

Research Article

Predefined-Time Consensus of Nonlinear First-Order Systems Using a Time Base Generator

J. Armando Colunga ¹, Carlos R. Vázquez ²,
Héctor M. Becerra ¹, and David Gómez-Gutiérrez ^{2,3}

¹Centro de Investigación en Matemáticas (CIMAT), A.C., Computer Science Department, GTO, Mexico

²Tecnologico de Monterrey, Escuela de Ingeniería y Ciencias, Zapopan, Mexico

³Intel Tecnología de México, Multi-Agent Autonomous Systems Lab, Intel Labs, Zapopan, Mexico

Correspondence should be addressed to Héctor M. Becerra; hector.becerra@cimat.mx

Received 16 June 2018; Revised 11 September 2018; Accepted 17 September 2018; Published 9 October 2018

Academic Editor: Ju H. Park

Copyright © 2018 J. Armando Colunga et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a couple of consensus algorithms for multiagent systems in which agents have first-order nonlinear dynamics, reaching the consensus state at a predefined time independently of the initial conditions. The proposed consensus protocols are based on the so-called time base generators (TBGs), which are time-dependent functions used to build time-varying control laws. Predefined-time convergence to the consensus is proved for connected undirected communication topologies and directed topologies having a spanning tree. Furthermore, one of the proposed protocols is based on the super-twisting controller, providing robustness against disturbances while maintaining the predefined-time convergence property. The performance of the proposed methods is illustrated in simulations, and it is compared with finite-time, fixed-time, and predefined-time consensus protocols. It is shown that the proposed TBG protocols represent an advantage not only in the possibility to define a settling time but also in providing smoother and smaller control actions than existing finite-time, fixed-time, and predefined-time consensus.

1. Introduction

Recent years have seen an increasing interest in algorithms allowing a group of systems to reach a common value for its internal state through local interaction. This problem has been addressed, from different viewpoints, in the consensus of multiagent systems (MASs) [1, 2] and in the synchronization of complex dynamical networks [3–5]. On the one hand, consensus protocols have been applied to flocking [6], formation control [7, 8], and distributed resource allocation [9, 10]. On the other hand, the results on synchronization of complex dynamical networks have been applied to neuroscience [3], power-grids [3], and the chaotic synchronization for secure communication in swarms [11].

Several works have been published proposing consensus and synchronization algorithms for different types of systems, considering static [12–14] and dynamic networks [4, 5]. Regarding first-order agents, the standard protocol (the input

of an agent is a linear combination of the errors between the agent's state and those of its neighbors) achieves consensus if the graph topology is strongly connected [15, 16]. This algorithm achieves consensus also for dynamic topologies, switching among connected graphs [15, 17]. For directed graphs topologies, a common requirement is that the graph contains a spanning-tree (e.g., [18]). All these algorithms are based on linear protocols and achieve consensus in an asymptotic fashion, where convergence rate is a function of the algebraic connectivity of the graph.

With the aim of developing consensus protocols satisfying real-time constraints, finite-time consensus has received a great deal of attention. The main focus has been in defining nonlinear protocols evaluated either on each of the neighbors' errors or on the sum of the neighbors' errors. For finite-time convergence, binary protocols based on the sign(\bullet) function have been proposed to achieve consensus to the average value [19, 20], the average-min-max value [21], the

median value [22], and the maximum or minimum value [23] of the agent's initial conditions. Continuous finite-time protocols, based on the $|\cdot|^\alpha \text{sign}(\cdot)$ function, with $0 < \alpha < 1$, have been introduced in [12, 24]. Some algorithms have been extended to achieve finite-time consensus for strongly connected dynamic topologies [20, 25]. Fixed-time protocols have also been proposed in [26, 27]. In these, there exists a bound for the convergence time that is independent of the initial conditions [28, 29]. The protocols are mainly based on functions of the form $|\cdot|^p \text{sign}(\cdot) + |\cdot|^q \text{sign}(\cdot)$, with $0 < q < 1 < p$.

The methods mentioned above cannot be used in applications where real-time constraints have to be accomplished, since in both finite-time and fixed-time consensus, the relationship between the protocol parameters and the convergence time is not straightforward. Thus it is difficult to define the convergence time bound as a function of the protocol parameters. Moreover, the convergence time bound is often overestimated.

To address the consensus design problem with real-time constraints, predefined-time convergence has to be investigated. In this, the time at which consensus is achieved is predefined a priori as a parameter of the consensus protocol, being independent of the initial conditions. Few works have addressed the predefined-time consensus problem. In [30, 31], linear protocols that use time-varying control gains were proposed for reaching network consensus at a preset time. By using a motion planning strategy, predefined-time consensus protocols based on a time-varying sampling sequence convergent to a specified settling time are proposed in [32, 33]. Preliminary versions of a couple of predefined-time protocols were proposed in [34], taking advantage of time base generators (TBGs), which are parametric time signals that converge to zero in a specified time [35], and can be tracked using feedback controllers.

In this work, a couple of consensus algorithms based on TBGs are proposed as an extension to [34]. One algorithm is based on a linear-feedback, and the other is based on the super-twisting controller for the tracking of the TBG signal. The proposed approach can be applied to systems in which agents belong to the class of first-order controllable linear systems and nonlinear systems that can be linearized by state feedback [36]. Comparisons between the proposed protocols and finite-time, fixed-time, and predefined-time protocols are provided through simulations, showing the advantages of the proposed approach. In contrast with works on predefined-time consensus [30–32], the contribution of this paper is threefold: First, the proposed controllers yield smoother auxiliary control signals (without considering the linearization terms) with smaller magnitudes. Second, there are no parameters in the proposed consensus protocols depending on the algebraic connectivity of the graph considered. Third, a super-twisting controller is presented to deal with perturbed dynamics in the network's agents. With respect to our preliminary conference version [34], two main extensions are reported herein: (1) the new linear-feedback protocol drives the agents towards the initial conditions average (for undirected graphs) or the weighted sum of the initial conditions (for directed graphs), whereas in our previous work the consensus value was artificially specified;

(2) convergence to the consensus value in predefined time is demonstrated even if uncertainty in the knowledge of the initial state is introduced in the controller, at difference of our previous results, where it was assumed that the initial state was perfectly known.

It is known that second-order systems can represent a broader class of real systems than first-order systems. However, the last class of systems has also broad applicability, for instance, in robotics for kinematic control of holonomic robots. We are very interested in generalizing our results even for systems of a higher order than two, taking into account existing results like the ones in [37, 38].

The rest of this paper is organized as follows. Section 2 introduces some background in algebraic graph theory, defines the class of systems for which the proposed method is applicable, and formulates the problem to be solved in this paper. Section 3 presents the proposed consensus protocols: a linear-feedback and a super-twisting controller for the tracking of the TBG signal. Section 4 presents simulation results and a comparison of the proposed controllers with existing approaches. Section 5 provides conclusions.

2. Preliminaries and Problem Definition

2.1. Algebraic Graph Theory. Agents' communication is represented by a graph $\mathcal{G} = (\mathbf{V}, \mathbf{E}, \mathbf{A})$, which consists of a set of vertices (nodes or agents in this work) $\mathbf{V} = \{1, \dots, N\}$, a set of edges $\mathbf{E} \subseteq \mathbf{V} \times \mathbf{V}$, and a weighted adjacency matrix $\mathbf{A} = [a_{ij}]$ with nonnegative adjacency elements a_{ij} , i.e., $a_{ij} > 0$ iff $(i, j) \in \mathbf{E}$. The set of neighbors of agent i is denoted by $\mathbf{N}_i = \{j \in \mathbf{V} : (j, i) \in \mathbf{E}\}$.

Definition 1 (see [39, 40]). A graph \mathcal{G} is called *directed* if there is an entry of \mathbf{A} that fulfills $a_{ij} > 0$; i.e., $(i, j) \in \mathbf{E}$ represents that agent j receives the information of the state value x_i of the agent i ; otherwise, $a_{ij} = 0$, i.e., $(j, i) \notin \mathbf{E}$. A graph \mathcal{G} is called *undirected* if the pairs of nodes are not ordered $a_{ij} = a_{ji} > 0$; otherwise, $a_{ij} = a_{ji} = 0$, i.e., $(i, j), (j, i) \notin \mathbf{E}$.

