A Generalized Time Transformation Method for Finite-Time Control*

Dzung Tran[†] and Tansel Yucelen[‡]

***Abstract*— This paper introduces a new class of scalar, time-varying gain functions entitled as "generalized finite-time gain functions" for converting a (original) baseline control algorithm into a time-varying one, which can be used for time-critical applications. That is, the convergence time τ can be directly assigned by users. The relationship between a generalized finite-time gain function and its corresponding generalized time transformation function is established such that one can be obtained using the other one. Thanks to the generalized time transformation function, the resulting time-varying algorithm over the time interval $[0, \tau)$ is transformed to an equivalent algorithm over the stretched infinite-time interval $[0, \infty)$ for stability analysis. In addition, conditions to guarantee the convergence of the state as well as the boundedness of its time derivative are given. Finally, we present a numerical example to illustrate the efficacy of the proposed finite-time control methodology.**

I. INTRODUCTION

Finite-time control is a research field with a rich literature; however, the standard algorithms for finite-time convergence depend on dynamical systems' initial conditions (see, for example, [1]–[7]); thus, the convergence time τ may not be directly assigned by a control engineer. In order to solve this problem, the authors of, for example, [8]–[14] focus on finding the upper bound for the convergence time. Notably, some recent results allow users to assign the convergence time τ to the finite-time algorithms utilized in time-critical applications (see, for example, [15]–[27], where we also refer to the introduction sections of [15]–[17] for related discussions). This paper's results contribute to the recent studies documented in [15], [16] utilizing the time transformation method.

Here, we focus on finite-time control of perturbed dynamical systems based on the time transformation method. Specifically, a new class of scalar, time-varying gain functions entitled as "generalized finite-time gain functions" is introduced for converting a (original) baseline control algorithm into a time-varying one, which can be used for time-critical applications; that is, the control algorithm is executed over a prescribed time interval $[0, \tau)$ and meet the objective at τ, the convergence time assigned by the control engineer. The connection between a generalized finite-time gain function and its corresponding generalized time transformation function is established such that one can be obtained using the other one. Since analyzing a time-varying system over the time interval $[0, \tau)$ can be generally difficult, the generalized time transformation function is utilized to transform the resulting time-varying algorithm over the time interval $[0, \tau)$ to an equivalent algorithm over the stretched infinite-time interval $[0, \infty)$, which is easier to analyze. Furthermore, conditions to guarantee the convergence of the state as well as the boundedness of its time derivative are given. Finally, an application of our theoretical findings to a distributed control problem is presented.

This paper is organized as follows. We state the necessary preliminaries in Section II. The proposed generalized time transformation functions and the corresponding finite-time control problem over the prescribed time interval $[0, \tau)$ are then given in Section III. We then show the mentioned distributed control application in Section IV. Finally, Section V summarizes concluding remarks.

II. PRELIMINARIES

In this paper, $\mathbb{R}$ stands for the set of real numbers, $\mathbb{R}^n$ stands for the set of $n \times 1$ real column vectors, $\mathbb{R}_+^{n \times n}$ (resp., $\overline{\mathbb{R}}_+^{n \times n}$) stands for the set of $n \times n$ positive-definite (resp., positive semi-definite) real matrices, $\mathbf{1}_n$ stands for the $n \times 1$ vector of all ones, and I_n stands for the $n \times n$ identity matrix. We also use $(\cdot)^{\mathrm{T}}$ for transpose, $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ respectively for the minimum and maximum eigenvalue of a matrix A, $\mathrm{diag}(a)$ for diagonal matrix with vector a on its diagonal, $[x]_i$ for the entry of vector x on the *i*-th row, and A_{ij} for the entry of matrix A on the *i*-th row and *j*-th column. Moreover, we now recall several graph-theoretical notions (see [28] and [29] for details). Specifically, an undirected graph $\mathcal{G}$ is defined by a set $\mathcal{V}_{\mathcal{G}} = \{1, \ldots, N\}$ of nodes and a set $\mathcal{E}_{\mathcal{G}} \subset \mathcal{V}_{\mathcal{G}} \times \mathcal{V}_{\mathcal{G}}$ of edges. If $(i, j) \in \mathcal{E}_{\mathcal{G}}$, then the nodes i and j are neighbors and the neighboring relation is indicated with $i \sim j$. The number of agent i's neighbors is its degree and denoted as d_i. The degree matrix of a graph

[†]D. Tran is a Graduate Research Assistant of the Mechanical Engineering Department and a member of the Laboratory for Autonomy, Control, Information, and Systems (LACIS, `http://lacis.eng.usf.edu/`) at the University of South Florida, Tampa, Florida 33620, United States of America (email: `dtran3@mail.usf.edu`).

[‡]T. Yucelen is an Assistant Professor of the Mechanical Engineering Department and the Director of the Laboratory for Autonomy, Control, Information, and Systems (LACIS, `http://lacis.eng.usf.edu/`) at the University of South Florida, Tampa, Florida 33620, United States of America (email: `yucelen@usf.edu`).

