+1} ||y(0) - g(0)||.
\]
Since \( 0 < L_f < 1 \), \( ||y(k+1) - g(k+1)|| \leq ||y(k) - g(k)|| \).

Therefore, \( ||y(k) - g(k)|| \) decreases monotonically for each \( 0 \leq k \leq (M_{\eta} - 1) \).

According to the above, it is reasonable to suppose that
\[
||y(M_{\eta}) - g(M_{\eta})|| \leq L_f^{M_{\eta}} ||y(0) - g(0)|| \leq \epsilon.
\]
Let \( L_f^{M_{\eta}} ||y(0) - g(0)|| = \epsilon \). Then, \( \eta = \log_{L_f} \left( \frac{\epsilon}{||y(0) - g(0)||} \right) \).

Let \( M_{\eta} = [\eta] + 1 \), and we can get (6).

From (1) and (7), when \( k \geq (M_{\eta} + 1) \), \( y(k) = g(k) \). But, it is not certain whether \( y(M_{\eta}) = g(M_{\eta}) \). For this reason, we can only say that the system output \( y(k) \) can completely track \( g(k) \) after \( k = M_{\eta} \).

In summary, the output \( y(k) \) of system (1) can completely track \( g(k) \) in finite time.

It is the same as we discussed after the proof of Theorem 1, that control input (7) should also be bounded for bounded trajectory to be tracked. Owing to the limitation of space, we do not go into details here.

The contribution of control law (7) is that it not only can design the instant when the trajectory tracking is accomplished; but also can make the tracking error decrease monotonically before the system output starts to completely track the expected trajectory. Here, the constant number \( \epsilon \) plays a role as the threshold in determining the control switching point, after which the two trajectories begin to coincide with each other. Furthermore, when \( ||y(0) - g(0)|| > \epsilon \), since \( 0 < L_f < 1 \), \( \eta > 0 \). If \( \epsilon \) increases, \( \frac{||y(0) - g(0)||}{\epsilon} \) will be closer to 1 and \( \eta \) will be smaller, which means a smaller \( M_{\eta} \) and a faster tracking speed.

At the end of this section, we would like to point out that control laws (3), (4) and (7) are designed in a constructive way, and naturally, we can design the control laws in some other forms, which play the same roles in finite-time trajectory tracking control.

**III. Finite-Time Partial Output Trajectory Tracking Control**

In practical engineering, the case of \( m = n \) is rare, and the common case is that \( m < n \). But when \( m < n \), we can only control \( m \) components of the system output, while keeping the other \( n - m \) components bounded. Consider the system:
\[
\begin{align*}
y(k+1) &= f_1(y(k)) + f_m(y(k)) + G(y(k)) y_k(k) \\
y_m(k+1) &= f_m(y_m(k)) + G_m(y(k)) u_m(k).
\end{align*} \tag{9}
\]
where \( u(k) \in \mathbb{R}^m \), \( y(k) \in \mathbb{R}^n \) with \( m < n \) are the input and the output of system (9), respectively. In (9), we denote
\[
\begin{align*}
y(k) &= (y_1, y_2, \ldots, y_m, y_{m+1}, \ldots, y_n)^T, \\
y_m(k) &= (y_m, y_{m+1}, \ldots, y_n)^T, \\
y(k) &= (y_1, y_2, \ldots, y_m)^T, \\
f_1(y(k)) &= f_1(y_1), f_2(y_2), \ldots, f_m(y_m)(k), \ldots, f_n(y_n)(k))^T, \\
f_m(y(k)) &= f_m(y_m)(k), f_{m+1}(y_{m+1})(k), \ldots, f_n(y_n)(k))^T,
\end{align*}
\]
where \( f_1 : \mathbb{R}^n \to \mathbb{R}^{n-m} \), \( f_m : \mathbb{R}^n \to \mathbb{R}^m \) with \( f_1(0) = 0, f_m(0) = 0 \), \( f_i(y(k)) (1 \leq i \leq n) \) are all piecewise continuous with respect to \( y \); \( G(y(k)) \in \mathbb{R}^{m \times m} \) is the continuous input gain matrix.

In this section, we study how to control the partial output of system (9) to completely track any given bounded trajectory in finite time. Define the set of the trajectories to be tracked:
\[
S_2 \triangleq \{ h(k) \mid k \geq 0, \ h(k) \in \mathbb{R}^m, \ ||h(k)|| \leq \beta < \infty \}. \tag{11}
\]
\( h(t) \) is continuous with respect to \( t \in \mathbb{R}^+ \) and \( h(t) \) is the discrete value of \( h(t) \) at \( t = kT \). In this way, \( 0 < \beta < \infty \) is the boundary of any \( h(k) \in S_2 \). Moreover, we suppose that each given \( h(k) \in S_2 \) is known for all \( k \geq 0 \).
Assumption 3: For system (9), assume that \( \forall y(k) \in \mathbb{R}^n \),
1) \( \|f_1(y(k))\| \leq L_1 \|y(k)\| + L_2 \|y(n)\| \), where \( 0 < L_1 < 1 \)
and \( 0 \leq L_2 < \infty \) are constant numbers;
2) \( \text{Rank } [G(y(k))] = m \).

From 2) of Assumption 3, \( y_n(k) \) is controllable. To facilitate discussion, we suppose that \( \|y_n(0)\| \leq \beta \) for any given \( y_n(0) \); otherwise, we may first control this partial output to satisfy this initial condition, and then discuss the tracking problem.