Definition 2 (see [39, 40]). A directed path (respectively, undirected path) from nodes v_j to v_i is a sequence of distinct edges of the form $(v_i, v_{i_1}), (v_{i_1}, v_{i_2}), \dots, (v_i, v_j)$ in a directed graph (respectively, undirected graph). A graph is strongly connected (respectively, connected) if there exists a directed path (respectively, undirected path) between any two distinct vertices v_i and v_j in \mathcal{G} .

Definition 3 (see [16, 18]). A directed graph \mathcal{G} has a directed spanning tree if there exists a node v_i (a root) such that all other nodes can be linked to v_i via a directed path.

If an undirected graph has a directed spanning tree, then the graph is connected. A strongly connected directed graph contains at least one directed spanning tree [16, 18].

Definition 4 (see [16]). Let \mathcal{G} be a graph on N vertices labelled as $1, \dots, N$. The *Laplacian matrix* of \mathcal{G} is defined as the $N \times N$ matrix $\mathbf{L} = [l_{ij}]$, where

$$l_{ij} = \begin{cases} -a_{ij}, & \text{if } i \neq j, \\ \sum_{k=1, k \neq i}^N a_{ik}, & \text{if } i = j. \end{cases} \quad (1)$$

In this work, we will propose and analyze consensus protocols for two kinds of networks: connected undirected graphs and directed graphs with a directed spanning tree [2, 16, 18].

Lemma 5 ([2, 16, 18]). \mathbf{L} has at most one zero eigenvalue $\lambda_1(\mathbf{L})$ with $\mathbf{1}_N$ as the corresponding right eigenvector, where $\mathbf{1}_N$ denotes a column vector with all entries equal to one, i.e., $\lambda_1(\mathbf{L}) = 0$, $\text{Re}(\lambda_i(\mathbf{L})) > 0$, $\forall i = \{2, \dots, N\}$ and $\mathbf{L}\mathbf{1}_N = \mathbf{0}_N$. Furthermore, if the graph is undirected and connected, \mathbf{L} has a left eigenvector $\boldsymbol{\gamma}$ that satisfies $\boldsymbol{\gamma}^T \mathbf{L} = \mathbf{0}_N^T$ and $\boldsymbol{\gamma} = \mathbf{1}_N$, where $\mathbf{0}_N$ represents a column vector of zeros. If the graph is directed and has a spanning tree then $\boldsymbol{\gamma}^T \mathbf{L} = \mathbf{0}_N^T$ and $\boldsymbol{\gamma}^T \mathbf{1}_N = 1$.

Finally, let us introduce a matrix transformation that will be useful later on. Let $\mathbf{J} \in \mathbb{C}^{N \times N}$ be the Jordan form associated with \mathbf{L} . Then, there exists a nonsingular matrix $\mathbf{S} \in \mathbb{C}^{N \times N}$ such that

$$\mathbf{S}^{-1} \mathbf{L} \mathbf{S} = \mathbf{J} = \begin{bmatrix} 0 & \mathbf{0}_{N-1}^T \\ \mathbf{0}_{N-1} & \hat{\mathbf{J}} \end{bmatrix} \quad (2)$$

where $\hat{\mathbf{J}} \in \mathbb{C}^{(N-1) \times (N-1)}$ is the Jordan matrix associated with the nonzero eigenvalues of \mathbf{L} [39, 41] and $\mathbf{0}_{N-1}$ represents a column vector of zeros of size $N - 1$.

2.2. Problem Definition. In this paper, we will consider a set of N agents connected through a network, which are modeled as first-order nonlinear systems as follows

$$\dot{x}_i(t) = f_i(x_i) + g_i(x_i)u_i(t) + \rho_i(t), \quad i \in \{1, \dots, N\} \quad (3)$$

Where, for the agent i , $x_i(t) \in \mathbb{R}$ is the system's state, $u_i(t) \in \mathbb{R}$ is the control input, $f_i(x_i)$ and $g_i(x_i)$ are smooth nonlinear functions, and $\rho_i(t) \in \mathbb{R}$ is a bounded disturbance. In order to represent the multiagent system (MAS) dynamics, denote $\mathbf{x} = [x_1, \dots, x_N]^T$, $\mathbf{u} = [u_1, \dots, u_N]^T$, $\mathbf{f} = [f_1, \dots, f_N]^T$, $\mathbf{g} = [g_1, \dots, g_N]^T$, and $\boldsymbol{\rho} = [\rho_1, \dots, \rho_N]^T \in \mathbb{R}^N$. Then, the complete dynamics (3) can be written as

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}(t) + \boldsymbol{\rho}(t). \quad (4)$$

Let us propose a linearizing control law of the form $u_i = (-f_i(x_i) + v_i)/g_i(x_i)$ for each agent (3), where v_i is an auxiliary control input and it is assumed that $g_i(x_i) \neq 0$, i.e., the relative degree is well defined [36] and the agent is controllable. The closed-loop dynamics for the i -th agent are now linear and given by

$$\dot{x}_i(t) = v_i(t) + \rho_i(t), \quad i \in \{1, \dots, N\}. \quad (5)$$

The complete closed-loop dynamics for all the agents can be written as

$$\dot{\mathbf{x}}(t) = \mathbf{v}(t) + \boldsymbol{\rho}(t) \quad (6)$$

where $\mathbf{v} = [v_1, \dots, v_N]^T \in \mathbb{R}^N$.

In a MAS, the consensus error of agent i with respect to its neighbors is defined as [2]

$$e_i(t) = \sum_{j \in \mathcal{N}_i} a_{ij} (x_j(t) - x_i(t)), \quad i \in \{1, \dots, N\}. \quad (7)$$

The consensus error for the complete MAS can be expressed in a compact form as [2]

$$\mathbf{e}(t) = [e_1(t), \dots, e_N(t)]^T = -\mathbf{L}\mathbf{x}(t). \quad (8)$$

Problem statement: given a MAS with agents modeled as first-order nonlinear systems (3) and with an associated graph \mathcal{G} , design a consensus protocol for each agent in the form $u_i = \gamma(e_i, x_i, t)$, such that the state of all the agents reaches a consensus value x^* in a predefined time t_f from any initial state $\mathbf{x}(0)$, i.e., $x_i(t) \rightarrow x^*$, $\forall i = \{1, \dots, N\}$, as $t \rightarrow t_f$. Therefore, the consensus error (8) converges to zero at time t_f .

Remark 6. The consensus problem in MASs is closely related to the synchronization problem in complex dynamical networks, but they have been addressed from different viewpoints [42]. In fact, the consensus problem can be posed as a complete-synchronization problem. In the former, the analysis is often based on the matrix Laplacian approach [2] and the agents have simple dynamics such as single integrator dynamics, a chain of integrators, or linear systems. The recent focus has been on defining finite-time, fixed-time, and predefined-time protocols for fixed and dynamic networks. In the synchronization problem in complex networks the analysis is often based in the Master stability function formalism [3, 42] and addresses interconnected systems with complex dynamics, for instance, chaotic attractors. Although recently some works have addressed synchronization in dynamic networks, e.g., Markov jump topologies [4, 5], most of the work has been focused on static topologies. For further details on the synchronization problem, we refer the reader to [3].

2.3. Time Base Generator (TBG). Time base generators are parametric functions of time, particularly designed to stabilize a system in such a way that its state describes a convenient transient profile. TBGs have been previously used to achieve predefined-time convergence of first and higher order dynamics for single systems in [35]. For the case of single first-order systems, TBGs have been used for the control of robots at kinematic level [43].

A *time base generator (TBG)* is a continuous and differentiable time-dependent polynomial function $h(t)$ described as [35]

$$h(t) = \begin{cases} \boldsymbol{\tau}(t) \cdot \mathcal{E} & \text{if } t \in [0, t_f] \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

where $\boldsymbol{\tau}(t) = [t^r \ t^{r-1} \ \dots \ t \ 1]$ is the time basis vector and \mathcal{E} is a vector of coefficients of proper dimensions. For first-order systems, it is required to compute a TBG fulfilling the following conditions at initial time and final time t_f ,

$$h(t) = \begin{cases} 1, & \text{if } t = 0 \\ 0, & \text{if } t \geq t_f \end{cases} \quad (10)$$

$$\dot{h}(t) = 0, \quad \text{if } t = 0 \text{ or } t \geq t_f.$$

In [35], a couple of procedures for calculating the coefficients \mathcal{C} , fulfilling the required constraints, have been introduced. In particular, when a high degree polynomial is used, an optimization procedure can be applied to find the best coefficients.