*This research was supported by the Army Research Office under Grant W911NF-17-1-0582.

$\mathcal{G}$, $\mathcal{D}(\mathcal{G}) \in \mathbb{R}^{N\times N}$, is defined by $\mathcal{D}(\mathcal{G}) \triangleq \mathrm{diag}(d)$. Here, $d = [d_1, \ldots, d_N]^{\mathrm{T}}$. In addition, a path $i_0 i_1 \ldots i_L$ is a finite sequence of nodes such that $i_{k-1} \sim i_k$, $k = 1, \ldots, L$, and a graph $\mathcal{G}$ is said to be connected when there exists a path between any pair of distinct nodes. The adjacency matrix of a graph $\mathcal{G}$, $\mathcal{A}(\mathcal{G}) \in \mathbb{R}^{N\times N}$, is also defined by $[\mathcal{A}(\mathcal{G})]_{ij} = 1$ when $(i,j) \in \mathcal{E}_{\mathcal{G}}$ and $[\mathcal{A}(\mathcal{G})]_{ij} = 0$ otherwise. Finally, a graph's Laplacian matrix, $\mathcal{L}(\mathcal{G}) \in \overline{\mathbb{R}}_+^{N\times N}$, is defined by $\mathcal{L}(\mathcal{G}) \triangleq \mathcal{D}(\mathcal{G}) - \mathcal{A}(\mathcal{G})$.

III. A Generalized Time Transformation Method

In this paper, the perturbed dynamical system given by

$$\dot{x}(t) = \alpha(t) f\big(x(t)\big) + g(t, x(t)), \quad x(0) = x_0, \tag{1}$$

is considered, where $x(t) \in \mathbb{R}^n$ stands for the state vector, $\alpha(t) \in \mathbb{R}_+$ stands for a positive, time-varying scalar function entitled as "generalized finite-time gain function" (see below for details), $g(t, x(t)) \in \mathbb{R}^n$ stands for a bounded perturbation term that satisfies $\|g(t, x(t))\|_2 \leq g^*$, and $f\big(x(t)\big)$ is continuously differentiable and globally Lipschitz. Moreover, we assume the origin of the nominal dynamical system $\dot{x}(t) = f\big(x(t)\big)$ to be globally exponentially stable. Note that the nominal dynamics $\dot{x}(t) = f\big(x(t)\big)$ can also be viewed as error dynamics resulting from an original baseline control algorithm without perturbation (i.e., $g(t, x(t)) \equiv 0$) and with $\alpha(t) = 1$.

We now elucidate the above discussion in the following example. Specifically, consider a (baseline) scalar command following control algorithm $\dot{z}(t) = u(t)$ with $u(t) = -\big(z(t) - c(t)\big)$. Here, $z(t)$ stands for the state, $u(t)$ stands for the control, and $c(t)$ stands for a time-varying bounded command with bounded time rate of change. Let the error be $x(t) \triangleq z(t) - c(t)$, then we have $\dot{x}(t) = -x(t) - \dot{c}(t)$. When $c(t)$ is constant (i.e., $\dot{c}(t) = 0$), then $\dot{x}(t) = -x(t)$, where this is the so-called nominal dynamical system with $f\big(x(t)\big) = -x(t)$. Here, if we let $u(t) = -\alpha(t)\big(z(t) - c(t)\big)$ through multiplying the right hand side of the baseline algorithm with (the generalized finite-time gain function) $\alpha(t)$, we can obtain its time-varying version as $\dot{z}(t) = \alpha(t)\big(-z(t) + c(t)\big)$. In this case, the resulting error dynamics satisfies (1) as $\dot{x}(t) = \alpha(t)\big(-x(t)\big) - \dot{c}(t)$, where $g\big(t, x(t)\big) = -\dot{c}(t)$ is a bounded term.

This paper aims to construct a class of generalized finite-time gain functions $\alpha(t)$ and the corresponding conditions in order to guarantee that the solution $x(t)$ of (1) converges to zero as $t \to \tau$, where $\tau \in \mathbb{R}_+$ is a user-defined convergence time. To this end, we begin with the following assumption.

Assumption 1. *The generalized finite-time gain function $\alpha(t)$ satisfies:* • *$\alpha(t)$ is continuous differentiable on $t \in [0, \tau)$.* • *$\alpha(t) > m$ for all $t \in [0, \tau)$ and for some $m > 0$.* • *$\lim_{t\to\tau} \alpha(t) = \infty$.*

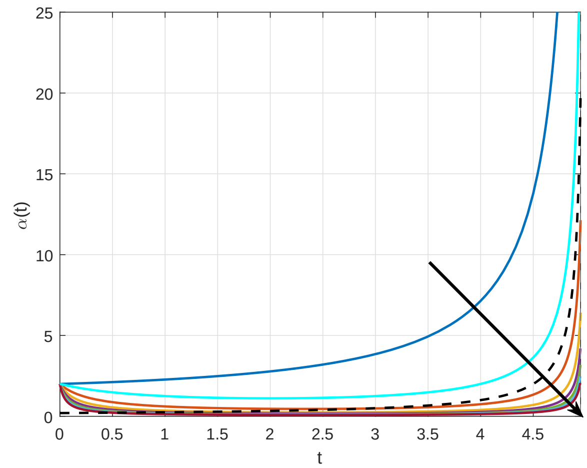

Fig. 1. Plots of the family of $\alpha(t)$ with $a = 0.1$, $m \in [0.01, 2]$ and $\tau = 5$, where the arrow pointing in the increasing direction of m. The dashed line represents $\alpha(t) = 1/(\tau - t)$ for comparison purpose.

