Distributed Adaptive Output Agreement in a Class of Multi-Agent Systems

Veysel Gazi

Abstract—**In this article we consider the distributed output agreement problem in a class of heterogeneous multi-agent dynamic systems composed of agents with nonlinear and uncertain dynamics. We develop a distributed direct adaptive fuzzy control methodology which using only local information guarantees achievement of agreement of the agent outputs despite the uncertainties in the agent dynamics. The performance of the proposed strategy is verified for various neighborhood topologies using representative numerical simulations.**

I. INTRODUCTION

The distributed agreement problem is a problem in which multiple independent agents are required to agree on variables of interest using only limited local information. The variables of interest can be the states or the outputs (i.e., some function of the states) of the agents. The agents can obtain information only from their neighbors and do not have access to global information. There are various studies in the literature which consider the problem from different perspectives under various related terminology such as consensus, synchronization, rendezvous, and distributed agreement [1], [2]. Some authors use rendezvous to describe achieving agreement in finite time, consensus to describe achieving agreement at a common point, whereas synchronization to describe achieving agreement at a common trajectory. In this article we adapt the terminology distributed agreement as a general term to describe any type of agreement.

Developing conditions which facilitate agreement under limited local information and developing local agreement controllers which can be implemented in engineering multi-agent systems are the central questions in the literature on distributed agreement. Most of the studies in the literature are on the state agreement problem [3]–[8]. Popular state agreement models are averaging models under bidirectional communication [3], time-dependent unidirectional communication [4], time-dependent unidirectional communication with asynchronous operation and time delays in the information flow [5]. General nonlinear contracting models [6], which are related to the averaging models, have been also considered. Works on continuous time models with time delays [7] and more complicated agent dynamics such as unicycles [8] are also available in the literature. All of the above mentioned studies agree on the importance of the connectivity between the agents in the group/swarm for the information flow and therefore for facilitating agreement. In particular, an important condition is the requirement that the connectivity graph needs to possess a spanning tree uniformly in time. In other words, in order to be able to achieve agreement among the agents there must be an agent which is uniformly connected to all other agents in the swarm.

The output agreement problem, which is in general a more difficult problem, has been also considered in the literature [9]–[14]. The studies on output agreement build on the results on state agreement in a sense that the necessary connectivity requirements mentioned above are assumed to be satisfied. Then, given more complicated agent dynamics, the studies focus on developing additional conditions [9], [10] and/or controllers [11]–[14] facilitating agreement. Parametric uncertainties [12]–[14] and local disturbances [14] are also considered. The adverse effects of system uncertainties and unknown disturbances can be suppressed using robust or adaptive strategies. In the cases in which the uncertainties are parametric one can apply conventional adaptive control techniques. In the cases in which the uncertainties cannot be parameterized one can use intelligent control techniques such as neural networks or fuzzy systems to estimate and to counter affect them. Neural networks and fuzzy systems possess the universal approximation property meaning that they can arbitrarily closely approximate any smooth function on a compact set. This property has lead to extensive utilization of these intelligent systems in the adaptive control literature [15]–[20] including the literature on multi-agents dynamic systems [21]. Adaptive strategies with other type of universal approximators have been utilized as well [22]. Adaptive control schemes can be categorized into direct and indirect approaches. In the indirect adaptive control approach the unknown plant dynamics are estimated and utilized in a certainty equivalence based controller. In the direct adaptive control approach, on the other hand, the unknown ideal controller is directly estimated and utilized without estimating the plant dynamics.

In this article we consider the problem of distributed output agreement in a class of heterogeneous multi-agent dynamic systems composed of agents with nonlinear and uncertain dynamics. Inspired by the work in [18], we develop a distributed direct adaptive fuzzy control methodology which guarantees achievement of agreement of the agent outputs despite the uncertainties in the agent dynamics. The only information the agents can measure is the error between their outputs and the outputs of their neighbors. In order to improve robustness and to guarantee stability we also augment the direct adaptive fuzzy controller with sliding mode and bounding terms. The performance of the proposed strategy is verified for various neighborhood topologies using representative numerical simulations.

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under grant no 109E175.

Veysel Gazi is with İstanbul Kemerburgaz University, Department of Electrical and Electronics Engineering, Mahmutbey Dilmenler Caddesi No: 26, 34217 Bağcılar, İstanbul, TURKEY (email: veysel.gazi@kemerburgaz.edu.tr).