3. Consensus Protocols Using the TBG

In this section, we present consensus protocols that take advantage of the TBG to enforce convergence of a MAS in predefined time. In the first two subsections, we present results for unperturbed systems, i.e., for $\rho = \mathbf{0}_N$, while in Section 3.3 we consider perturbed systems.

3.1. Previous Results. In [34], a consensus protocol was proposed for the MAS (4) by introducing a time-varying control gain in order to drive the system to a neighborhood about a given consensus value x^* in a predefined time t_f , from any initial state $\mathbf{x}(0)$. Consequently, the consensus error (8) converges to a neighborhood about the origin at t_f . Such consensus protocol is similar in nature to those in [30, 31].

In the same preliminary work [34], we have proposed the following open-loop controller to enforce predefined-time convergence to a desired consensus value \bar{x} , provided that the MAS (4) is not perturbed and the initial state is known,

$$u_i(t) = \frac{1}{g_i(x_i)} (-f_i(x_i) + v_i), \quad i \in \{1, \dots, N\} \quad (11)$$

$$v_i = \dot{h}(t)(x_i(0) - \bar{x})$$

where $h(t)$ fulfills (10). In that work, it was demonstrated that consensus to \bar{x} is achieved in time t_f and the resulting consensus error trajectory in the interval $t \in [0, t_f]$ is given by

$$\widehat{\mathbf{e}}(t) = h(t) \mathbf{e}_0 = -h(t) \mathbf{L} \mathbf{x}_0 \quad (12)$$

where $\mathbf{e}(0) = \mathbf{e}_0 = [e_1(0), \dots, e_N(0)]^T$ is the consensus error vector (8) at the initial condition $\mathbf{x}(0) = \mathbf{x}_0 = [x_1(0), \dots, x_N(0)]^T$.

To provide closed-loop stability of the tracking error, a different protocol was proposed in [34] to address the predefined-time consensus problem as a trajectory tracking problem, where the reference trajectory depends on the TBG. Such protocol is described as follows,

$$u_i(t) = \frac{1}{g_i(x_i)} (-f_i(x_i) + v_i), \quad i \in \{1, \dots, N\} \quad (13)$$

$$v_i = \dot{h}(t)(x_i(0) - \bar{x}) + k_f(e_i(t) - h(t)e_i(0)).$$

This protocol guarantees stability during the transient behavior and also after the settling time, i.e., for any $t \geq t_f$. The controller for each agent in (13) consists of two terms: the first one is a feed-forward term that yields the evolution of each agent's error from the initial value $e_i(0)$ to the origin at time t_f , following a transient profile defined by the TBG as $\widehat{e}_i(t) = h(t)e_i(0)$; the second term is a feedback term that corrects any deviation of $e_i(t)$ from the transient profile $\widehat{e}_i(t)$, providing global stability of the error dynamics.

3.2. Consensus Based on the Error Feedback. In the conference paper [34], whose main results were summarized above, consensus protocols were proposed achieving predefined-time convergence to a consensus value \bar{x} , which must be specified in the control law. Moreover, it was assumed that the initial state is accurately known since this value is used in the protocols.

Nevertheless, in most of the protocols proposed in the literature, the consensus value is a quantity that inherently results from the initial states and the connectivity of the graph model [15]. The consensus value may represent important information to be recovered, as in a network of sensors. For this reason, in this paper, we propose a new consensus protocol that yields predefined-time convergence of the unperturbed MAS (4) without the need of specifying a consensus value. Moreover, we consider uncertainties in the knowledge of the initial conditions.

In the sequel, consider that the consensus protocol has an estimate of the initial state, denoted as $\widehat{\mathbf{x}}_0 = [\widehat{x}_1(0), \dots, \widehat{x}_N(0)]^T$, instead of the real initial state \mathbf{x}_0 . In this way,

$$\widehat{\mathbf{x}}_0 = \mathbf{x}_0 + \mathbf{d} \quad (14)$$

where $\mathbf{d} = [d_1, \dots, d_N]^T$ is a vector of uncertainty in the initial state. From (14), the uncertain initial consensus error can be computed as

$$\widehat{\mathbf{e}}_0 = [\widehat{e}_1(t), \dots, \widehat{e}_N(t)]^T = -\mathbf{L} \widehat{\mathbf{x}}_0. \quad (15)$$

The following theorem introduces one of the main results of this paper, a feedback-based consensus protocol able to achieve consensus without the need of specifying the consensus value and able to track the trajectories given by the TBG when there is uncertainty in the initial state $\widehat{\mathbf{x}}_0$ introduced in the controller.

Theorem 7. *Consider a MAS with a connected undirected communication topology or directed topology having a spanning tree modeled as in (4). Let $k_f \in \mathbb{R}^+$ be a constant state-feedback gain. Then, the time-varying feedback control law*

$$u_i(t) = \frac{1}{g_i(x_i)} (-f_i(x_i) + v_i), \quad i \in \{1, \dots, N\} \quad (16)$$

$$v_i = -\dot{h}(t)\widehat{e}_i(0) + k_f(e_i(t) - h(t)\widehat{e}_i(0)),$$

with $h(t)$ as in (10) and $\widehat{e}_i(0)$ computed from the initial state estimate $\widehat{\mathbf{x}}_0$, provides global asymptotic stability of the tracking error $\boldsymbol{\xi}(t) = \mathbf{e}(t) - h(t)\widehat{\mathbf{e}}_0$. Therefore, assuming $\rho(t) = \mathbf{0}_N$, predefined-time convergence of the MAS (4) to a consensus value x^* is achieved at t_f from any initial state \mathbf{x}_0 .

Proof. First, let us demonstrate the stability of the tracking error. The dynamics (3) with $\rho_i(t) = 0, \forall i = \{1, \dots, N\}$, under the control action (16), can be written in vector notation as

$$\dot{\mathbf{x}}(t) = -\dot{h}(t)\widehat{\mathbf{e}}_0 + k_f(\mathbf{e}(t) - h(t)\widehat{\mathbf{e}}_0). \quad (17)$$

Computing $\dot{e}(t)$ from (8) and using the linearized dynamics (17), we have

$$\begin{aligned}\dot{\xi}(t) &= \dot{e}(t) - \dot{h}(t)\widehat{e}_0 = -L\dot{x}(t) - \dot{h}(t)\widehat{e}_0 \\ &= -L\dot{x}(t) + \dot{h}(t)L\widehat{x}_0 \\ &= -\dot{h}(t)L^2\widehat{x}_0 - k_f L\xi(t) + \dot{h}(t)L\widehat{x}_0 \\ &= -k_f L\xi(t) + \dot{h}(t)L(\widehat{x}_0 - L\widehat{x}_0).\end{aligned}\quad (18)$$

By using $h(0) = 1$, the initial tracking error is computed as

$$\begin{aligned}\xi(0) &= e_0 - h(0)\widehat{e}_0 = -Lx_0 + L(x_0 + d) = Ld \\ &= SJS^{-1}d,\end{aligned}\quad (19)$$

with $L = SJS^{-1}$, where $J \in \mathbb{C}^{N \times N}$ is the Jordan form associated with L and $S \in \mathbb{C}^{N \times N}$ is a nonsingular matrix [39].

Let $\zeta(t) = S^{-1}\xi(t)$, then $\zeta(0) = JS^{-1}d$, and according to (2), it follows that the first entry of $\zeta(0)$ is null, i.e., $\zeta_1(0) = 0$. Now, taking the time derivative of $\zeta(t)$ and using (18), we have

$$\begin{aligned}\dot{\zeta}(t) &\triangleq [\dot{\zeta}_1(t), \dots, \dot{\zeta}_N(t)]^T = S^{-1}\dot{\xi}(t) \\ &= S^{-1}(-k_f L\xi(t) + \dot{h}(t)L(\widehat{x}_0 - L\widehat{x}_0)) \\ &= -k_f S^{-1}(SJS^{-1})S\zeta(t) \\ &\quad + \dot{h}(t)S^{-1}(SJS^{-1})(\widehat{x}_0 - (SJS^{-1})\widehat{x}_0) \\ &= -k_f J\zeta(t) + \dot{h}(t)J(S^{-1} - JS^{-1})\widehat{x}_0.\end{aligned}\quad (20)$$

Let $\phi(t) = [\zeta_2(t), \dots, \zeta_N(t)]^T$, and $S^{-1} = \begin{bmatrix} r \\ Y \end{bmatrix}$, where $r \in \mathbb{R}^{1 \times N}$ and $Y \in \mathbb{C}^{(N-1) \times N}$. Then, we can verify that (20) can be written as

$$\begin{aligned}\dot{\zeta}_1(t) &= 0 \\ \dot{\phi}(t) &= -k_f \widehat{J}\phi(t) + F(t)\end{aligned}\quad (21)$$

where $F(t) = \dot{h}(t)\widehat{J}(Y - \widehat{J}Y)\widehat{x}(0)$ and $\widehat{J} \in \mathbb{C}^{(N-1) \times (N-1)}$ is the Jordan matrix associated with L for $\{\lambda_2, \dots, \lambda_N\}$, $\text{Re}(\lambda_i(L)) > 0$, $\forall i = \{2, \dots, N\}$.