With a generalized finite-time gain function $\alpha(t)$ subject to Assumption 1, we can construct the resulting generalized time transformation function $t = \theta(s)$; see the next lemma.

Lemma 1. *Consider a generalized finite-time gain function $\alpha(t)$ that satisfies Assumption 1 and: i) $\frac{dt}{ds} = \frac{d(\theta(s))}{ds} = \frac{1}{\alpha(\theta(s))}$ (i.e, $\alpha(\theta(s))d(\theta(s)) = ds$). ii)) $\theta(0) = 0$. iii) $\lim_{s\to\infty} \theta(s) = \tau$. When the generalized time transformation function $\theta(s)$ is obtained from the differential equation i) and satisfies ii) and iii), then: a) $\theta(s)$ is continuous differentiable and strictly increasing on $s \in [0, \infty)$. b) Let $h(s) \triangleq \frac{d(\theta(s))}{ds}$. Then, $h(s)$ is bounded and $\lim_{s\to\infty} h(s) = 0$.*

The proof of this lemma is omitted due to page limitation and will be reported elsewhere.

In order to numerically elucidate Lemma 1, we now present several generalized finite-time gain functions $\alpha(t)$. Specifically, a common choice for the finite-time gain function is $\alpha(t) = 1/(\tau - t)$ (see, for example, [15], [16]) with its corresponding time transformation function $\theta(s) = \tau(1 - e^{-s})$. In addition, we introduce a family of finite-time gain functions defined by $\alpha(t) \triangleq \frac{1}{(\tau - t)(mt + a)}$, where $m \in \mathbb{R}_+$ and $a \in \mathbb{R}_+$. Note that $\alpha(t)$ satisfies the conditions of Assumption 1. In addition, $\theta(s) = \frac{\tau a (e^{(a+m\tau)s} - 1)}{a e^{(a+m\tau)s} + m\tau}$ is obtained by solving the differential equation *i)* of Lemma 1 and satisfies both *ii)* and *iii)*. Figures 1 and 2 show, respectively, plots of $\alpha(t)$ and $\theta(s)$ as m is increasing while a is fixed. In these figures, the arrows point in the increasing direction of m. Moreover, the common finite-time gain function $\alpha(t) = 1/(\tau - t)$ and its $\theta(s) = \tau(1 - e^{-s})$

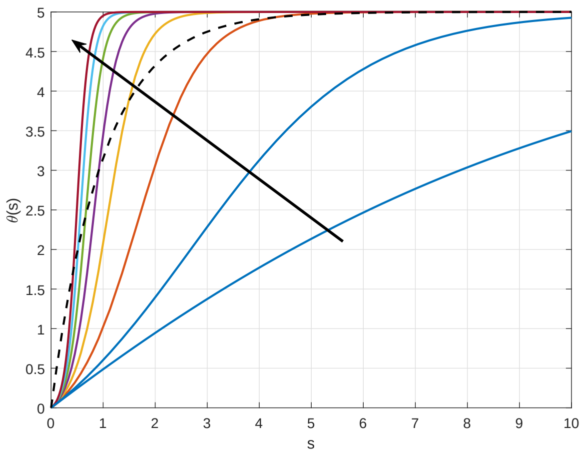

Fig. 2. Plots of the family of $\theta(s)$ with $a = 0.1$, $m \in [0.01, 2]$, and $\tau = 5$ over the stretched time domain, where the arrow pointing in the increasing direction of m. The dashed line represents $\theta(s) = \tau(1 - e^{-s})$ for comparison purpose.

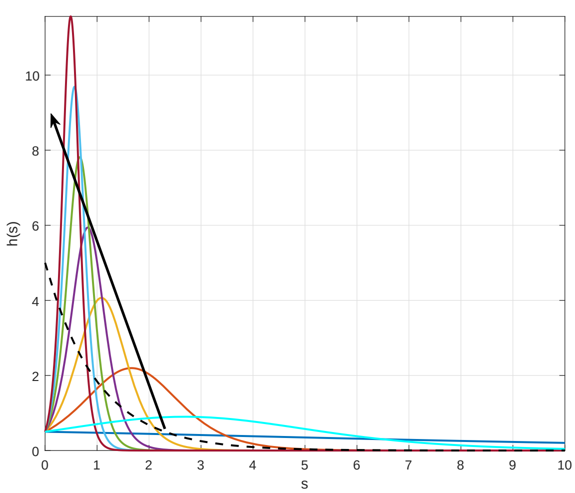

Fig. 3. Plots of the family of $h(s)$ with $a = 0.1$, $m \in [0.01, 2]$, and $\tau = 5$ over the stretched time domain, where the arrow pointing in the increasing direction of m. The dashed line represents $h(s) = \tau e^{-s}$ for comparison purpose.

are also plotted (dashed lines). Furthermore, plots of $h(s)$ are shown in Figure 3 to illustrate the result b) of Lemma 1.