978-1-4673-5769-2/13/$31.00 ©2013 IEEE

II. PROBLEM DEFINITION

Consider a multi-agent system composed of N agents with individual agent dynamics

$$\begin{aligned} \dot{x}_i &= f_i(x_i) + g_i(x_i)u_i \\ y_i &= h_i(x_i) \end{aligned} \tag{1}$$

where $x_i \in \mathbb{R}^n$ is the state vector, $u_i \in \mathbb{R}^m$ is the control input and $y_i \in \mathbb{R}^m$ is the output of agent i. The functions $f_i : \mathbb{R}^n \to \mathbb{R}^n$, $g_i : \mathbb{R}^n \to \mathbb{R}^{n\times m}$, and $h_i : \mathbb{R}^n \to \mathbb{R}^m$ are sufficiently smooth functions for all $i = 1, ..., N$. We assume that in the region of interest the agent dynamics of all agents have a well defined vector relative degree $\{r, r, ..., r\}$ and can be transformed into the form

$$\begin{aligned} \dot{z}_i &= f_{oi}(z_i, \xi_i) \\ \dot{\xi}_{i,1} &= \xi_{i,2} \\ &\cdots \\ \dot{\xi}_{i,r-1} &= \xi_{i,r} \\ \dot{\xi}_{i,r} &= [\alpha_{ki}(t) + \alpha_i(x_i)] + [\beta_i(x_i)]\, u_i \\ y_i &= \xi_{i,1} \end{aligned} \tag{2}$$

where $\xi_{i,j} \in \mathbb{R}^m, j = 1, ..., r$, $\xi_i^\top = [\xi_{i,1}^\top, ..., \xi_{i,r}^\top] \in \mathbb{R}^{rm}$, and $z_i \in \mathbb{R}^{n-rm}$. In these dynamics it is assumed that the functions $\alpha_{ki}(t)$ are known. In contrast, the functions $\alpha_i(x_i)$ and $\beta_i(x_i)$ are assumed to be unknown. Since the agents have well defined relative degree the $m \times m$ matrices $[\beta_i(x_i)]$ are always non-singular and assumed to satisfy $0 < \underline{\beta_i} \leq \|\beta_i(x_i)\| \leq \overline{\beta_i} < \infty$. Moreover, it is assumed that within the region of operation we have $\left\|\dot{\beta}_i(x_i)\right\| = \left\|\frac{\partial \beta_i}{\partial x_i}\dot{x}_i\right\| \leq B_i(x_i)$ for some known functions $B_i(x_i) \geq 0$. Note that if $[\beta_i(x_i)]$ is a constant matrix this assumption is trivially satisfied. Furthermore, we assume that for all i the agent dynamics are minimum phase and the zero dynamics of the agents are exponentially stable. Note that all these assumptions are reasonable assumptions.

Consider the problem in which the agents are required to achieve agreement on their output variables $y_i(t)$ using only local information. By local information it is meant that the agents have access to information from only their neighbors. Let us denote with $N_i(t), i = 1, ..., N$ the set of agents from which agent i can obtain information at time t (i.e., the neighbors of agent i at time t). Then, more formally the distributed output agreement problem can be stated as follows.

Problem 1: (Distributed Output Agreement Problem) For every agent $i, i = 1, \ldots, N$, design the control inputs $u_i(t)$ based on information from only their neighbors $N_i(t)$ such that the agent outputs $y_i(t)$ satisfy

$$\lim_{t\to\infty} \|y_i(t) - y_j(t)\| = 0 \tag{3}$$

for all i and j, $1 \leq i, j \leq N$.