Theorem 3: Suppose that system (9) satisfies Assumption 3. Then, for any given \( y_n(0), y_n(0) \) and \( h(k) \in S_2 \), the partial output \( y_n(k) \) of system (9) can completely track \( h(k) \) in finite time. In the mean time, the other partial output \( y(k) \) is bounded for all \( k \geq 0 \).

Proof: When \( y_n(0) = h(0) \), design the control input as
\[
  u(k) = G^{-1}(y(k)) \left[ h(k+1) - f_u(y(k)) \right], \quad (k \geq 0).
\]  
(12)

From (9) and (12), \( y_n(k) = h(k) \) \((k \geq 1)\). In this way, \( y_n(0) \) completely tracks \( h(k) \) instantaneously.

When \( y_n(0) \neq h(0) \), preset a tracking instant \( M \geq 1 \) when the trajectory tracking is just accomplished. Then, design the control input as
\[
  u(k) = \begin{cases} 
    G^{-1}(y(k)) \left[ \frac{k+1}{M} h(k+1) - f_u(y(k)) \right], & (0 \leq k \leq M-1); \\
    G^{-1}(y(k)) \left[ h(k+1) - f_u(y(k)) \right], & (k \geq M). 
  \end{cases}
\]  
(13)

From (9) and (13), we shall have
\[
  y_n(k+1) = \begin{cases} 
    \frac{k+1}{M} h(k+1), & (0 \leq k \leq M-1); \\
    h(k+1), & (k \geq M). 
  \end{cases}
\]  
(14)

When \( k = M - 1 \), \( y_n(k+1) = y_n(M) = h(M) \); and when \( k \geq M, y_n(k+1) = h(k+1) \). Then, \( y_n(k) \) completely tracks \( h(k) \) from \( k = M \) on.

In summary, the partial output \( y_n(k) \) of system (9) can completely track any given \( h(k) \in S_2 \) in finite time.

Next, we prove the boundedness of \( y_n(k) \). Here, we choose the case of \( y_n(0) \neq h(0) \) to demonstrate the proof. The proof for the other case is similar, and thus is omitted here.

By repeatedly using the inequality in Assumption 3,
\[
  \|y_n(k+1)\| \leq \|f_1(y(k))\| \leq L_1 \|y(k)\| + L_2 \|y(n)\| \leq L_1^{k+1} \|y(0)\| + L_2 \sum_{j=0}^{k} L_1^{k-j} \|y(j)\|. 
\]  
(15)

From (14) and (15), since \( 0 < L_1 < 1 \) and \( \|h(k)\| \leq \beta \) for all \( k \geq 0 \), then for \( 0 \leq k \leq (M - 1) \),
\[
  \|y_n(k+1)\| 
  \leq L_1^{k+1} \|y(0)\| + L_2 L_1^k \|y(0)\| + L_2 \sum_{j=1}^{k} L_1^{k-j} \frac{j}{M} \|h(j)\| 
  < \|y(0)\| + L_2 \|y(0)\| + (M - 1) \beta L_2 \frac{1}{M} < \infty. 
\]

From (14) and (15), for \( k \geq M \), we shall have
\[
  \|y_n(k+1)\| 
  \leq L_1^{k+1} \|y(0)\| + L_2 L_1^k \|y(0)\| + \beta L_2 \sum_{j=1}^{M-1} L_1^{M-1-j} \frac{j}{M} 
  + \beta L_2 \sum_{j=M}^{k} L_1^{k-j} 
  < \|y(0)\| + L_2 \|y(0)\| + \beta L_2 \sum_{j=M}^{k} L_1^{k-j} 
  \leq \|y(0)\| + L_2 \|y(0)\| + \beta \frac{L_2}{1 - L_1} < \infty. 
\]

In summary, the partial output \( y_n(k) \) of system (9) is bounded for all \( k \geq 0 \).

So far, we only discussed the boundedness of \( y_n(k) \). Here, we will study the boundedness of \( y_n(k) \). When \( y_n(0) = h(0) \),
\[
  \|y_n(k)\| \leq \|h(k)\| \leq \beta \quad (1 \leq k \leq M); \\
  \|y_n(k)\| \leq \beta, \quad (k \geq M + 1). 
\]

Since we have supposed that \( \|y_n(0)\| \leq \beta \) for any given \( y_n(0) \),
\[
  \|y_n(k)\| \leq \beta \quad \text{for all } k \geq 0. 
\]
In summary, the partial output \( y_n(k) \) also has a boundary of \( 0 < \beta < \infty \).

Although not specifically mentioned above, the subsystem \( y_n(k+1) = f_u(y(k)) + G(y(k)) u(k) \) is supposed to be globally asymptotically stable. Otherwise, according to Assumption 3, this subsystem is stabilizable, we can stabilize it first and then study the tracking problem. Notice that whether control inputs (12) and (13) are bounded is not discussed. But, the control input should always be constrained in engineering applications, and we will discuss it below.