Building on the result of Lemma 1, the convergence of the solution of the perturbed dynamical system given by (1) to zero over the prescribed regular time interval $[0, \tau)$ is stated in the following Theorem.

Theorem 1. *Consider the perturbed dynamical system in (1). If the generalized finite-time gain function $\alpha(t)$ subjects to Assumption 1 and there is a corresponding generalized time transformation function $\theta(s)$ (see Lemma 1), then* $\lim_{t\to\tau} x(t) = 0$.

Due to page limitation, once again the proof of this result will be reported elsewhere. For interested readers, it follows by first defining $\bar{x}(s) \triangleq x(\theta(s)) \equiv x(t)$ and utilizing time transformation method to rewrite (1) in the stretched infinite-time interval $s \in [0, \infty)$ as

$$\bar{x}'(s) \triangleq \frac{d\bar{x}(s)}{ds} = \frac{d\theta(s)}{ds}\frac{d\bar{x}(s)}{d\theta(s)} = f\big(\bar{x}(s)\big) + h(s)g\big(\theta(s), \bar{x}(s)\big),\ \bar{x}(0) = x_0, \tag{2}$$

where $h(s) \triangleq 1/\alpha(\theta(s))$. By Lemma 1, $h(s)g\big(\theta(s), \bar{x}(s)\big)$ is bounded and goes to 0 as $t \to \infty$. Furthermore, the origin of the nominal dynamical system $\dot{x}(t) = f\big(x(t)\big)$ of (1) is globally exponentially stable; hence, the result follows from Lemma 4.6 of [30]. We note that the above result also holds locally if the local conditions in Theorem 4.14 of [30] are satisfied.

At this point, it is natural to question the boundedness of $\dot{x}(t)$ over $t \in [0, \tau)$. In fact, $\dot{x}(t)$ depicted by the perturbed dynamical system given by (1) is bounded on $t \in [0, \tau)$, if the following conditions are satisfied:

i) $\frac{\dot{\alpha}(t)}{\alpha^2(t)}$ is bounded on $t \in [0, \tau)$, and $\lim_{t\to\tau} \frac{\dot{\alpha}(t)}{\alpha^2(t)} = \kappa < \infty$.

ii) $\bar{r}'(s) = \left(\frac{df\left(\bar{x}(s)\right)}{d\bar{x}} + \frac{d\alpha(\theta(s))}{d\theta(s)}h^2(s)\mathrm{I}_n\right)\bar{r}(s)$ is globally exponentially stable, where $r(t) = r\big(\theta(s)\big)$ and $\bar{r}(s) \triangleq r\big(\theta(s)\big)$.

This can be shown by proving that $r(t) \triangleq \alpha(t)f\big(x(t)\big)$ is bounded over $t \in [0, \tau)$. Particularly, based on (1), one can obtain the dynamics for $\dot{r}(t)$, transform the resulting dynamics to the stretch time interval $s \in [0, \infty)$, and utilize the same discussion and analysis as in Theorem 1 to conclude the boundedness of $r(t)$. Once again, details will be reported elsewhere due to page limitation.

Theorem 1 shows that under the effect of the generalized finite-time gain function $\alpha(t)$, the origin of the dynamical system (1) is robust with respect to bounded perturbations. Thus, for time-critical applications, if a control algorithm for the dynamical system is designed such that the error dynamics can be written in the form given by (1), then the stability is guaranteed. To summarize, a three-step procedure below is proposed when designing a control algorithm for applications that require finite-time convergence: *i)* Form a baseline control algorithm to exponentially meet objectives of a considered application over $t \in [0, \infty)$, the standard time interval. *ii)* Find a generalized finite-time gain function $\alpha(t)$ satisfying Assumption 1 and its corresponding generalized time transformation function $\theta(s)$ (see Lemma 1). *iii)* Obtain the finite-time control algorithm by multiplying the design control baseline algorithm with the generalized finite-time gain function $\alpha(t)$.

IV. Numerical Example

In this section, a numerical example is introduced to highlight the performance of the finite-time control algorithm under different generalized finite-time gain functions. For this purpose, we consider a multiagent system with 5 agents exchanging information under a path graph, where the first agent is the leader and the rest are followers. The following baseline control algorithm (see, for example, [31]) is considered

$$\dot{x}_i(t) = -\gamma \left(\sum_{i\sim j} \big(x_i(t) - x_j(t)\big) + k_i\big(x_i(t) - c(t)\big) \right), \quad x_i(0) = x_{i0}. \tag{3}$$

Here, $x_i(t) \in \mathbb{R}$ stands for the state of agent i, $i = 1, \ldots, 5$, $c(t) \in \mathbb{R}$ stands for the bounded command with bounded time rate of change, $\gamma \in \mathbb{R}_+$ stands for a scalar gain, $k_1 = 1$ for the first leader agent, and $k_i = 0$ for the follower agents.