From the studies on distributed state agreement [3]–[8] we know that the connectivity between the agents plays important role in the information flow in the swarm. Let us represent the neighborhood topology of the swarm at time t with a directed graph $\mathcal{G}(t) = (\mathcal{N}, \mathcal{E}(t), \mathcal{W}(t))$, where $\mathcal{N} = \{1, 2, \ldots, N\}$ is the set of nodes (representing the agents), $\mathcal{E}(t) \subset \mathcal{N} \times \mathcal{N}$ denotes the set of directed arcs (representing information flow links between agents) at time t, and the matrix $\mathcal{W}(t) \in \mathbb{R}^{N\times N}$ is the adjacency matrix. In other words, the i^{th} node/vertex in the graph corresponds to agent $i \in \mathcal{N}$, whereas an arc $(i, j) \in \mathcal{E}(t)$ represents a directed information flow link from agent i to agent j at time t. Agent i is said to be connected to agent j if there is a sequence of arcs $(i_1, i_2), (i_2, i_3), \ldots, (i_{p-1}, i_p)$ such that $i = i_1$ and $j = i_p$. The adjacency matrix $\mathcal{W}(t) = [w_{ij}(t)]$ represents the weights of the information flow links between the agents and $(i, j) \in \mathcal{E}(t)$ if and only if $w_{ij}(t) \geq \underline{w}$ at time t for some lower bound $\underline{w} > 0$. A directed graph is called a directed tree if every node, except the root, has exactly one incoming arc. If the tree connects all the nodes of the graph, then it is called a spanning tree. These definitions hold for static neighborhood topology and can describe also instantaneous connectivity in a dynamic neighborhood topology. However, in a dynamic topology, connectivity over time is more important than the instantaneous connectivity. To define connectivity over time consider an interval $\mathcal{I}$ with length I (i.e., $I = |\mathcal{I}|$ and as in [6], [11] define $\bar{\mathcal{W}}(\mathcal{I}) = \frac{1}{I}\int_{\mathcal{I}} \mathcal{W}(\tau)d\tau$ with elements $\bar{w}_{ij}(\mathcal{I}), 1 \leq i, j \leq N$ and let $\bar{\mathcal{E}}(\mathcal{I})$ be the arc set such that $(i, j) \in \bar{\mathcal{E}}(\mathcal{I})$ if and only if $w_{ij}(\mathcal{I}) \geq \underline{w}$. Then, the graph $\mathcal{G}(t)$ is said to have a spanning tree over an interval $\mathcal{I}$ if the graph defined as $\mathcal{G}(\mathcal{I}) = (\mathcal{N}, \mathcal{E}(\mathcal{I}), \mathcal{W}(\mathcal{I}))$ has a spanning tree. Furthermore, it is said that $\mathcal{G}(t)$ uniformly has a spanning tree if there is an I such that for every interval $\mathcal{I}$ of length I the corresponding $\mathcal{G}(\mathcal{I})$ has a spanning tree. Note that these are continuous time counterparts of the discrete time definitions in [4], [5]. Now, we have the following assumption.

Assumption 1: There exists a constant $I \geq 0$ such that for every interval $\mathcal{I}$ of length I the corresponding agent neighborhood graph $\mathcal{G}(\mathcal{I})$ has a spanning tree.

Note that in the above setting it is assumed that the number of inputs and outputs of the agents are the same, i.e., $u_i, y_i \in \mathbb{R}^m$ for all i. $\mathbb{R}^m$ is the space in which the output trajectories of the agents evolve. Similarly, it is assumed that the number of internal states and relative degrees of the agents are also the same, i.e., the same n and r for all agents. However, the latter assumption can easily be relaxed allowing inclusion of agents with different number of internal states and relative degrees.

III. CONTROLLER DESIGN

The agent dynamics in (2) are feedback linearizable and one can immediately see that the controllers

$$u_i^\star = [\beta_i(x_i)]^{-1}\Big[-[\alpha_{ki}(t) + \alpha_i(x_i)] + \nu_i(t)\Big], i = 1, ...N, \tag{4}$$

where $\nu_i(t)$ is an outer controller convert the system into the form

$$\begin{aligned} \dot{z}_i &= f_{oi}(z_i, \xi_i) \\ \dot{\xi}_{i,1} &= \xi_{i,2} \\ &\cdots \\ \dot{\xi}_{i,r-1} &= \xi_{i,r} \\ \dot{\xi}_{i,r} &= \nu_i(t) \end{aligned} \tag{5}$$

In other words, provided that the functions $\alpha_{ki}(t)$, $\alpha_i(x_i)$, and $\beta_i(x_i)$ are known, with the use of the feedback linearizing controllers in (4), the problem is reduced to designing $\nu_i(t)$ for the systems in (5) such that to achieve agreement. However, the controller $u_i^\star$ in (4) is not implementable since $\beta_i(x_i)$ and $\alpha_i(x_i)$ are unknown. Still, the knowledge that such $u_i^\star$ exists allows us to take a direct adaptive control approach and approximate it using an adaptive fuzzy system.