Suppose that \( \sup_{y(k) \in \mathbb{R}^n} \left\|G^{-1}(y(k))\right\| = M_G < \infty \). Then if \( y_n(0) = h(0) \), from (11) and (12), \( u(k) \) should satisfy
\[
  \|u(k)\| \leq \|G^{-1}(y(k))\| \left[ \|h(k+1)\| + \|f_u(y(k))\| \right] \leq M_G \|\beta + f_u(y(k))\|, 
\]

When the subsystem \( y_n(k+1) = f_u(y(k)) + G(y(k)) u(k) \) is globally asymptotically stable, \( \|f_u(y(k))\| < \|y_n(k)\| \) with the conclusion that \( \|y_n(k)\| \leq \|h(k)\| \leq \beta \) for all \( k \geq 0 \), we can obtain \( \|u(k)\| < M_G \|\beta + f_u(y(k))\| \leq 2 \beta M_G \) for all \( k \geq 0 \), which indicates the boundary of control input (12) for finite-time trajectory tracking control. For control input (13) where \( y_n(0) \neq h(0) \), we can still get the same result, which is omitted here.

It is to be observed that when \( y_n(0) \neq h(0) \), control law (13) cannot guarantee that the norm of the tracking error decreases for all \( 0 \leq k \leq (M - 1) \), which could cause problems in some particular situations. To overcome this shortcoming, we give an additional assumption below.

Assumption 4: For system (9), assume that \( \forall y(k), y(n) \in \mathbb{R}^n \), function \( f_u(\cdot) \in \mathbb{R}^n \) satisfies
\[
  \|f_u(y(k)) - f_u(y(k))\| \leq L_u \|y_n(k) - y_n(k)\|, 
\]
where \( 0 < L_u < 1 \) is a constant number, and \( y_n(k) \) is the same form of \( y_n(k) \) in (10).
As in Theorem 2, we would like to introduce a constant number $0 < \delta < \infty$ that can be designed to adjust the instant when the trajectory tracking is accomplished.

**Theorem 4:** Suppose that system (9) satisfies Assumptions 3 and 4. Then, for any given $y_l(0), y_u(0)$ and $h(k) \in S_2$, the partial output $y_l(k)$ of system (9) can completely track $h(k)$ in finite time. In addition, for given constant numbers $0 < \delta < \infty$, when $\|y_l(0) - h(0)\| > \delta$, there must exist a control law which can make $\|y_l(k) - h(k)\|$ decrease monotonically before the trajectory tracking is accomplished. In the mean time, the other partial output $y_u(k)$ is bounded for all $k \geq 0$.

**Proof:** When $\|y_l(0) - h(0)\| \leq \delta$, we design the control input as follows.

\[
u(k) = G^{-1}(y(k)) \left[ h(k+1) - f_u(y(k)) \right] \quad (k \geq 0). \quad (16)
\]

From (9) and (16), we can infer that $y_l(k) = h(k)$ $(k \geq 1)$. In particular, if $y_l(0) = h(0)$, the partial output $y_l(k)$ can completely track $h(k)$ instantaneously.

When $\|y_l(0) - h(0)\| > \delta$, let

\[
\theta = \log_{L_\theta} \frac{\delta}{\|y_l(0) - h(0)\|}, \quad M_\theta = [\theta] + 1, \quad (17)
\]

\[
H(k) = \left[ I_{\sin}, h^T(k) \right] T \in \mathbb{R}^n,
\]

where $[\theta]$ denotes the largest integer less than or equal to $\theta$. Then, we design the control input as

\[
u(k) = \begin{cases} 
G^{-1}(y(k)) \left[ h(k+1) - f_u(H(k)) \right], \\
(0 \leq k \leq M_\theta - 1); \\
G^{-1}(y(k)) \left[ h(k+1) - f_u(y(k)) \right], \\
(k \geq M_\theta).
\end{cases}
\]

(18)

From (9) and (18), when $0 \leq k \leq (M_\theta - 1)$,

\[y_u(k+1) = h(k+1) + [f_u(y(k)) - f_u(H(k))].
\]

By repeatedly using the inequality in Assumption 4, we have

\[
\|y_u(k+1) - h(k+1)\| = \|f_u(y(k)) - f_u(H(k))\|
\leq L_{\sin}\|y_u(k) - h(k)\|
\leq L_{\sin}\|f_u(y(k-1)) - f_u(H(k-1))\|
\leq L_{\sin}\|y_u(k-1) - h(k-1)\|
\leq \cdots
\leq L_{\sin}^{M_\theta}\|y_u(0) - h(0)\|.
\]

From the above discussion, it is reasonable to suppose that

\[\|y_u(M_\theta) - h(M_\theta)\| \leq L_{\sin}^{M_\theta}\|y_u(0) - h(0)\| \leq \delta.
\]

Let $L_{\sin}^{M_\theta}\|y_u(0) - h(0)\| = \delta$. Then, $\theta = \log_{L_\theta}\|y_u(0) - h(0)\|).

Let $M_\theta = [\theta] + 1$, and then we get (17).

From (9) and (18), when $k \geq (M_\theta + 1)$, $y_u(k) = h(k)$. But, it is not certain whether $y_l(M_\theta) = h(M_\theta)$. For this reason, we can only say that $y_u(k)$ completely tracks $h(k)$ after $k = M_\theta$.

In summary, the partial output $y_u(k)$ of system (9) can completely track $h(k)$ in finite time.

The rest of the proof is about the boundedness of the other partial output $y_l(k)$, which is similar to that part of the proof of Theorem 3, and thus is omitted here.

It is the same as we discussed after the proof of Theorem 3, that the partial output $y_l(k)$ as well as control inputs (16) and (18) are also bounded for bounded trajectory to be tracked. Owing to the space limitation, we do not discuss it here.