By defining the error $\tilde{x}_i(t) \triangleq x_i(t) - c(t)$ and taking its time derivative, we obtain

$$\dot{\tilde{x}}_i(t) = -\gamma \left(\sum_{i\sim j} \big(\tilde{x}_i(t) - \tilde{x}_j(t)\big) + k_i\tilde{x}_i(t) \right) - \dot{c}(t), \quad \tilde{x}_i(0) = \tilde{x}_{i0}. \tag{4}$$

The above equation can be rewritten compactly as

$$\dot{\tilde{x}}(t) = -\gamma \left(\mathcal{L}(\mathcal{G}) + K\right) \tilde{x}(t) - \mathbf{1}_5\dot{c}(t), \quad \tilde{x}(0) = \tilde{x}_0, \tag{5}$$

where $\tilde{x}(t) = [\tilde{x}_1(t), \ldots, \tilde{x}_5(t)]^{\mathrm{T}}$, $\mathcal{L}(\mathcal{G}) \in \mathbb{R}^{5\times 5}$ is the Laplacian matrix of the path graph with 5 agents and $K = diag([1, 0, 0, 0, 0])$. Here, $-\left(\mathcal{L}(\mathcal{G}) + K\right)$ is a Hurwitz matrix (see, for example, Lemma 3.3 of [31]).

Multiplying the generalized finite-time gain function $\alpha(t)$ with the baseline algorithm (3) yields the following time-varying distributed control algorithm

$$\dot{x}_i(t) = u_i(t), \quad x_i(0) = x_{i0}, \tag{6}$$

$$u_i(t) = -\gamma\alpha(t) \left(\sum_{i\sim j} \big(x_i(t) - x_j(t)\big) + k_i\big(x_i(t) - c(t)\big) \right). \tag{7}$$

Then, the compact form of the resulting error dynamics can be given as

$$\dot{\tilde{x}}(t) = -\gamma\alpha(t) \left(\mathcal{L}(\mathcal{G}) + K\right) \tilde{x}(t) - \mathbf{1}_5\dot{c}(t), \quad \tilde{x}(0) = \tilde{x}_0, \tag{8}$$

which is identical to the dynamics given by (1) with $f\big(x(t)\big) = -\gamma \left(\mathcal{L}(\mathcal{G}) + K\right) \tilde{x}(t)$ and $g\big(t, x(t)\big) = -\mathbf{1}_5\dot{c}(t)$.

As discussed in the second paragraph after Lemma 1, $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$ is a valid generalized finite-time gain function. In what follows, we choose $\tau = 5$ seconds and consider three cases: $a = 1$ and $m = 0$ are chosen for the first case (see, for example, [15]), $a = 0.5$ and $m = 0.005$ are chosen for the second case, and $a = 0.1$ and $m = 0.085$ are chosen for the third case. Along the lines of the discussion in the second paragraph below Theorem 1, in order to guarantee the boundedness of $\dot{x}(t)$, we need to choose parameters to satisfy conditions *i)* and *ii)*. For the first case, since $\dot{\alpha}(t)/\alpha^2(t) = 1$ on $t \in [0, \tau)$; hence, $\kappa = 1$ and the condition *i)* is readily satisfied. In addition, let $M \triangleq -\left(\mathcal{L}(\mathcal{G}) + K\right)$, then in order to satisfy the condition *ii)*, we need to choose $\gamma > -\kappa/\lambda_{\max}(M) = 12.3435$ (see, for example, Theorem 2 of [15]). Thus, $\gamma = 13$ is an appropriate choice for this first case. For the last two cases, since the upper bound

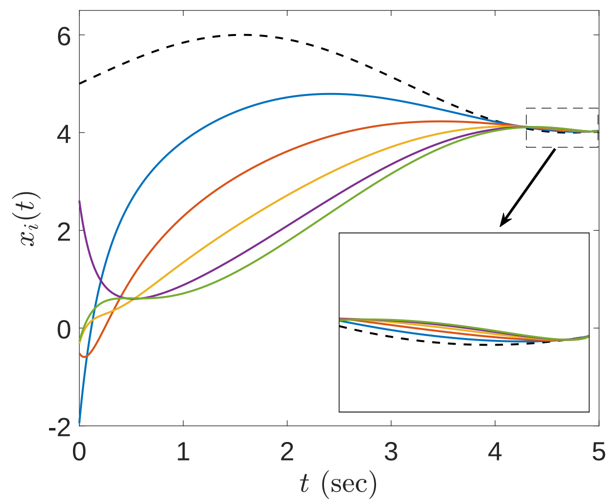

Fig. 4. Evolution of the states of agents under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 1$, $m = 0$, $\tau = 5$ seconds, and $\gamma = 13$, where the dashed line shows the tracking command.