By assumption the agents have access to information only from their neighbors $N_i(t)$ and that is the only information they can use in their controllers. Since the problem is to achieve output agreement let us define the output error for each agent i as

$$e_i(t) = -\sum_{j\in N_i(t)} w_{ij}(t)(y_i(t)-y_j(t)) = -\sum_{j\in N_i(t)} l_{ij}(t)y_j(t) \tag{6}$$

where

$$l_{ij}(t) = \begin{cases} -w_{ij}(t), & i\neq j, \\ \sum_{j\in N_i(t)} w_{ij}(t), & i=j, \end{cases}$$

is the corresponding entry of the Laplacian matrix $\mathcal{L}(t)$ of the neighborhood graph $\mathcal{G}(t)$. Note that $e_i(t)$ are measurable variables and with their myopic local sensing the objective of every agent is to drive its $e_i(t)$ to zero. In order to achieve this, the control inputs u_i of the agents are defined in the form

$$u_i = \hat{u}_{ai} + u_{si} + u_{bi},\ 1\leq i\leq N, \tag{7}$$

where $\hat{u}_{ai}$ represent an adaptive control term, u_{si} represents a sliding mode control term, and u_{bi} represents a bounding control term. Here, as in [18], we augment the controller with bounding and sliding mode control terms in order to guarantee robustness. Below, we will discuss each of the control terms in (7).

A. Adaptive Control Term

The objective of the adaptive control term $\hat{u}_{ai}$ in (7) is to approximate the unknown ideal controller $u_i^\star$ in (4) using a fuzzy system. For that purpose we use a Takagi-Sugeno type fuzzy system [19], [20]. The parameters of the fuzzy systems are updated such that to force the output errors $e_i(t)$ to converge to a small neighborhood of zero.

A fuzzy system is a rule-based system which has the ability to approximate any smooth function arbitrarily closely in a compact set. The rule base is composed of IF-THEN type of rules of the form

$\mathcal{R}_{ij}$: If s_{i1} is L_{i1}^k and ... and s_{ip} is L_{ip}^l Then
$c_{ij} = a_{j,0}^i + a_{j,1}^i\theta_1^i(S_i) + ... + a_{j,M_i-1}^i\theta_{M_i-1}^i(S_i)$

where i denotes the agent number and j denotes the rule number. The vector $s_i = [s_{i1}, s_{i2}, \ldots, s_{ip}]^\top$ is the input vector for the fuzzy system of agent i and c_{ij} is the output of its j^{th} rule, $j=1,...,R_i$. For every input s_{iq} there is a set of corresponding possible linguistic variables (fuzzy sets) $L_{iq}^1,...,L_{iq}^{q_i}$ which can be expressed in the form

$$L_{iq}^k = (s_{iq}, \mu_{L_{iq}^k}(s_{iq})):\ s_{iq}\in\mathbb{R},\ 1\leq i\leq N$$

where $k=1,...,q_i$ denotes the label of the linguistic value or basically denotes the k'th fuzzy set of the q'th input of agent i and $\mu_{L_{iq}^k}(s_{iq})$ is the corresponding membership function. The membership function $\mu_{L_{iq}^k}(s_{iq})$ specifies how much the input s_{iq} belongs to the fuzzy set L_{iq}^k. The functions $\theta_1^i(S_i),...,\theta_{M_i-1}^i(S_i)$ are input functions which are Lipschitz continuous and S_i represents the variables of interest which include the state of the agents x_i, the output error e_i, and the derivatives of e_i.

Once the outputs of the rules c_{ij} are calculated they need to be combined in order to form the output of the fuzzy system. There are various methods which can be used for that purpose. Here we use the center-average method

$$\hat{y}_i = \frac{\sum_{j=1}^{R_i} c_{ij}\mu_{ij}}{\sum_{j=1}^{R_i}\mu_{ij}},\ 1\leq i\leq N$$

where μ_{ij} is the output membership function for rule $\mathcal{R}_{ij}$ which can be calculated using the product operator as

$$\mu_{ij} = \mu_{L_{i1}^k}(s_{i1})\times\cdots\times\mu_{L_{ip}^l}(s_{ip})$$

Defining the vectors c_i and ζ_i as $c_i = [c_{i1},...,c_{iR_i}]^\top$ and $\zeta_i^\top = [\mu_{i1},...,\mu_{iR_i}]/[\sum_{j=1}^{R_i}\mu_{ij}]$, the output of the fuzzy system for agent i (which is its control input $\hat{u}_{ai}$) can be written in the form

$$\hat{u}_{ai} = \hat{y}_i = c_i^\top\zeta_i$$

Moreover, note that defining z_i as

$$z_i = [1, \theta_1^i(S_i), ..., \theta_{M_i-1}^i(S_i)]^\top \in \mathbb{R}^{M_i}$$