In 1) of Assumption 3, we assumed that $0 < L_1 < 1$. One may be interested in what would happen in other cases. To this end, we define a new set of trajectories to be tracked:

\[
S_3 \triangleq \left\{ p(k) \right| k \geq 0, \quad p(k) \in \mathbb{R}^m, \quad \sum_{j=0}^{\infty} \|p(j)\| \leq \phi < \infty \}.
\]

\[
p(t) \text{ is continuous with respect to } t \in \mathbb{R}^+, \quad p(k) \text{ is the discrete value of } p(t) \text{ at } t = kT. \text{In this way, } 0 < \phi < \infty \text{ is the boundary of any } p(k) \in S_3. \text{Moreover, we suppose that each given } p(k) \in S_3 \text{ is known for all } k \geq 0.
\]

**Assumption 5:** For system (9), assume that $\forall y(k) \in \mathbb{R}^n$,

1) $\|f_1(y(k))\| \leq \|y(k)\| + L_2\|y(k)\|$, where $0 \leq L_2 < \infty$ is a constant number;

2) Rank $[G(y(k))] = m$.

**Corollary 1:** Suppose that system (9) satisfies Assumption 5. Then, for any given $y_l(0), y_u(0)$ and $p(k) \in S_3$, the partial output $y_u(k)$ of system (9) can completely track $p(k)$ in finite time. In the mean time, the other partial output $y_l(k)$ is bounded for all $k \geq 0$.

**Proof:** When $y_l(0) = p(0)$, design the control input as

\[
u(k) = G^{-1}(y(k)) \left[ p(k+1) - f_u(y(k)) \right] \quad (k \geq 0). \quad (21)
\]

From (9) and (21), $y_u(k) = p(k) (k \geq 1)$. In this way, $y_u(k)$ completely tracks $p(k)$ instantaneously.

When $y_l(0) \neq p(0)$, preset a tracking instant $M \geq 1$ when the trajectory tracking is just accomplished. Then, design the control input as

\[
u(k) = \begin{cases} 
G^{-1}(y(k)) \left[ \frac{k+1}{M} p(k+1) - f_u(y(k)) \right], \\
(0 \leq k \leq M - 1); \\
G^{-1}(y(k)) \left[ p(k+1) - f_u(y(k)) \right], \\
(k \geq M).
\end{cases}
\]

(22)

From (9) and (22), we shall have

\[y_u(k+1) = \begin{cases} 
\frac{k+1}{M} p(k+1), & (0 \leq k \leq M - 1); \\
p(k+1), & (k \geq M).
\end{cases}
\]

When $k = M - 1$, $y_u(k+1) = y_u(M) = p(M)$; when $k \geq M$, $y_u(k+1) = p(k+1)$. Then, $y_u(k)$ completely tracks $p(k)$ from $k = M$ on. In summary, the partial output $y_u(k)$ of system (9) can completely track $p(k)$ in finite time.

Next, we prove the boundedness of $y_l(k)$. Here, we choose the case of $y_l(0) \neq p(0)$ to demonstrate the proof. The proof for the other case is similar, and thus is omitted here.
By repeatedly using the inequality in (1) of Assumption 5,
\[ \|y_n(k+1)\| = \|f_1(y(k))\| \leq \|y_n(k)\| + L_2 \|y_n(k)\| \]
\[ \leq \|y_n(0)\| + L_2 \sum_{j=0}^{k} \|y_n(j)\|. \]  
(24)

From (23) and (24), since \( \sum_{j=0}^{\infty} \|p(j)\| \leq \phi \) and \( \frac{j+1}{M} \leq 1 \) for \( 0 \leq j \leq (M-1) \), we can obtain that \( \forall k \geq 0 \),
\[ \|y_n(k+1)\| \leq \|y_n(0)\| + L_2 \sum_{j=0}^{k} \|p(j)\| \]
\[ < \|y_n(0)\| + L_2 \|y_n(0)\| + L_2 \sum_{j=0}^{\infty} \|p(j)\| \]
\[ \leq \|y_n(0)\| + L_2 \|y_n(0)\| + \phi L_2 < \infty. \]

Then, the partial output \( y_n(k) \) is bounded for all \( k \geq 0 \). ■

As in Theorems 2 and 4, we also introduce a constant number \( 0 < \xi < \infty \) that can be designed to adjust the instant when the trajectory tracking is accomplished.

**Corollary 2:** Suppose that system (9) satisfies Assumptions 4 and 5. Then, for any given \( y_n(0), y_n(0) \) and \( p(k) \in S_3 \), the partial output \( y_n(k) \) of system (9) can completely track \( p(k) \) in finite time. In addition, for given constant number \( 0 < \xi < \infty \), when \( \|y_n(0) - p(0)\| > \xi \), there must exist a control law which can make \( \|y_n(k) - p(k)\| \) decrease monotonically before the trajectory tracking is accomplished. In the mean time, the other partial output \( y_2(k) \) is bounded for all \( k \geq 0 \).

The proof is similar to the proofs of Theorem 4 and Corollary 1. When \( \|y_n(0) - p(0)\| > \xi \), we let
\[ \mu = \log_{10} \left( \frac{\xi}{\|y_n(0) - p(0)\|} \right), \quad M_\mu = [\mu + 1], \]
\[ P(k) = \left[ g^T, \ t^T(k) \right]^T \in \mathbb{R}^n, \]
and we design the control input as
\[ u(k) = \begin{cases} 
G^{-1}(y(k)) \left[ p(k+1) - f_1\left( P(k) \right) \right], \\
(0 \leq k \leq M_\mu - 1); \\
G^{-1}(y(k)) \left[ p(k+1) - f_1(y(k)) \right], \\
(k \geq M_\mu). 
\end{cases} \]
(26)

By using control law (26), \( \|y_n(k) - p(k)\| \) decreases monotonically for each \( 0 \leq k \leq (M_\mu - 1) \), and the partial output \( y_n(k) \) completely tracks \( p(k) \in S_3 \) after \( k = M_\mu \).