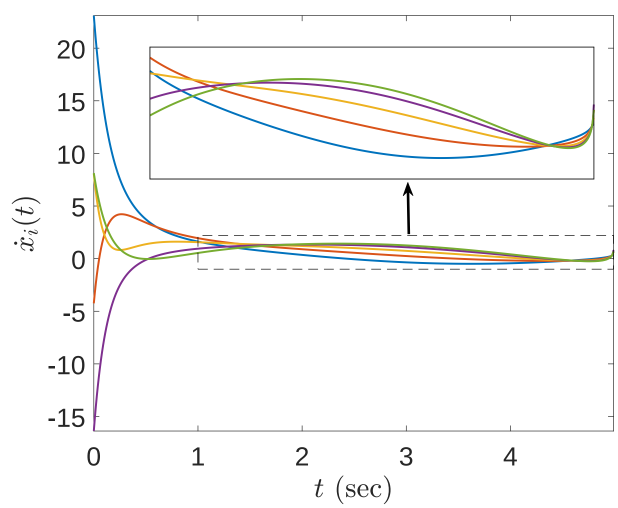

Fig. 5. Evolution of the time derivative of agents' states $\dot{x}_i(t)$ under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 1$, $m = 0$, $\tau = 5$ seconds, and $\gamma = 13$.

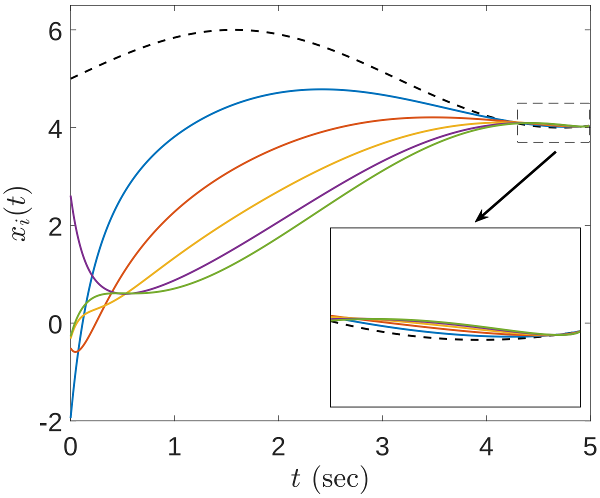

Fig. 6. Evolution of the states of agents under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 0.5$, $m = 0.005$, $\tau = 5$ seconds, and $\gamma = 6.5$, where the dashed line shows the tracking command.

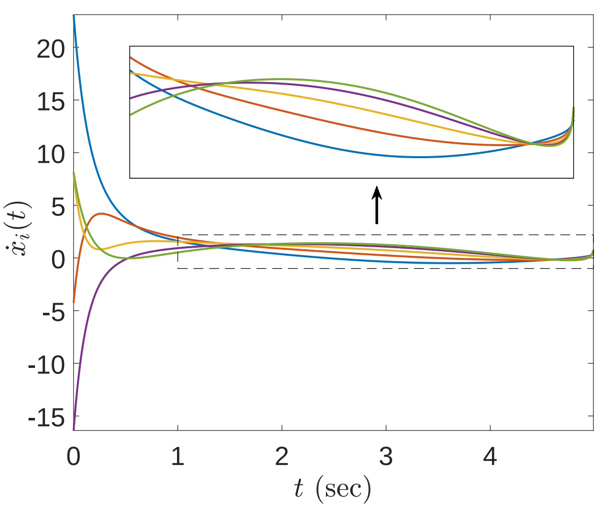

Fig. 7. Evolution of the time derivative of agents' states $\dot{x}_i(t)$ under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 0.5$, $m = 0.005$, $\tau = 5$ seconds, and $\gamma = 6.5$.

of $\dot{\alpha}(t)/\alpha^2(t)$ on $t \in [0, \tau)$ is $\kappa = m\tau + a = 0.525$ for these two cases, the condition *i)* is satisfied. Note that $-\kappa/\lambda_{\max}(M) = 6.4804$; then by choosing $\gamma = 6.5$ for both cases, the condition *ii)* is satisfied. Furthermore, the tracking command $c(t) = 5 + sin(t)$ is chosen for the algorithm given by (7), and random initial conditions for agents are utilized for both cases.