It can be shown that the solutions of the differential equations (21) are the following [44]:

$$\begin{aligned}\zeta_1(t) &= \zeta_1(0) = 0 \\ \phi(t) &= e^{-k_f \widehat{J}t} \left(\phi(0) + \int_0^t e^{k_f \widehat{J}s} F(s) ds \right), \\ &\quad t \in [0, t_f]\end{aligned}\quad (22)$$

where $e^{-k_f \widehat{J}t} \in \mathbb{C}^{(N-1) \times (N-1)}$ is the exponential matrix. Then, it follows that

$$\lim_{t \rightarrow +\infty} \phi(t) = \lim_{t \rightarrow +\infty} [\zeta_2(t), \dots, \zeta_N(t)]^T = \mathbf{0}_{N-1}, \quad (23)$$

since the integral is a finite value due to $\dot{h}(t) = 0$, $\forall t \geq t_f$, and $-k_f \widehat{J}$ has eigenvalues with negative real parts [18, 45]. Notice that the convergence speed is determined by the dominant eigenvalue λ_2 of L ([15]) and the control gain k_f .

Since $\lim_{t \rightarrow +\infty} \zeta_i(t) = 0$, $\forall i \in \{1, \dots, N\}$, then

$$\begin{aligned}\lim_{t \rightarrow +\infty} \xi(t) &= S \lim_{t \rightarrow +\infty} \zeta(t) \\ &= S \lim_{t \rightarrow +\infty} [\zeta_1(t), \dots, \zeta_N(t)]^T \\ &= S[0, \dots, 0]^T = \mathbf{0}_N.\end{aligned}\quad (24)$$

Therefore, the tracking error converges to the origin. In fact, from (21) it can be seen that the tracking error is globally asymptotically stable, since the real part of the eigenvalues of $-k_f \widehat{J}$ is negative, $\zeta_1(t) = 0$, and $F(t)$ can be seen as a disturbance that becomes null for $t \geq t_f$. The stability of the tracking error implies that $e(t)$ follows $h(t)\widehat{e}_0$; thus $e(t)$ converges to the origin at the predefined-time t_f . Any small final error in time $t \geq t_f$ is corrected by the control input $u_i = (1/g_i(x_i))(-f_i(x_i) + k_f e_i(t))$, which is applied to each agent for $t \geq t_f$.

By using the property $L\mathbf{1}_N = \mathbf{0}_N$ (in accordance to Lemma 5), the converge of the consensus error (8) to the origin implies

$$x(t) \rightarrow x(t_f) = x^* \mathbf{1}_N \quad \text{as } t \rightarrow t_f, \quad (25)$$

i.e.,

$$x_i(t) \rightarrow x^* \quad \text{as } t \rightarrow t_f, \quad \forall i = \{1, \dots, N\}. \quad (26)$$

That is, the system (4) achieves predefined-time convergence to a consensus value x^* at t_f using the consensus protocol (16).

Now, let us compute the final consensus value. First, define $y(t) = \gamma^T x(t)$, where γ is defined in Lemma 5. For the two types of considered graphs, $y(t)$ is an invariant quantity, since

$$\begin{aligned}\dot{y}(t) &= \gamma^T \dot{x}(t) = \gamma^T (-\dot{h}(t)\widehat{e}_0 + k_f (e(t) - h(t)\widehat{e}_0)) \\ &= \gamma^T L(\dot{h}(t)\widehat{x}_0 - k_f (x(t) - h(t)\widehat{x}_0)) = 0, \\ &\quad \forall x(t)\end{aligned}\quad (27)$$

where $\gamma^T L = \mathbf{0}_N^T$ by Lemma 5. Thus,

$$\lim_{t \rightarrow t_f} y(t) = y(0) = \gamma^T x(0) \quad (28)$$

Moreover, from (25) it follows that

$$\lim_{t \rightarrow t_f} y(t) = \gamma^T \lim_{t \rightarrow t_f} x(t) = \gamma^T (x^* \mathbf{1}_N). \quad (29)$$

Then, for a connected undirected graph and by Lemma 5 ($\gamma = \mathbf{1}_N$), it is obtained that

$$\begin{aligned}\gamma^T x^* \mathbf{1}_N &= \gamma^T x_0, \\ x^* (\mathbf{1}_N)^T \mathbf{1}_N &= (\mathbf{1}_N)^T x_0,\end{aligned}\quad (30)$$

and solving for x^* , one has

$$x^* = \frac{\sum_{i=1}^N x_i(0)}{N} = \text{Ave}(\mathbf{x}_0). \quad (31)$$

Hence, the consensus value x^* is the average of the initial state \mathbf{x}_0 for connected undirected graphs. Now, for a directed graph with a directed spanning tree and by Lemma 5 ($\mathbf{y}^T \mathbf{1}_N = 1$), it results in

$$x^* = \frac{\mathbf{y}^T \mathbf{x}_0}{\mathbf{y}^T \mathbf{1}_N} = \mathbf{y}^T \mathbf{x}_0. \quad (32)$$

Therefore, the consensus value is given by (32) for a directed graph with a directed spanning tree.

Thus, system (4) with $\boldsymbol{\rho}(t) = \mathbf{0}_N$ achieves predefined-time convergence to a consensus value x^* at t_f in the presence of uncertainty in the initial state \mathbf{x}_0 using the consensus protocol (16). \square

Remark 8. It is worth noting that, in contrast to the use of the consensus protocol (13), the feed-forward term of v_i in (16) is not sufficient to drive the consensus error trajectory to the TBG reference trajectory $\hat{\mathbf{e}}(t) = h(t)\hat{\mathbf{e}}_0$ even for $\mathbf{d} = \mathbf{0}_N$; consequently, system (4) under the first term of (16) does not reach predefined-time convergence to a consensus value x^* at t_f . In such a case, the final state is given by $\mathbf{x}(t_f) = (\mathbf{I}_N - \mathbf{L})\mathbf{x}_0 - \mathbf{L}\mathbf{d}$, where \mathbf{I}_N denotes the identity matrix of size N . Therefore, the use of the feedback term in (16) is required in any case.

It can be seen that the consensus protocol (16) can solve the problem of predefined-time consensus stated in Section 2.2 by tracking the trajectories given by the TBG even when considering uncertainty \mathbf{d} in the initial state. In this sense, the compensation term for tracking must guarantee reaching the TBG trajectories before the desired convergence time t_f . This means that the tracking control gain k_f must be higher for a larger uncertainty in the initial state. Then, there is a limit on the values of \mathbf{d} that can be managed by the consensus protocol.

Remark 9. Theorem 7 represents a novel contribution on consensus for nonlinear MAS: First, the predefined-time consensus is solved for nonlinear first-order systems. Second, the auxiliary control inputs $v_i(t)$ in (16) are smooth signals even at $t = 0$, which is not the case of other approaches reported in the literature. Third, the proposed consensus protocols do not have parameters depending on the algebraic connectivity of the graph considered. Finally, the stability analysis is performed by using standard techniques for linear systems (upon feedback linearization of the original systems), while most of the works regarding finite-time and fixed-time and even the few existing works on predefined-time convergence adopt more sophisticated techniques such as homogeneity theory and Lyapunov functions.

Remark 10. The result introduced by Theorem 7, compared to our previous conference work [34], is twofold: (1) a new consensus protocol is proposed, achieving predefined-time

convergence of the unperturbed MAS (4) without the need of specifying a consensus value; (2) the effect of considering uncertainty in the knowledge of the initial conditions is analyzed, demonstrating that convergence to a consensus value can still be achieved.