The rest part of the proof is similar to the corresponding parts of the proofs of Theorem 4 and Corollary 1, and thus is not discussed in details here.

In practical engineering, there is a class of 2-dimensional systems which have the same structure of system (9). They describe the dynamics of an object that moves along the preset track, such as the linear motor systems [25], [26], etc. Their Euler discretized models are in the following form.
\[ \begin{cases} 
y_1(k+1) = y_1(k) + T y_2(k) \\
y_2(k+1) = f_2(y(k)) + G(y(k))u(k) \end{cases} \quad (k \geq 0), \]
(27)

where \( u(k) \in \mathbb{R}, y(k) = [y_1(k), y_2(k)]^T \in \mathbb{R}^2 \) are the input and the output, respectively. \( y_1(k) \) and \( y_2(k) \) represent the position and the velocity of the moving object, respectively. In system (27), \( m = 1, n = 2 \). \( f_2 : \mathbb{R}^2 \to \mathbb{R} \) is piecewise continuous with respect to \( y(k) \), \( f_2(0) = 0 \), and \( G(y(k)) \) is the continuous input gain.

For the moving objects described by (27), we will control the velocity \( y_2(k) \) to make the position \( y_1(k) \) completely track a given trajectory in finite time. Define the set of the trajectories to be tracked:
\[ S_4 \triangleq \left\{ y(k) \mid k \geq 0, \ y(k) \in \mathbb{R}, \ |y(k)| \leq \zeta < \infty \right\}. \]
(28)

\([\cdot]_\gamma\) denotes the absolute value in this paper. \( q(t) \) is continuous with respect to \( t \in \mathbb{R}^+ \), and \( k(q(t)) \) is the discrete value of \( q(t) \) at \( t = kT \). Then, \( 0 < \zeta < \infty \) is the boundary of any \( y(k) \in S_4 \). Besides, we suppose that each given \( q(k) \in S_4 \) is known for all \( k \geq 0 \).

**Theorem 5:** Suppose that in system (27), \( G(y(k)) \neq 0 \) for all \( y(k) \in \mathbb{R}^2 \). Then, for any given \( y_1(0), y_2(0), T > 0 \) and \( q(k) \in S_4 \), the partial output \( y_2(k) \) of system (27) can completely track \( q(k) \) in finite time. In the mean time, the other partial output \( y_2(k) \) is bounded for all \( k \geq 0 \).

**Proof:** Preset a tracking instant \( M \geq 3 \) when the partial output trajectory tracking is just accomplished, and let
\[ \gamma = \frac{2[q(M) - y_1(0) - T y_2(0)]}{T^2(M - 1)(M - 2)}. \]
(29)

Then, we design the control input as
\[ u(k) = \begin{cases} 
G^{-1}(y(k)) \left[ y_2(k) - f_2(y(k)) \right], \\
(0 \leq k \leq M - 1); \\
G^{-1}(y(k)) \left[ y_2(k) - f_2(y(k)) \right], \\
(k \geq M). 
\end{cases} \]
(30)

From (27) and (30), for \( 0 \leq k \leq (M-1) \), the partial output \( y_2(k+1) = \gamma kT \). From (29), we can obtain
\[ y_1(M) = y_1(M-1) + T y_2(M-1) \]
\[ = y_1(0) + T y_2(0) + \gamma T^2 \sum_{j=0}^{M-2} j \]
\[ = y_1(0) + T y_2(0) + \frac{1}{2} \gamma T^2 (M - 1)(M - 2) \]
\[ = q(M). \]
(31)

For \( k \geq M \), substitute (30) into (27), we shall have
\[ y_2(k+1) = \frac{1}{T} \left[ y_2(k) - y_2(k+1) \right], \]
y\[ y_1(k+1) = y_1(k) + \frac{1}{T} \left[ y_2(k) - y_2(k+1) \right]. \]
(32)

Observe (31) and (32), \( \forall k \geq M, y_1(k) = q(k) \). Then, the partial output \( y_1(k) \) can completely track any given \( q(k) \in S_4 \) from \( k = M \) on.

On the other hand, we can see that
\[ y_2(k) \leq \max \left\{ y_2(0), \gamma T(M - 1) \right\}, \quad (0 \leq k \leq M); \]
\[ y_2(k) \leq \frac{1}{T} \left[ \left| y_2(k) \right| + \left| q(k) \right| \right] \leq \frac{1}{T} \zeta, \quad (k \geq M + 1). \]

In summary, \( y_2(k) \) is bounded for all \( k \geq 0 \). ■
Note that the control laws presented in this section are designed in a constructive way. Naturally, we can design the control laws in some other forms, which play the same roles in finite-time trajectory tracking control.

IV. SIMULATION EXAMPLES

In this section, we will give some simulation examples to demonstrate the research results of this paper.

A. Example of Finite-Time Total Output Trajectory Tracking

For the total output case, where \( m = n \), we present the following example to illustrate Theorems 1 and 2.