and $A_i\in\mathbb{R}^{R_i}\times\mathbb{R}^{M_i}$ as

$$A_i^\top := \begin{bmatrix} a_{1,0}^i & a_{1,1}^i & \cdots & a_{1,M_i-1}^i \\ a_{2,0}^i & a_{2,1}^i & \cdots & a_{2,M_i-1}^i \\ \vdots & \vdots & \ddots & \vdots \\ a_{R_i,0}^i & a_{R_i,1}^i & \cdots & a_{R_i,M_i-1}^i \end{bmatrix}$$

one can express the vectors c_i as

$$c_i^\top = z_i^\top A_i$$

Then, the control input $\hat{u}_{ai}$ of agent i becomes

$$\hat{u}_{ai} = z_i^\top A_i\zeta_i \tag{8}$$

which is a linearly parameterized system. Its parameters are updated using

$$\dot{A}_i = Q_i^{-1}z_i\zeta_i^\top e_{si} \tag{9}$$

where $Q_i\in\mathbb{R}^{M_i}\times\mathbb{R}^{M_i}$ are positive definite diagonal matrices and e_{si} are calculated as

$$e_{si} = k_{i0}e_i + k_{i1}\dot{e}_i + \ldots + k_{i,r-2}e_i^{(r_i-2)} + e_i^{(r_i-1)}$$

The coefficients $k_i = [k_{i0},...,k_{i,r-2},\ 1]^\top$ are chosen such that the corresponding polynomials $T_i(s) = s^{r-1} + k_{i,r-2}s^{r-2} + ... + k_{i1}s + k_{i0}$ are Hurwitz.

The update law in (9) depends on e_{si} and as it converges (i.e., the output error e_i and its derivatives converge) to zero the update stops. However, during the process the elements of the parameter matrix A_i can become very large. Therefore, it is good to project the elements of A_i to some compact set. This can be useful especially in the case in which some bounds on the entries of A_i are known a priori. Consider a set $\Omega_i = [A_i^{min}, A_i^{max}]^{R_i\times M_i}$ and assume that there is a need to guarantee that $A_i\in\Omega_i$. In other words, it is required that the elements of A_i are within $[A_i^{min}, A_i^{max}]$. To achieve this one can project the values of the elements of A_i within that set. One possible strategy for that, which was also used in [18], is instead of (9) to use an update law of the form

$$\dot{A}_i = Q_i^{-1}\hat{A}_i \tag{10}$$

where $\hat{A}_i$ is a matrix the elements of which are determined as

$$\hat{a}^i_{p,m} = \left\{ \begin{array}{ll} 0, & \text{if } a^i_{p,m} \notin (A_i^{min}, A_i^{max}) \text{ and} \\ & \bar{a}^i_{p,m}(a^i_{p,m} - a^{ic}_{p,m}) > 0 \\ \bar{a}^i_{p,m}, & \text{otherwise} \end{array} \right\} \tag{11}$$

where $\bar{a}^i_{p,m}$ is the (p,m)'th element of $z_i\zeta_i^\top e_{si}$ in (9) and $a^{ic}_{p,m}$ is an element of a fixed matrix $A_i^c \in \Omega_i$ (or basically $a^{ic}_{p,m} \in [A_i^{min}, A_i^{max}]$), and $p = 1, ..., R_i$ and $m = 1, ..., M_i$. Note that the update law in (10)-(11) sets the value of $\hat{a}^i_{p,m} = 0$ if its corresponding parameter $a^i_{p,m}$ is outside of the allowed region (A_i^{min}, A_i^{max}) and the calculated value of $\bar{a}^i_{p,m}$ is such that the update will be in the direction of further divergence from (A_i^{min}, A_i^{max}). In the case in which the $a^i_{p,m}$ is inside of the allowed region (A_i^{min}, A_i^{max}) or the calculated value of $\bar{a}^i_{p,m}$ is such that the update will be in the direction towards (A_i^{min}, A_i^{max}) the update is performed with $\hat{a}^i_{p,m} = \bar{a}^i_{p,m}$. This update rule saturates the fuzzy system parameters so that they remain within the sets Ω_i.