3.3. Robust Predefined-Time Consensus. In the previous consensus protocol, we have considered unperturbed dynamics (i.e., $\boldsymbol{\rho}(t) = \mathbf{0}_N$). To effectively deal with disturbances, a robust protocol is introduced in this section, by combining the super-twisting controller (STC), which can compensate for matched uncertainties/disturbances [46], with the TBG, leading to a robust predefined-time controller.

In this scheme, it is assumed that there exists a leader $x_l(t) \in \mathbb{R}$ in the network system, whose dynamics can be modeled as in [47] by

$$\dot{x}_l(t) = u_l(t) \quad (33)$$

where $u_l(t) \in \mathbb{R}$ is the leader's control input. In this case, N agents are now the followers with dynamics (3). The consensus error for each follower is defined as [47]

$$e_i^f(t) = \sum_{j \in N_i} a_{ij} (x_j(t) - x_i(t)) - b_i (x_i(t) - x_l(t)) \quad (34)$$

where $b_i \in \mathbb{R}$ represents the leader's adjacency with $b_i > 0$ if agent i is a neighbor of the leader; otherwise $b_i = 0$, $\forall i \in \{1, \dots, N\}$.

The tracking error is given by

$$\xi_i(t) = e_i^f(t) - h(t)\hat{\mathbf{e}}_i(0) \quad (35)$$

where $h(t)\hat{\mathbf{e}}_i(0)$ is the TBG reference trajectory for the i -th and $\hat{\mathbf{e}}_i(0)$ is computed from the initial state estimate $\hat{\mathbf{x}}_0$ as in (15). The tracking error will be used as the sliding surface; thus

$$s_i(t) = \xi_i(t) = e_i^f(t) - h(t)\hat{\mathbf{e}}_i(0). \quad (36)$$

Theorem 11. Consider a MAS modeled as in (4) with a leader (33). For each agent i , define $\beta_i = (\sum_{j \in N_i} a_{ij} + b_i)$ and the sliding surface (36). Consider a function $h(t)$ as in (10). There exist gains $k_1 > 0$ and $k_2 > \varrho(M)$, for some function $\varrho(\bullet)$ and a constant $M > 0$, such that the following nonlinear controller defined for each agent

$$\begin{aligned} u_i(t) &= (\beta_i g_i)^{-1} (\bar{v}_i + v_i), \quad i \in \{1, \dots, N\} \\ \bar{v}_i &= b_i u_l + \sum_{j \in N_i} a_{ij} (f_j + g_j u_j) - \beta_i f_i - \dot{h}(t)\hat{\mathbf{e}}_i(0) \\ v_i &= k_1 |s_i|^{1/2} \text{sign}(s_i) - w_i \\ \dot{w}_i &= -k_2 \text{sign}(s_i) \end{aligned} \quad (37)$$

achieves predefined-time convergence to the leader's state x_l at t_f , allowing $\rho_i(t) \neq 0$, from any initial state $x_i(0)$.

Proof. First, taking the time derivative of the tracking error (35) and using the dynamics of the followers (3) and the leader (33), it is obtained that

$$\begin{aligned}
\dot{\xi}_i(t) &= \dot{e}_i^f(t) - \dot{h}(t) \hat{e}_i(0) \\
&= \sum_{j \in N_i} a_{ij} (\dot{x}_j(t) - \dot{x}_i(t)) - b_i (\dot{x}_i(t) - \dot{x}_i(t)) \\
&\quad - \dot{h}(t) \hat{e}_i(0) \\
&= -\beta_i (f_i + g_i u_i) + \sum_{j \in N_i} a_{ij} (f_j + g_j u_j) + b_i u_i \\
&\quad - \beta_i \rho_i + \sum_{j \in N_i} a_{ij} \rho_j - \dot{h}(t) \hat{e}_i(0).
\end{aligned} \tag{38}$$

Now, using (38) and plugging the control law (37) in the time derivative of the sliding surface (36), it results in

$$\begin{aligned}
\dot{s}_i &= \dot{\xi}_i(t) = -k_1 |s_i|^{1/2} \text{sign}(s_i) + w_i + \rho_{d_i} \\
\dot{w}_i &= -k_2 \text{sign}(s_i),
\end{aligned} \tag{39}$$

where $\rho_{d_i} = \sum_{j \in N_i} a_{ij} \rho_j - \beta_i \rho_i$. Let $z_i = w_i + \rho_{d_i}$; then the previous expression can be rewritten as

$$\begin{aligned}
\dot{s}_i &= -k_1 |s_i|^{1/2} \text{sign}(s_i) + z_i \\
\dot{z}_i &= -k_2 \text{sign}(s_i) + \dot{\rho}_{d_i}.
\end{aligned} \tag{40}$$

It has been proven in [46] that, for a bounded continuously differentiable disturbance, i.e., if $|\rho_{d_i}| < L$ and $|\dot{\rho}_{d_i}| < M$ for some constants $L > 0$, $M > 0$, the second-order dynamics (40) converge globally to the origin ($s_i = 0, z_i = 0$) in finite time in spite of the disturbance, if adequate positive control gains $k_1 > 0$ and $k_2 > \varrho(M)$ are used. Moreover, the remaining dynamics of the tracking error system are constrained to the sliding surface, such that $s_i = \dot{s}_i = 0$, meaning that each agent $e_i^f(t)$ follows $h(t) \hat{e}_i(0)$. Consequently, the consensus error converges to zero in the predefined-time t_f and the consensus value is given by the leader's state x_l [47]. \square

Remark 12. To the best of our knowledge, the only works that propose predefined-time consensus protocols by other authors are [30–32]. The consensus control law (37) has the advantage, over those protocols, of providing robustness against large matched perturbations.

Some important issues to be considered during real-world applications of the proposed distributed control protocols are threefold: First, all the clocks of the agents in the network must be synchronized to achieve predefined-time convergence. Second, physical constraints of the systems must be taken into account to set t_f , considering that a small t_f will result in large control inputs. Thus the maximum allowable input of each agent must be taken into consideration to set t_f . Third, we are currently assuming no time delay in the agent's communication; however, time delays may exist in some systems and may affect the convergence to the consensus.

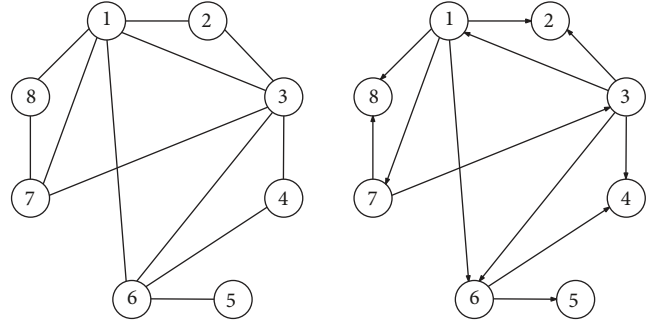


FIGURE 1: Communication graphs. Left: undirected graph \mathcal{G}_1 . Right: directed graph \mathcal{G}_2 .

4. Simulation Results

In this section, simulation results are shown to illustrate the advantages of the proposed protocols. All the simulations consider a MAS of 8 agents, described by (3) with $f_i(x_i) = x_i^2$ for agents 1 to 4, $f_i(x_i) = \sin(x_i)$ for agents 5 to 8, and $g_i(x_i) = 5$ for all the 8 agents. The convergence time is preset to $t_f = 5$ seconds. For this, the TBG is described by the function $h(t) = 2(t/t_f)^3 - 3(t/t_f)^2 + 1$, which fulfills (9), i.e., $h(0) = 1$, $h(5) = 0$, and $\dot{h}(0) = \dot{h}(5) = 0$. Both proposed distributed control laws are implemented as defined in (16) and (37). The implementation essentially requires the computation of the TBG reference and its time derivative, which means to evaluate the functions $h(t)$ and $\dot{h}(t)$ at each iteration and compute the consensus error for each agent depending on the neighbors state as given by (7).

We use the communication topologies shown in Figure 1 (obtained from [30]), denoted as \mathcal{G}_1 and \mathcal{G}_2 , where \mathcal{G}_1 is a connected undirected graph and \mathcal{G}_2 is a directed graph having a directed spanning tree. The simulations were implemented in MATLAB using the Euler forward method to approximate the time derivatives with a time step of 0.1 ms.

The initial states of the eight agents are $\mathbf{x}_0 = [-0.61, 3.52, -1.07, 0.73, 1.48, 0.19, 1.97, 0.72]^T$. The uncertainties in the initial states are set to be $\mathbf{d} = [0.22, 0.15, -0.39, 0.27, 0.34, 0.21, -0.36, -0.09]^T$.