**Example 1:** Consider a nonlinear discrete-time system in the form of (1), where

\[
\begin{align*}
\dot{y}(k) &= \begin{bmatrix}
\frac{1}{3}y_2(k) \\
\frac{1}{3}\text{sat}(y_1(k) + y_2(k)) \\
2 + \cos(y_1(k))
\end{bmatrix}, \\
B(y(k)) &= \begin{bmatrix}
1 \\
0 \\
-1
\end{bmatrix}.
\end{align*}
\]

(33)

Here, \( m = n = 2 \) and \( \text{sat}(\cdot) \) is the saturation function, which is defined as

\[
\forall x \in \mathbb{R}, \quad \text{sat}(x) = \begin{cases}
1, & \text{if } x > 1; \\
x, & \text{if } -1 \leq x \leq 1; \\
-1, & \text{if } x < -1.
\end{cases}
\]

(34)

We can see that both \( f(y(k)) \) and \( B(y(k)) \) are continuous with respect to \( y(k) \), and \( f(0) = 0 \). Moreover, \( \forall y(k) \in \mathbb{R}^2 \), \( \text{Rank}[B(y(k))] = 2 = n \) and \( \|B(y(k))\| \leq 3.5414 < \infty \). Thus, system (33) satisfies model (1) and Assumption 1.

Based on the fact that \( \forall a, b \in \mathbb{R}, \|\text{sat}(a) - \text{sat}(b)\| \leq |a - b| \), \( \forall y, \dot{y} \in \mathbb{R}^2 \), we shall have

\[
\|f(y) - f(\dot{y})\|^2 \\
\leq (y_2 - \dot{y}_2)^2 + \frac{1}{9}(y_1 + y_2 - \dot{y}_1 - \dot{y}_2)^2 \\
\leq \frac{1}{9}[(y_1 - \dot{y}_1)^2 + 2(y_1 - \dot{y}_1)(y_2 - \dot{y}_2) + 2(y_2 - \dot{y}_2)^2].
\]

Using the inequality that \( \forall a, b \in \mathbb{R}, a^2 + 2ab + 2b^2 \leq \lambda_{\max}\left[\begin{bmatrix}1 & 1 & 1\end{bmatrix}\right]\times\left[\begin{bmatrix}a \\
b\end{bmatrix}\right]\|^2 \),

we may conclude that

\[
\forall y, \dot{y} \in \mathbb{R}^2, \quad \|f(y) - f(\dot{y})\| \leq 0.5393\|y - \dot{y}\|.
\]

Here, we used a property of positive semidefinite symmetric matrices, which is \( y^T Py \leq \lambda_{\max}(P)y^Ty \) for all \( y \in \mathbb{R}^n \), where \( \lambda_{\max}(\cdot) \) is the maximum eigenvalue. Then, \( f(y(k)) \) satisfies Assumption 2, where \( L_f = 0.5393 \).

In this example, the trajectory to be tracked is given as

\[
g(k) = \begin{bmatrix}
3\sin(0.1kπ) \\
\cos(0.2kπ) \exp(-0.1k)
\end{bmatrix}, \quad (k \geq 0).
\]

(35)

Hence, \( g(0) = [0,1]^T \) and \( \|g(k)\| \leq \sqrt{10} \) for all \( k \geq 0 \), which satisfies the definition of \( S_1 \) in (2).

For both Theorems 1 and 2, we will use the same initial value \( y(0) = [2, -2]^T \), such that \( y(0) \neq g(0) \). To illustrate Theorem 1, we set \( M = 5 \). From (1), (4), (33) and (35),

\[
y(k) = \begin{cases}
\frac{k}{5}\begin{bmatrix}
3\sin(0.1kπ) \\
\cos(0.2kπ) \exp(-0.1k)
\end{bmatrix}, & (1 \leq k \leq 5); \\
\frac{k}{5}\begin{bmatrix}
3\sin(0.1kπ) \\
\cos(0.2kπ) \exp(-0.1k)
\end{bmatrix} + \begin{bmatrix}
0.5\sin(0.2kπ) \\
0.5\cos(0.2kπ)
\end{bmatrix}, & (k \geq 6).
\end{cases}
\]

(36)

We use Matlab program to simulate the trajectory tracking performances. The simulation results shown in Fig. 1 accord with Theorem 1, that \( y(k) \) can completely track \( g(k) \) from \( k = 5 \) on. However, as we discussed after Theorem 1, control law (4) cannot guarantee the monotonic decrease of the norm of the tracking error when \( 0 \leq k \leq 4 \).

Fig. 1. Simulation of Theorem 1 with \( M = 5 \).

For Theorem 2, let \( \varepsilon = 0.1 \). From (6), with \( L_f = 0.5393 \) and \( \|y(0) - g(0)\| = 3.6056 \), we shall have \( \eta = 5.8059 \) and \( M_\eta = 6 \). We use Matlab program to simulate the trajectory tracking performances according to (1), (7), (33) and (35). The simulation results shown in Fig. 2 accord with Theorem 2 that \( \|y(k) - g(k)\| \) decreases monotonically when \( 0 \leq k \leq 5 \), and \( y(k) \) completely tracks \( g(k) \) after \( k = 6 \).

Fig. 2. Simulation of Theorem 2 with \( \varepsilon = 0.1 \) and \( M_\eta = 6 \).

B. Example of Finite-Time Partial Output Trajectory Tracking

For the partial output case, where \( m < n \), we will give two examples to illustrate Theorems 3, 4, and 5, respectively.