Figures 4 and 5 respectively show the evolution of agents' states and their time derivatives under the control algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 1$, $m = 0$, $\tau = 5$ seconds, and $\gamma = 13$. Figures 6 and 7 respectively show the evolution of agents' states and their time derivatives under the control algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 0.5$, $m = 0.005$, $\tau = 5$ seconds, and $\gamma = 6.5$. Similarly, Figures 8 and 9 respectively show the evolution of agents' states and their time derivatives under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 0.1$, $m = 0.085$, $\tau = 5$ seconds, and $\gamma = 6.5$. As expected from the result of Theorem 1, the states of all agents approach to the command $c(t)$ as $t \to \tau = 5$ seconds. Note that the performances of the first two cases look almost identical, yet the value of γ of the first case is twice as big as the second case. It can be seen that when $t = 0$, $\alpha(0) = 1/(\tau a)$; hence, a affects the initial value of $\alpha(t)$. Therefore, the value of $\gamma\alpha(0)$ of

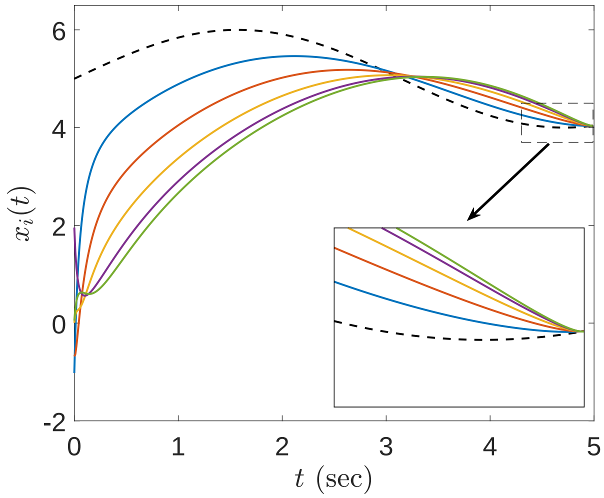

Fig. 8. Evolution of the states of agents under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 0.1$, $m = 0.085$, $\tau = 5$ seconds, and $\gamma = 6.5$, where the dashed line shows the tracking command.

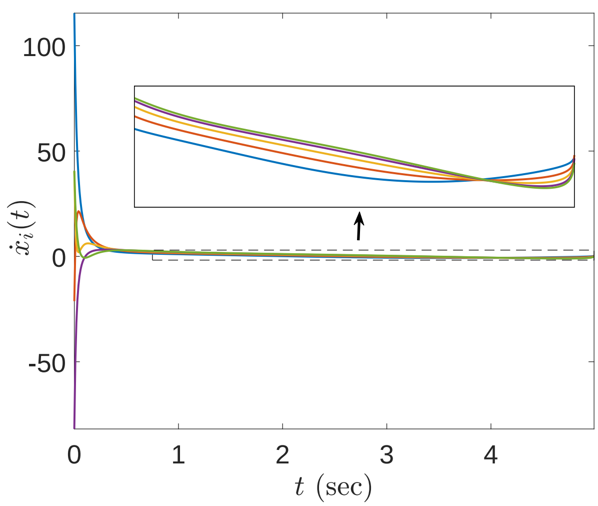

Fig. 9. Evolution of the time derivative of agents' states $\dot{x}_i(t)$ under algorithm (7) with $\alpha(t) \triangleq \frac{1}{(\tau-t)(mt+a)}$, $a = 0.1$, $m = 0.085$, $\tau = 5$ seconds, and $\gamma = 6.5$.

the first two cases are the same. Furthermore, $m = 0.005$ in the second case plays the role of damping $\alpha(t)$ when t get big, but the choice of $\tau = 5$ is not sufficiently big to see the difference in behavior of the second case compare to the first case. Since the third case has a smaller value for a, the initial value of $\gamma\alpha(t)$ in the third case is larger than in the first two cases. This is depicted by the higher initial values of $\dot{x}(t)$ in Figure 9 compared to the ones in Figures 5 and 7. Furthermore, as expected from the discussion below Theorem 1, Figures 5, 7 and 9 show that $\dot{x}_i(t)$ remains bounded over $t \in [0, \tau)$. In general, different transient behaviors of the resulting networked multiagent system are observed when different values of a and m are chosen. Specifically, the states of agents in the third case approach the tracking command faster than the ones of the first two cases as shown in Figures 4, 6 and 8.

V. Conclusion

The paper contributes to recent studies on finite-time control based on the time transformation method. Specifically, a new class of scalar, time-varying gain functions (generalized finite-time gain functions) was investigated, and its relationship with the generalized time transformation function was established. We showed how these functions can convert an original baseline control algorithm into a time-varying one in order to allow its execution over a prescribed time interval. Furthermore, conditions to guarantee the stability as well as the boundedness of the state's time derivative are shown. The efficacy of the proposed finite-time control methodology was also demonstrated through an illustrative numerical example.

References

[1] S. P. Bhat and D. S. Bernstein, "Continuous finite-time stabilization of the translational and rotational double integrators," *IEEE Transactions on automatic control*, vol. 43, no. 5, pp. 678–682, 1998.

[2] ——, "Finite-time stability of continuous autonomous systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 3, pp. 751–766, 2000.

[3] J. Cortés, "Finite-time convergent gradient flows with applications to network consensus," *Automatica*, vol. 42, no. 11, pp. 1993–2000, 2006.

[4] Q. Hui, W. M. Haddad, and S. P. Bhat, "Finite-time semistability and consensus for nonlinear dynamical networks," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1887–1900, 2008.