4.1. Simulation of the Consensus Protocol Based on the Error Feedback. This subsection is devoted to showing the performance of the TBG protocol (16). Figures 2 and 3 show the transient behavior of the MAS under (16) with $k_f = 100$, for \mathcal{G}_1 and \mathcal{G}_2 , respectively. For the case shown in Figure 2, perfect knowledge of the initial conditions is assumed, while in Figure 3, there exists uncertainty in the initial condition knowledge, given by \mathbf{d} . In both cases the agent's state (top graphs in the figures) reaches consensus at the predefined 5 seconds.

It can be seen in Figure 2 (middle) that the auxiliary control inputs $v_i(t)$ generated by the control protocol (16) are smooth signals even at $t = 0$. This is due to the properties of the TBG, which yields that each v_i starts at zero and returns to zero at time t_f . This is not the case for the control inputs u_i , since they include the nonlinearities that must be cancelled.

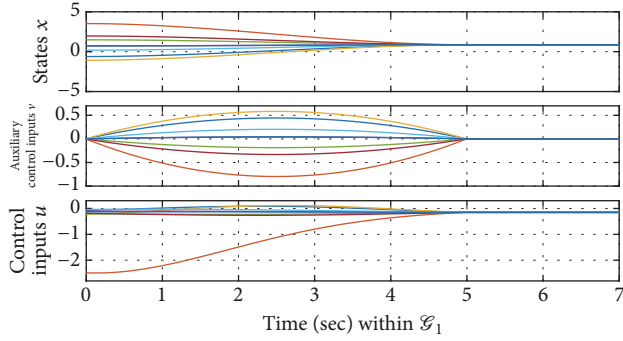


FIGURE 2: Predefined-time TBG-tracking controller (16) for \mathcal{S}_1 . The convergence time is preset to 5 sec., and the consensus value is $\sum_{i=1}^N x_i(0)/N = 0.8676$.

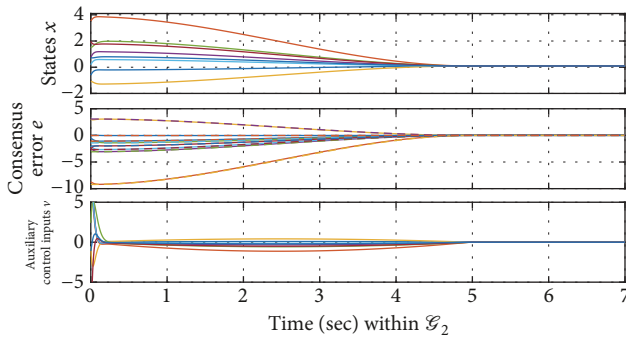


FIGURE 3: Predefined-time TBG-tracking controller (16) for \mathcal{S}_2 and uncertainty in the initial state. In the error trajectories, the continuous lines represent the evolution of the errors whereas the TBG reference is drawn with dash line. The convergence time is preset to 5 sec., and the consensus value is $\gamma^T x_0 = 0.0946$.

In the case where the uncertainty \mathbf{d} in the initial state is considered (Figure 3), the second term of v_i in (16) is not initially null to compensate the deviation from the reference given by the TBG. It can be seen in Figure 3 (middle and bottom) that the consensus errors do not initiate over the references and consequently the auxiliary controls are not null at $t = 0$.

4.2. Comparison with Others Approaches. In this subsection, first, we compare the proposed control law (16) with other predefined-time control protocols [30–32]. As described in Section 1, to the authors' knowledge, these are the existing approaches that solve the problem of predefined-time convergence. Each one of these works refers to their proposal with a different adjective: preset-time consensus [30], prescribed-time consensus [31], and specified-time consensus [32]. We can see in Figures 4–6 that consensus is achieved by using any of these approaches. We have to say that the control protocols of [30, 31] are very similar; both of them are based on the use of a time-varying control gain. Between their similarities, we can see in Figures 4 and 5 the large initial value of the auxiliary control inputs. This is also the case for the specified-time consensus [32] of Figure 6. Notice that the auxiliary control inputs of the TBG-based controller, shown

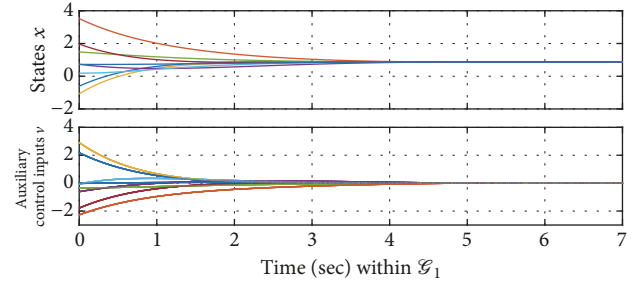


FIGURE 4: Preset-time controller [30] for \mathcal{S}_1 . The convergence time is preset to 5 sec., and the consensus value is $\sum_{i=1}^N x_i(0)/N = 0.8676$.

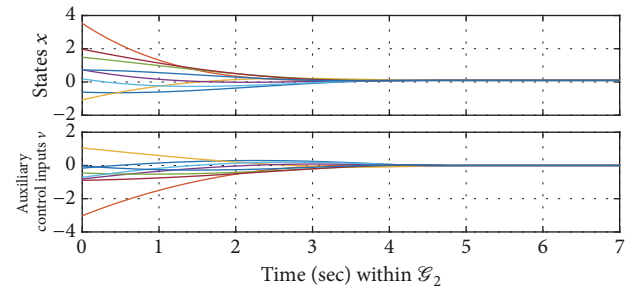


FIGURE 5: Prescribed-time controller [31] for \mathcal{S}_2 . The convergence time is preset to 5 sec., and the consensus value is 0.0946.

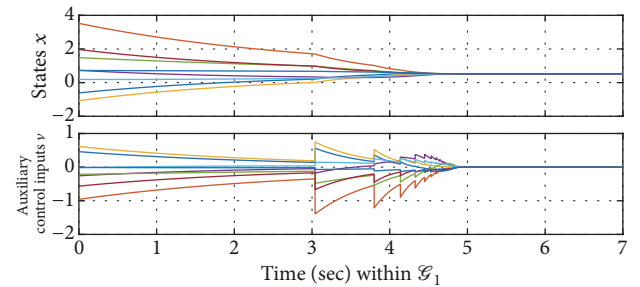


FIGURE 6: Specified-time controller [32] for \mathcal{S}_1 . The convergence time is preset to 5 sec., and the consensus value is 0.5196.

in Figure 2, started in zero and were smoother and of lower magnitude compared to the large initial auxiliary control effort of the other predefined-time consensus approaches.

Additionally, we can see that the specified-time consensus protocol in Figure 6 generates discontinuities in the auxiliary inputs along the time, according to its planned switching strategy. Moreover, although this consensus approach was tested on the undirected graph \mathcal{S}_1 , the consensus value is not the average of the initial state as one could expect.

Following the comparison, we evaluate the TBG controller (16) and different finite-time, fixed-time, and predefined-time consensus protocols reported in the literature, by applying the protocols to the MAS with the communication topologies \mathcal{S}_1 and \mathcal{S}_2 . In particular, the protocols in [20, 48] ensure finite-time convergence, and the one proposed in [27] guarantees fixed-time convergence. The predefined-time protocols [30–32] and the proposed

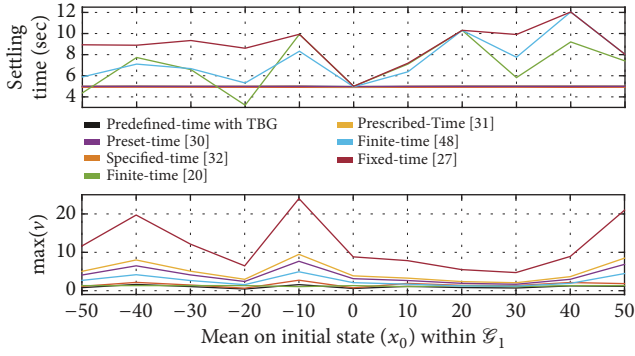


FIGURE 7: Comparison of the TBG-tracking controller (16) with other approaches for \mathcal{S}_1 . Top: settling time as a function of the initial condition \mathbf{x}_0 . Bottom: maximum value of the auxiliary control input as a function of the initial condition \mathbf{x}_0 .