B. Bounding Control Term

The bounding control term u_{bi} is a robust component of the control input which is used to bound e_{si} (the output errors and their derivatives). As in [18], we define u_{bi} as

$$u_{bi} = \Pi_i(e_{si})k_{bi}(t)\text{sign}(e_{si}) \tag{12}$$

where the time-varying coefficient $k_{bi}(t)$ is calculated as

$$k_{bi}(t) = |\hat{u}_{ai}| + |u_{si}| + \frac{\bar{\alpha}_i(x_i) + |\nu_i|}{\underline{\beta_i}} \tag{13}$$

In this equation $\bar{\alpha}_i(x_i)$ is a known bound on $\alpha_i(x_i)$ satisfying $\bar{\alpha}_i(x_i) \geq |\alpha_i(x_i)|$ and linear inputs $\nu_i(t)$ are calculated as

$$\nu_i(t) = \eta_i e_{si} + \bar{e}_{si} + \alpha_{ki}(t)$$

where η_i is a constant and $\bar{e}_{si} = [\dot{e}_{0i}, ..., e_{0i}^{(r-1)}][k_{i0}, ..., k_{i,r-2}]^\top$. Note also that the operations sign and $|\cdot|$ in the above equations are performed elementwise on their input vectors. The $\Pi_i(e_{si})$ function is a saturation function with dead band whose j'th element $\Pi_i^j(t)$, which is plotted in Figure 1, can be expressed as

$$\Pi_i^j(t) = \left(\begin{array}{ll} 1, & \text{if } M_e \leq |e^j_{si}| \\ \frac{|e^j_{si}| + \epsilon_M - M_e}{\epsilon_M}, & \text{if } M_e - \epsilon_M \leq |e^j_{si}| < M_e \\ 0, & \text{otherwise} \end{array} \right)$$

where the constant M_e as the bound for e_{si} and ϵ_M is a constant such that $0 < \epsilon_M \leq M_e$.

Note that the bounding control term u_{bi} is activated when $|e_{si}| \geq M_e$. Beside preventing unboundedness of e_{si} it also guarantees that the fuzzy system performs its approximation objective in a compact set.

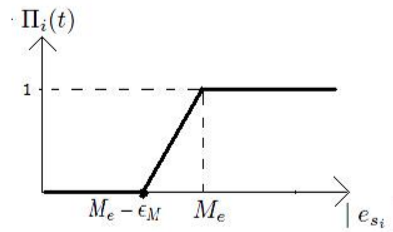


Fig. 1. $\Pi_i(t)$ for the bounding control term.

C. Sliding Mode Control Term

The fuzzy systems, and therefore the adaptive term u_{ai}, always contain approximation errors. Let $D_{ui}(x_i)$ be a known upper bound for the approximation error between u_{ai} in (8) and $u_i^\star$ in (4). In other words, defining $d_{ui}(x_i) = u_i^\star - \hat{u}_{ai}$ let $D_{ui}(x_i)$ be such that $|d_{ui}(x_i)| \leq D_{ui}(x_i)$. Then, one can define a sliding mode term as

$$u_{si} = k_{si}(t)\text{sign}(e_{si}) \tag{14}$$

where

$$k_{si}(t) = \frac{B_i(x_i)|e_{si}|}{2\underline{\beta_i}^2} + D_{ui}(x_i) \tag{15}$$

This term is effective in suppressing the approximation errors and guaranteeing asymptotic stability. Note that the operation of this term is not directly like conventional sliding mode controllers and it does not force the error to slide on a sliding surface.

With the controller in (7) composed of adaptive, bounding, and sliding mode control terms the output errors e_i will converge to zero for all agents i. Provided that the neighborhood graph uniformly has a spanning tree this will result in distributed agreement of the agent outputs. In the following section we will provide numerical simulations to verify the effectiveness of the proposed method.

IV. Simulation Results

Consider multi-agent system consisting of agents with dynamics

$$\begin{aligned} \dot{\xi}_{i,1} &= \xi_{i,2} \\ \dot{\xi}_{i,2} &= \xi_{i,3} \\ \dot{\xi}_{i,3} &= \alpha_i + \beta_i u_i \\ y_i &= x_i \end{aligned} \tag{16}$$

where the state of agent i is $x_i = [\xi_{i,1}, \xi_{i,2}, \xi_{i,3}]^\top \in \mathbb{R}^6$ ($n = 6$), $y_i = \xi_{i,1} \in \mathbb{R}^2$ is its output ($m = 2$), and $u_i \in \mathbb{R}^2$ is its control input. The values/expressions of α_i and β_i are assumed to be unknown. In the simulations we utilize periodic α_i and constant β_i for all agents. In particular, we use $\alpha_i = [(i - p_{i1}), (i - p_{i2})]^\top \cos(3t)$ and $\beta_i = diag\{p_{i3}, p_{i4}\}$, where $diag$ represents a diagonal matrix with the specified entries in the diagonal and $\{p_{i1}, p_{i2}, p_{i3}, p_{i4}\}$ are unknown uniformly distributed randomly generated numbers such that $p_{i1}, p_{i2} \in [0, 3]$ and $p_{i3}, p_{i4} \in [1, 3]$. As can be seen the agent dynamics are fully linearizable (i.e., no zero dynamics) and the relative degree of the agent dynamics is $\{r, r\} = \{3, 3\}$ for all agents.