**Example 2:** Consider a nonlinear discrete-time system in the form of (9), where

\[
\begin{align*}
f_1(y(k)) &= \frac{1}{2}y_1(k) + y_2(k), \\
f_2(y(k)) &= \text{sat}(0.5y_2(k)), \\
G(y(k)) &= 2 + \sin(y_1(k) - y_2(k)).
\end{align*}
\]

(37)
Here, \( m = 1, n = 2 \) and \( \text{sat}(\cdot) \) is defined in (34). \( f_1(y(k)) \) and \( G(y(k)) \) are all continuous with respect to \( y(k) = [y_1(k), y_2(k)]^T \in \mathbb{R}^2 \), and \( f_1(0) = 0, f_\Pi(0) = 0 \). Thus, system (37) satisfies the description of model (9). Besides, \( y_1, y_2 \in \mathbb{R}^2 \), we have
\[
\|f_1(y)\| \leq \frac{1}{3}|y_1| + |y_2|,
\|
f_\Pi(y) - f_\Pi(\hat{y})\| = |\text{sat}(0.5y_2) - \text{sat}(0.5\hat{y}_2)| \\
\leq 0.5|y_2 - \hat{y}_2|,
\]
\[\text{Rank } [G(y)] = 1.\]

Then, Assumptions 3 and 4 are both satisfied, with \( L_1 = \frac{1}{3} \) and \( L_\Pi = 0.5 \).

In this example, the trajectory to be tracked is given as
\[
h(k) = \cos(0.2k\pi) \exp(-0.1k) \quad (k \geq 0). \tag{38}
\]

Then, \( h(0) = 1 \) and \( |h(k)| \leq 1 \) for all \( k \geq 0 \), which satisfies the definition of \( S_2 \) in (11). We still use \( y(0) = [2, -2]^T \) for both Theorems 3 and 4, such that \( y_2(0) \neq h(0) \).

To illustrate Theorem 3, we set \( M = 5 \). Based on (9), (13), (37) and (38), the system output will be
\[
y_1(k+1) = \frac{1}{3}y_1(k) + y_2(k) \quad (k \geq 0),
\]
\[
y_2(k+1) = \begin{cases} 
\frac{k + 1}{5} \cos \left(\frac{0.2(k + 1)\pi}{5}\right) \exp \left(-0.1(k + 1)\right), & (0 \leq k \leq 4); \\
\cos \left(\frac{0.2(k + 1)\pi}{5}\right) \exp \left(-0.1(k + 1)\right), & (k \geq 5).
\end{cases}
\tag{39}
\]

We use Matlab program to simulate the trajectory tracking performances. Fig. 3 shows the simulation results, which accord with Theorem 3, that the partial output \( y_2(k) \) completely tracks \( h(k) \) from \( k = 5 \) on, and the other partial output \( y_1(k) \) is bounded for all \( k \geq 0 \). But, as we discussed after Theorem 3, control law (13) cannot guarantee that the norm of the tracking error decreases monotonically for each \( 0 \leq k \leq 4 \).

For Theorem 4, let \( \delta = 0.1 \). From (17), with \( L_\Pi = 0.5 \) and \( |y_2(0) - h(0)| = 3 \), we have \( \theta = 4.9069 \) and \( M_\theta = 5 \). We use Matlab program to simulate the trajectory tracking performances based on (9), (18), (37) and (38). The simulation results shown in Fig. 4 accord with Theorem 4, that \( y_2(k) \) completely tracks \( h(k) \) after \( k = 5 \), \( y_2(k) - h(k) \) decreases monotonically when \( 0 \leq k \leq 4 \), and the other partial output \( y_1(k) \) is bounded for all \( k \geq 0 \).

Next, we employ a permanent magnet linear motor (PMLM) system as an example to illustrate Theorem 5. The structure of this system is shown in Fig. 5.

Example 3: The mathematical model of the PMLM system can be described by a differential equation [25], [26]:
\[
\ddot{x}(t) = \frac{1}{M_s} \left[ -K_f \dot{x}(t) + K_f u(t) - F \right], \tag{40}
\]
where \( x(t) \in [0, 1] \) (in m) is the motor position; \( u(t) \) (in V) is the control voltage; \( K_f \) (in N/A) is the amount of force produced by the motor; \( K_e \) (in V·s·m\(^{-1}\)) is the back EMF voltage; \( R \) (in Ohm) is the total resistance; \( M_s \) (in Kg) is the moving block mass; \( F \) (in N) is the disturbing force; and
\[
F = f_{\text{fric}} + f_{\text{ripple}} + f_{\text{load}}.
\]

In (41), \( f_{\text{fric}} \) and \( f_{\text{ripple}} \) (in N) denote the frictional force and the ripple force, respectively; while \( f_{\text{load}} \) (in N) is the loading force, and in this example we set \( f_{\text{load}} \equiv 0 \). \( f_{\text{e}} \) (in V) is the Coulomb friction coefficient; \( f_{\text{c}} \) (in V) is the static friction coefficient; \( v_s \) (in m/s) denotes the lubricant parameter, which may be determined by empirical experiments; \( f_{\text{v}} \) (in V·s·m\(^{-1}\)) is the viscous friction coefficient; \( A_r \) and \( \omega \) are constants. These parameters of the PMLM system are introduced in Table I. The peak terminal voltage is \( |u|_{\text{max}} = 110 \text{V} \); the peak velocity is \( |\dot{x}|_{\text{max}} = 2.6 \text{m/s} \); and the peak acceleration is \( |\ddot{x}|_{\text{max}} = 140 \text{m/s}^2 \).