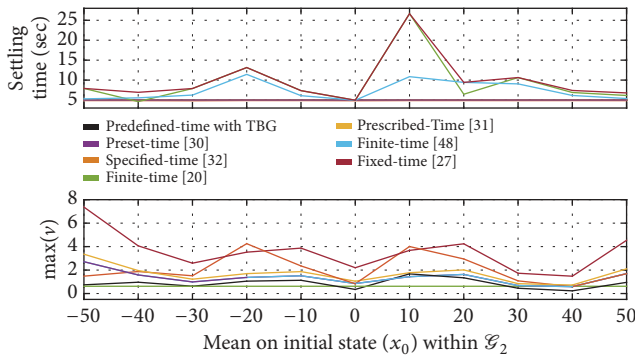


FIGURE 8: Comparison of the TBG-tracking controller (16) with other approaches for \mathcal{S}_2 . Top: settling time as a function of the initial condition \mathbf{x}_0 . Bottom: maximum value of the auxiliary control input as a function of the initial condition \mathbf{x}_0 .

TBG-tracking controller (16) achieve predefined-time convergence. The finite-time and fixed-time controllers were tuned to achieve a similar convergence time around the 5 seconds of predefined time, from the initial condition \mathbf{x}_0 . Then, several simulations were performed for different initial states of the form $\alpha\mathbf{x}_0$, where α was varying in such a way that the mean of $\alpha\mathbf{x}_0$ ranges from -50 to 50 . The same control gains were used in all the experiments. Finally, for every experiment, the convergence time of the system, defined as the time at which $\|\mathbf{e}\| < 1 \times 10^{-4}$, was measured; similarly, the maximum absolute value of the auxiliary control input \mathbf{v} was recorded.

The results are shown in Figures 7 and 8, where it can be observed that the proposed TBG-based control (16) and the other predefined-time controllers [30–32] are able to maintain the same convergence time for all the initial conditions. Furthermore, note that the maximum auxiliary control efforts were lower for both the TBG-based control and the finite-time control [20]. The last control law is of the form $k \text{sign}(\bullet)$. In particular, the consensus protocols [30, 31] explicitly include a parameter that must be set accordingly to an eigenvalue of the Laplacian matrix, which depends on the connectivity of the graph considered. This is not the case

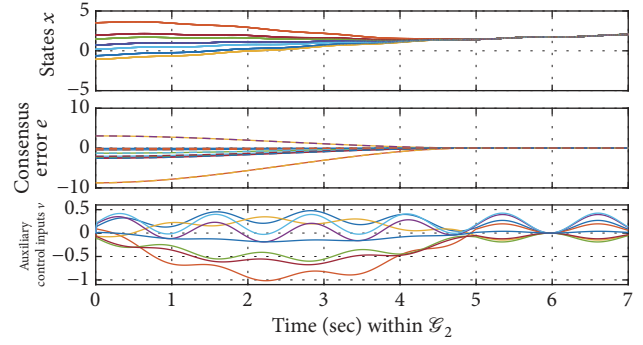


FIGURE 9: Predefined-time TBG-tracking controller (16), which is not robust for perturbed dynamics and \mathcal{S}_2 .

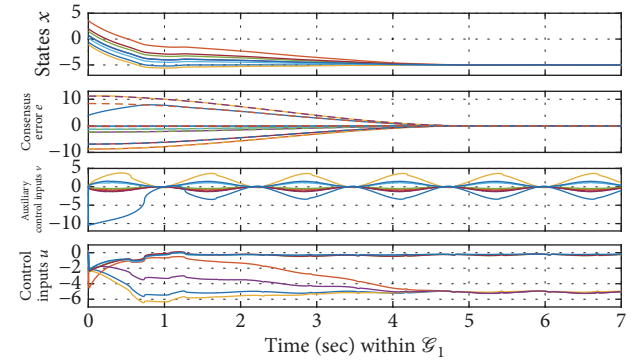


FIGURE 10: Robust predefined-time TBG controller (37) for perturbed systems and \mathcal{S}_1 . In the error trajectories, the continuous lines represent the evolution of the errors whereas the TBG reference is drawn with dash line. The convergence time is preset to 5 sec.

for the proposed control protocols. Therefore, the simulations show that the proposed TBG protocol (16) represents an advantage not only in the possibility to define a settling time but also in providing smooth and efficient control actions.

4.3. Robust Predefined-Time Consensus. All the previous simulations considered ideal dynamics in (3), i.e., $\rho_i(t) = 0$ for all the agents. In fact, the TBG-tracking controller (16) is not able to deal with matched bounded disturbances. For instance, Figure 9 shows the MAS with the topology \mathcal{S}_2 under this protocol and matching perturbations $\rho_i(t) = \alpha_i(1 + 1 \sin(5t))$, with α_i randomly selected in $(0, 0.5)$. It can be observed that the state of the MAS does not establish in a consensus value due to the perturbation. A similar behavior can be obtained with the other existing predefined-time consensus protocols [30–32].

In order to deal with the described matching perturbations $\rho_i(t) = \alpha_i(1 + 1 \sin(5t))$, the proposed robust predefined-time TBG controller (37) has to be used. For this, a leader is defined with dynamics (33), having communication only with the first follower agent, i.e., $b_1 = 1$ and $b_i = 0, \forall i = \{2, \dots, 8\}$. The gains of the robust controller (37) were set as $k_1 = 5$ and $k_2 = 5$; and the leader's state was maintained constant and equal to $x_l(t) = -5$ by setting its control input as $u_l = 0$. Figures 10 and 11 show the results obtained when the

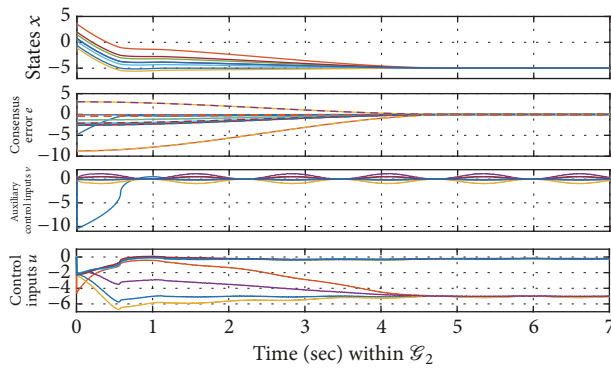


FIGURE 11: Robust predefined-time TBG controller (37) for perturbed systems and \mathcal{E}_2 . In the error trajectories, the continuous lines represent the evolution of the errors whereas the TBG reference is drawn with dash line. The convergence time is preset to 5 sec.

communication between the followers is described by \mathcal{E}_1 and \mathcal{E}_2 , respectively. Notice that the error trajectories are driven to zero in the predefined time $t_f = 5$, while the control input keeps oscillating after t_f in order to reject the disturbance $\rho(t)$.

5. Conclusions

In this paper, a couple of predefined-time consensus protocols for first-order nonlinear MAS under both undirected and directed communication topologies have been proposed. In these protocols, the convergence time to achieve consensus can be set by the user, and it is independent of the initial state conditions. For this, the TBG (time base generator) is combined with feedback controllers to achieve closed-loop stability and robustness. In particular, one protocol uses the super-twisting controller to deal with perturbations. The performance of the proposed controllers has been compared with existing finite-time, fixed-time, and predefined-time controllers through simulations. The results have shown that the proposed protocols achieve consensus in the predefined time, independently of the initial conditions, and exhibit closed-loop stability. Moreover, a benefit of the proposed controllers is that they yield smoother control signals with smaller magnitude than the existing approaches reported in the literature. Furthermore, the proposed schemes can reach predefined-time consensus even when there exists uncertainty in the knowledge of the initial conditions, and the super-twisting protocol is robust against matching perturbations.

Future works will focus on extending the results presented in this paper to the case of high-order nonlinear MAS, as well as in considering the predefined-time consensus problem in Markovian jump topologies [4, 5].

Data Availability

The paper is a theoretical result on consensus with predefined-time convergence. All proofs are presented, and simulations are provided to illustrate the result. In the

authors' opinion, there is no need to provide the numerical experiments; simulations can be provided and made available under request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Acknowledgments

The first two authors were supported in part by Conacyt [grant No. 220796]. The authors would also like to acknowledge the partial support of Intel Corporation for the development of this work.