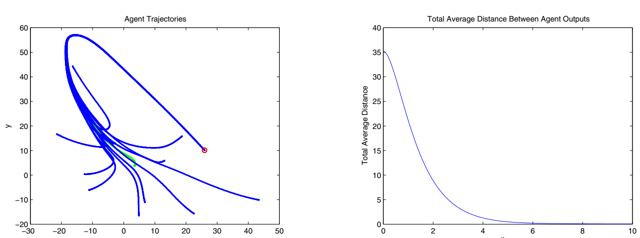
We perform simulations with $N = 11$ agents. For the adaptive control terms for all agents we use fuzzy systems consisting of $R = 9$ gaussian input membership functions uniformly distributed within $[-10,\ 10]$ such that the rightmost and the leftmost membership functions are centered at 10 and -10, respectively and are saturated and extended. The spreads of the membership functions are chosen equal and are given by $20/(R-1) = 2.5$. The input of the fuzzy systems are the local output errors e_i for each agent. The rule base of the fuzzy systems for the agents are the same and are in the form

R_{ij}: If e_i is L_j Then $c_{ij} = a^i_{j,0} + a^i_{j,1}\nu_i(t)$ where $j = 1, ..., R_i = 9$ represents the rule number (there is only one input and each input membership function corresponds to a rule) and $i = 1, ..., N$ represents the agent number. As can be seen we have $M_i = 2$ the output of the fuzzy system is determined by $z_i = [1, \nu_i(t)]^\top$. The initial values for the fuzzy system parameters $a^i_{p,m}(0)$ are generated randomly with a normal distribution with zero mean and variance 5 for all $i = 1, ..., N$, $p = 1, ..., R_i$ and $m = 1, ..., M_i$. The positive definite diagonal matrix $Q_i \in \mathbb{R}^{2\times 2}$ used for parameter update in (10) is chosen as $Q_i = I$, where I is the identity matrix.

The variables e_{si} is calculated as $e_{si} = [k_{0i}, k_{1i}, 1][e_i, \dot{e}_i, \ddot{e}_i]^\top$ where $[k_{0i}, k_{1i}, 1]^\top = [2, 3, 1]^\top$ (resulting in Hurwitz polynomial with poles $\{-2, -1\}$) for all $i = 1, ..., N$. For the simulations below we do not employ the parameter projection in (10)-(11). The bounds for e_{si} are defined as $M_e = 50$ for all agents i and $\epsilon_M = 1$. For the given system, as the bounds in (13) we use $\bar{\alpha}_i = N$, $B(x) = 0$, and $\underline{\beta_i} = 1$ for all agents i. Moreover, we have $\eta_i = 1$, $\bar{e}_{si} = [\dot{e}_i, \ddot{e}_i][k_{0i}, k_{1i}]$ and $\nu_i(t)$ is calculated as $\nu_i(t) = \eta_i e_{si} + \bar{e}_{si}$.

Since we have $B(x) = 0$ the $k_{si}(t)$ gain in (15) can be expressed as $k_{si}(t) = D_{ui}(x)$. We used $D_{ui}(x) = 1$ for all agents throughout the simulations. The initial states of the agents are chosen such that $\xi_{i,2} = \xi_{i,3} = [0, 0]^\top$ and $\xi_{i,1}$ are generated randomly with uniform distribution within $[0,\ 20]^2$. In the adjacency matrix we use either $w_{ij} = 1$ or $w_{ij} = 0$ to specify the neighborhood connections. In other words, if agent j is a neighbor of agent i (i.e., $j \in N_i(t)$), then we have $w_{ij} = 1$, otherwise we have $w_{ij} = 0$.