Let \( y = [y_1, y_2]^T = [x, \dot{x}]^T \) denote the output. \( T > 0 \) is the sampling period, and we choose \( T = 0.1 \text{s} \) for this example.
TABLE I  
PARAMETERS OF THE PERMANENT MAGNET LINEAR MOTOR SYSTEM

<table>
<thead>
<tr>
<th>Contents</th>
<th>Units</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_f$</td>
<td>N/A</td>
<td>130</td>
</tr>
<tr>
<td>$R$</td>
<td>Ohm</td>
<td>16.8</td>
</tr>
<tr>
<td>$K_e$</td>
<td>V·s·m$^{-1}$</td>
<td>123</td>
</tr>
<tr>
<td>$M_s$</td>
<td>Kg</td>
<td>5.4</td>
</tr>
<tr>
<td>$f_s$</td>
<td>V</td>
<td>1.3</td>
</tr>
<tr>
<td>$v_s$</td>
<td>m/s</td>
<td>0.1</td>
</tr>
<tr>
<td>$f_c$</td>
<td>V·s·m$^{-1}$</td>
<td>1.3</td>
</tr>
<tr>
<td>$A_r$</td>
<td></td>
<td>1.1</td>
</tr>
<tr>
<td>$\omega$</td>
<td>s$^{-1}$</td>
<td>100$\pi$</td>
</tr>
</tbody>
</table>

For convenience, we denote
\[
C = \frac{TK_f}{RM_e}, \quad C_1 = (1 - CK_e), \quad C_2 = CA_r, \quad C_3 = Cf_e, \quad C_4 = C (f_s - f_c), \quad C_5 = Cf_c. \tag{42}
\]

Then, we may discretize system (40) using Euler’s method:
\[
\begin{align*}
    y_1(k+1) &= y_1(k) + T y_2(k), \\
    y_2(k+1) &= f_2(y(k)) + C u(k),
\end{align*}
\]

where $0 \leq y_1 \leq 1$, $-2.6 \leq y_2 \leq 2.6$, $-110 \leq u \leq 110$, and
\[
    f_2(y(k)) = C_1 y_2(k) - C_2 \sin (\omega y_1(k)) - \left\{ C_3 + C_5 y_2(k) + C_4 \exp \left[ - (y_2(k)/v_s)^2 \right] \right\} \text{sgn}(y_2(k)).
\]

Here, $m_1 = 1$, $n = 2$, $f_2 : \mathbb{R}^2 \to \mathbb{R}$ is piecewise continuous with respect to $y(k)$, $f_2(0) = 0$, which satisfies the description of model (27), and $C = 0.1433$.

The trajectory to be tracked is given as
\[
q(k) = 0.2 + \sin(0.01k\pi) \exp(-0.02k) \quad (k \geq 0). \tag{45}
\]

Then, $q(0) = 0.2$ and $|q(k)| \leq 0.6452$ for all $k \geq 0$, which satisfies the definition of $S_1$ in (28).

Set the initial value $y(0) = [0, 0]^T$, and set the simulation period to 10s (i.e., 100T). In this way, the moving block will return to its original position. We arbitrarily choose $M_1 = 12$ and $M_2 = 20$, to show the effectiveness of control law (30).

We use Matlab program to simulate the trajectory tracking performances according to (30), (43), (44) and (45). The simulation results shown in Fig.s 6 and 7 accord with Theorem 5, that $y_1(k)$ completely tracks $q(k)$ from $k = 12$ (or, $k = 20$) on, and $y_2(k)$ is bounded for all $k \geq 0$.

\[\text{Fig. 6. Trajectories of } q(k) \text{ and } y_1(k) \text{ with } M_1 = 12 \text{ and } M_2 = 20.\]

\[\text{Fig. 7. Trajectories of } y_2(k) \text{ with } M_1 = 12 \text{ and } M_2 = 20.\]  

V. CONCLUSIONS

In this paper, we studied the finite-time output trajectory tracking problem of a class of nonlinear discrete-time systems, for both the total output case and the partial output case. The objective of our study is to make the system output (or, part of it) completely track the expected trajectory in finite time (while keeping the other part of the system output bounded). The trajectories to be tracked can be any given bounded trajectories, and there are no further special requirements on these expected trajectories.

The control methods given in the proofs of Theorems 1, 3, 5 and Corollary 1 can design the tracking instant when the (partial) output trajectory and the expected trajectory begin to coincide with each other, but cannot guarantee the monotonic decrease of the norm of the tracking error before that instant. The control methods given in the proofs of Theorems 2, 4 and Corollary 2 not only can determine the tracking instant after which the trajectory tracking is accomplished, but also can make the norm of the tracking error decrease monotonically before that tracking instant. The control methods given in Theorems 3, 4, 5 and Corollaries 1, 2 can keep the rest part of the system output bounded. Besides, it is worth mentioning that these control methods are all designed in a constructive way, and they may be designed in some other forms which play the same roles in trajectory tracking. Matlab simulation results demonstrated the effectiveness of these control methods.

Our research results have both theoretical value and practical significance, while some further work can be done. In the future, we would like to study the systems which have a more general form than models (1) and (9). The theorems proposed in this paper only give sufficient conditions for finite-time trajectory tracking, and we may study whether there are sufficient and necessary conditions.

REFERENCES