References

- [1] 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.
- [2] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.
- [3] A. Arenas, A. D. Guilera, J. Kurths, Y. Moreno, and C. Zhou, "Synchronization in complex networks," *Physics Reports*, vol. 469, no. 3, pp. 93–153, 2008.
- [4] H. Shen, J. H. Park, Z.-G. Wu, and Z. Zhang, "Finite-time H_∞ synchronization for complex networks with semi-Markov jump topology," *Communications in Nonlinear Science and Numerical Simulation*, vol. 24, no. 1–3, pp. 40–51, 2015.
- [5] H. Shen, S. Huo, J. Cao, and T. Huang, "Generalized state estimation for Markovian coupled networks under round-robin protocol and redundant channels," *IEEE Transactions on Cybernetics*, pp. 1–10, 2018.
- [6] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [7] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.
- [8] W. Ren, "Distributed attitude alignment in spacecraft formation flying," *International Journal of Adaptive Control and Signal Processing*, vol. 21, no. 2–3, pp. 95–113, 2007.
- [9] Y. Xu, G. Yan, K. Cai, and Z. Lin, "Fast centralized integer resource allocation algorithm and its distributed extension over digraphs," *Neurocomputing*, vol. 270, pp. 91–100, 2017.
- [10] Y. Xu, T. Han, K. Cai, Z. Lin, G. Yan, and M. Fu, "A distributed algorithm for resource allocation over dynamic digraphs," *IEEE Transactions on Signal Processing*, vol. 65, no. 10, pp. 2600–2612, 2017.
- [11] D. Gómez-Gutiérrez and R. De La Guardia, "Chaotic-based synchronization for secure network communications, US Patent 9438422B2," 2016.
- [12] 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.
- [13] Z. Zuo and L. Tie, "A new class of finite-time nonlinear consensus protocols for multi-agent systems," *International Journal of Control*, vol. 87, no. 2, pp. 363–370, 2014.

- [14] D. Gómez-Gutiérrez, J. Ruiz-León, S. Celikovsky, and J. D. Sánchez-Torres, "A finite-time consensus algorithm with simple structure for fixed networks," *Computación y Sistemas*, vol. 22, no. 2, 2018.
- [15] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [16] W. Ren and R. W. Beard, *Distributed Consensus in Multi-Vehicle Cooperative Control*, Springer, 2008.
- [17] K. Cai and H. Ishii, "Average consensus on arbitrary strongly connected digraphs with time-varying topologies," *Institute of Electrical and Electronics Engineers Transactions on Automatic Control*, vol. 59, no. 4, pp. 1066–1071, 2014.
- [18] Z. Li and Z. Duan, *Cooperative Control of Multi-Agent Systems: A Consensus Region Approach*, CRC Press, Boca Raton, Fla, USA, 2014.
- [19] H. Sayyaadi and M. R. Doostmohammadian, "Finite-time consensus in directed switching network topologies and time-delayed communications," *Scientia Iranica*, vol. 18, no. 1, pp. 75–85, 2011.
- [20] M. Franceschelli, A. Giua, A. Pisano, and E. Usai, "Finite-time consensus for switching network topologies with disturbances," *Nonlinear Analysis: Hybrid Systems*, vol. 10, pp. 83–93, 2013.
- [21] J. Cortés, "Finite-time convergent gradient flows with applications to network consensus," *Automatica*, vol. 42, no. 11, pp. 1993–2000, 2006.
- [22] M. Franceschelli, A. Giua, and A. Pisano, "Finite-time consensus on the median value with robustness properties," *Institute of Electrical and Electronics Engineers Transactions on Automatic Control*, vol. 62, no. 4, pp. 1652–1667, 2017.
- [23] B. Liu, W. Lu, and T. Chen, "Consensus in continuous-time multiagent systems under discontinuous nonlinear protocols," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 2, pp. 290–301, 2015.
- [24] Y.-K. Zhu, X.-P. Guan, and X.-Y. Luo, "Finite-time consensus for multi-agent systems via nonlinear control protocols," *International Journal of Automation and Computing*, vol. 10, no. 5, pp. 455–462, 2013.
- [25] Y. Shang, "Finite-time consensus for multi-agent systems with fixed topologies," *International Journal of Systems Science*, vol. 43, no. 3, pp. 499–506, 2012.
- [26] S. E. Parsegov, A. E. Polyakov, and P. S. Shcherbakov, "Fixed-time consensus algorithm for multi-agent systems with integrator dynamics," in *Proceedings of the 4th IFAC Workshop on Distributed Estimation and Control in Networked Systems, NecSys 2013*, pp. 110–115, September 2013.
- [27] Z. Zuo, Q. L. Han, B. Ning, X. Ge, and X. M. Zhang, "An overview of recent advances in fixed-time cooperative control of multiagent systems," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2322–2334, 2018.
- [28] E. Cruz-Zavala, J. A. Moreno, and L. Fridman, "Uniform Second-Order Sliding Mode Observer for mechanical systems," in *Proceedings of the 2010 11th International Workshop on Variable Structure Systems, VSS 2010*, pp. 14–19, Mexico, June 2010.
- [29] 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.
- [30] C. Yong, X. Guangming, and L. Huiyang, "Reaching consensus at a preset time: single-integrator dynamics case," in *Proceedings of the 31st Chinese Control Conference (CCC)*, pp. 6220–6225, IEEE, 2012.
- [31] 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.
- [32] Y. Zhao and Y. Liu, "Specified-time consensus for multi-agent systems," in *Proceedings of the 2017 Chinese Automation Congress (CAC)*, pp. 732–737, Jinan, October 2017.
- [33] Y. Liu, Y. Zhao, W. Ren, and G. Chen, "Appointed-time consensus: accurate and practical designs," *Automatica*, vol. 89, pp. 425–429, 2018.
- [34] J. A. Colunga, C. R. Vázquez, H. M. Becerra, and D. Gómez-Gutiérrez, "Predefined-time consensus using a time base generator (TBG)," in *Proceedings of the IFAC Second Conference on Modelling, Identification and Control of Nonlinear Systems (MICNON)*, vol. 51, pp. 246–253, 2018.
- [35] H. M. Becerra, C. R. Vazquez, G. Arechavaleta, and J. Delfin, "Predefined-time convergence control for high-order integrator systems using time base generators," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 5, pp. 1866–1873, 2018.
- [36] H. K. Khalil and J. W. Grizzle, *Nonlinear Systems*, vol. 3, Prentice Hall, New Jersey, NJ, USA, 2002.
- [37] Q. Ma, S. Xu, F. L. Lewis, B. Zhang, and Y. Zou, "Cooperative output regulation of singular heterogeneous multiagent systems," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp. 1471–1475, 2016.
- [38] Q. Ma, "Cooperative control of multi-agent systems with unknown control directions," *Applied Mathematics and Computation*, vol. 292, pp. 240–252, 2017.
- [39] A. H. Roger and R. J. Charles, *Matrix Analysis*, Cambridge University Press, New York, NY, USA, 2nd edition, 2013.
- [40] W. Yu, G. Wen, G. Chen, and J. Cao, *Distributed Cooperative Control of Multi-agent Systems*, John Wiley and Sons, 2016.
- [41] Z. Li and Z. Duan, "Distributed consensus protocol design for general linear multi-agent systems: a consensus region approach," *IET Control Theory & Applications*, vol. 8, no. 18, pp. 2145–2161, 2014.
- [42] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 213–224, 2010.
- [43] G. Jarquín, G. Arechavaleta, and V. Parra-Vega, "Time parametrization of prioritized inverse kinematics based on terminal attractors," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems: Celebrating 50 Years of Robotics, IROS'11*, pp. 1612–1617, USA, September 2011.
- [44] M. Braun, *Differential Equations and Their Applications: An Introduction to Applied Mathematics*, vol. 11, Springer Science and Business Media, Third edition, 1983.
- [45] H. K. Khalil and J. Grizzle, *Nonlinear systems*, vol. 3, Prentice hall, Upper Saddle River, New Jersey, NJ, USA, 2002.
- [46] J. A. Moreno and M. Osorio, "Strict Lyapunov functions for the super-twisting algorithm," *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 1035–1040, 2012.
- [47] S. Mondal, R. Su, and L. Xie, "Heterogeneous consensus of higher-order multi-agent systems with mismatched uncertainties using sliding mode control," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 13, pp. 2303–2320, 2017.
- [48] Y. Cao and W. Ren, "Finite-time consensus for multi-agent networks with unknown inherent nonlinear dynamics," *Automatica*, vol. 50, no. 10, pp. 2648–2656, 2014.