[5] F. Xiao, L. Wang, J. Chen, and Y. Gao, "Finite-time formation control for multi-agent systems," *Automatica*, vol. 45, no. 11, pp. 2605–2611, 2009.

[6] F. Jiang and L. Wang, "Finite-time information consensus for multi-agent systems with fixed and switching topologies," *Physica D: Nonlinear Phenomena*, vol. 238, no. 16, pp. 1550–1560, 2009.

[7] L. Wang and F. Xiao, "Finite-time consensus problems for networks of dynamic agents," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 950–955, 2010.

[8] M. Basin, Y. Shtessel, and F. Aldukali, "Continuous finite-and fixed-time high-order regulators," *Journal of the Franklin Institute*, vol. 353, no. 18, pp. 5001–5012, 2016.

[9] H. Hong, W. Yu, G. Wen, and X. Yu, "Distributed robust fixed-time consensus for nonlinear and disturbed multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1464–1473, 2017.

[10] B. Tian, Z. Zuo, and H. Wang, "Leader–follower fixed-time consensus of multi-agent systems with high-order integrator dynamics," *International Journal of Control*, vol. 90, no. 7, pp. 1420–1427, 2017.

[11] W. Lu, X. Liu, and T. Chen, "A note on finite-time and fixed-time stability," *Neural Networks*, vol. 81, pp. 11–15, 2016.

[12] H. B. Oza, Y. V. Orlov, and S. K. Spurgeon, "Robust finite time stability and stabilisation: A survey of continuous and discontinuous paradigms," in *2014 International Workshop on Variable Structure Systems*. IEEE, 2014, pp. 1–7.

[13] A. Polyakov, "Nonlinear feedback design for fixed-time stabilization of linear control systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 8, pp. 2106–2110, 2012.

[14] H. Ríos and A. R. Teel, "A hybrid observer for fixed-time state estimation of linear systems," in *2016 Conference on Decision and Control*. IEEE, 2016, pp. 5408–5413.

[15] T. Yucelen, Z. Kan, and E. Pasiliao, "Finite-time cooperative engagement," *IEEE Transactions on Automatic Control*, 2018.

[16] Z. Kan, T. Yucelen, E. Doucette, and E. Pasiliao, "A finite-time consensus framework over time-varying graph topologies with temporal constraints," *Journal of Dynamic Systems, Measurement, and Control*, vol. 139, no. 7, p. 071012, 2017.

[17] E. Arabi, T. Yucelen, and J. R. Singler, "Robustness of finite-time distributed control algorithm with time transformation," in *American Control Conference*, 2019.

[18] J. D. Sánchez-Torres, E. N. Sanchez, and A. G. Loukianov, "Predefined-time stability of dynamical systems with sliding modes," in *2015 American Control Conference*. IEEE, 2015, pp. 5842–5846.

[19] E. Jiménez-Rodríguez, J. D. Sánchez-Torres, D. Gómez-Gutiérrez, and A. G. Loukianov, "Predefined-time tracking of a class of mechanical systems," in *2016 International Conference on Electrical Engineering, Computing Science and Automatic Control*. IEEE, 2016, pp. 1–5.

[20] E. Jiménez-Rodríguez, J. D. Sánchez-Torres, and A. G. Loukianov, "On optimal predefined-time stabilization," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 17, pp. 3620–3642, 2017.

[21] C. Yong, X. Guangming, and L. Huiyang, "Reaching consensus at a preset time: Single-integrator dynamics case," in *2012 Chinese Control Conference*. IEEE, 2012, pp. 6220–6225.

[22] ——, "Reaching consensus at a preset time: Double-integrator dynamics case," in *2012 Chinese Control Conference*. IEEE, 2012, pp. 6309–6314.

[23] C. Wang, G. Xie, and M. Cao, "Forming circle formations of anonymous mobile agents with order preservation," *IEEE Transactions on Automatic Control*, vol. 58, no. 12, pp. 3248–3254, 2013.

[24] J. Wen, C. Wang, W. Luo, and G. Xie, "Finite-time consensus of networked multiagent systems with time-varying linear control protocols," *Mathematical Problems in Engineering*, vol. 2016, 2016.

[25] N. Harl and S. Balakrishnan, "Impact time and angle guidance with sliding mode control," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 6, pp. 1436–1449, 2012.

[26] H. Wang, C. Wang, and G. Xie, "Finite-time containment control of multi-agent systems with static or dynamic leaders," *Neurocomputing*, vol. 226, pp. 1–6, 2017.

[27] Y. Wang, Y. Song, D. J. Hill, and M. Krstic, "Prescribed-time consensus and containment control of networked multiagent systems," *IEEE Transactions on Cybernetics*, 2018.

[28] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[29] C. D. Godsil, G. Royle, and C. Godsil, *Algebraic graph theory*. Springer New York, 2001, vol. 207.

[30] H. K. Khalil, *Nonlinear systems*. Prentice Hall, 2002, vol. 3.

[31] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative control of multi-agent systems: Optimal and adaptive design approaches*. Springer Science & Business Media, 2014.