Figure 2 shows a simulation for a fully connected neighborhood topology (i.e., $w_{ij} = 1$ for all i and j). The paths

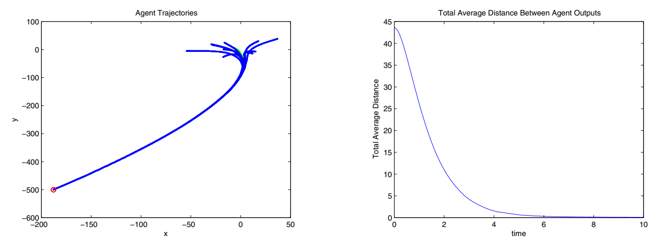


Agent outputs Average distance

Fig. 2. Results for fully connected neighborhood topology ($N = 11$).

of the agents are shown in Figure 2(a), whereas the average distance between the outputs of the agents with respect to time is presented in Figure 2(b). As can be seen from the figure the average distance between agent outputs converges to zero and the agents are able to achieve output agreement. In the fully connected neighborhood topology every agent can obtain information from every other agent in the swarm. Therefore, it can be considered as the simplest case for testing the algorithm.

Figure 3 shows another simulation in which the neighborhood topology is a unidirectional ring. In this topology every

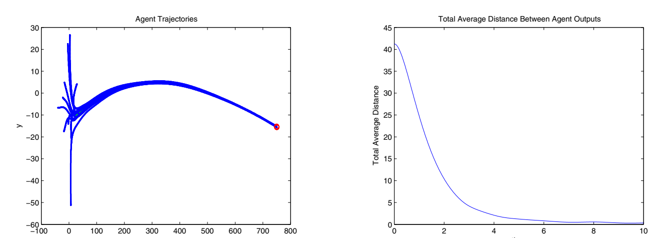


Agent outputs Average distance

Fig. 3. Results for unidirectional ring neighborhood topology ($N = 11$).

agent i can receive information only from its preceding agent $i-1$ and agent 1 can receive information from agent N. In other words, for $i = 1$ we have $w_{iN} = 1$, for $i = 2, ...N$ we have $w_{ij} = 1$ if $j = i - 1$, and all other $w_{ij} = 0$. The swarm with this topology is in a sense in cyclic pursuit [23]. As in the previous case the paths of the agents are shown in Figure 3(a), whereas the average distance between the outputs of the agents with respect to time is presented in Figure 3(b). One can see that in this case as well the average distance between agent outputs converges to zero and again the agents are able to achieve output agreement.

The next simulation shown in Figure 4 is for random neighborhood topology. For this case the neighbors of the

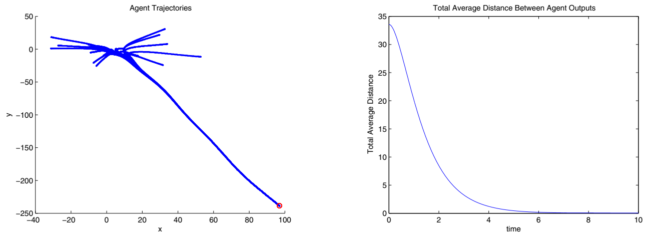


Agent outputs Average distance

Fig. 4. Results for random neighborhood topology ($N = 11$).

agents are assigned randomly with uniform probability of 0.7 at the beginning of the simulation (i.e., the values of $w_{ij} = 1$ or $w_{ij} = 0$ are determined randomly with corresponding probabilities of 0.7 and 0.3, respectively). Assumption 1 is satisfied for the presented simulation. As one can observe once more the agents are able to achieve agreement despite the uncertainties in the agent dynamics and using only local information.

Figure 5 shows another simulation with random neighborhood topology. For this simulation the probability for neighbor

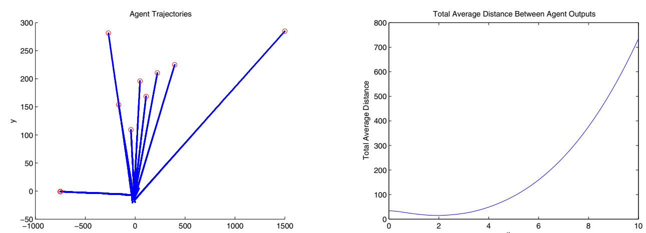


Agent outputs   Average distance

Fig. 5. Results for random neighborhood topology ($N = 21$).

assignment was decreased to 0.5 and the number of agents was increased to $N = 21$. The connectivity requirement in Assumption 1 is still satisfied. As one can see increasing the number of agents does not result in qualitative change in the distributed output agreement performance of the swarm. In fact the results are independent of the number of agents and hold for any finite N.

The last simulation shown in Figure 6 presents a simulation for a case in which the connectivity requirement in Assumption 1 is not satisfied. The number of agents in this simulation is $N = 11$ similar to most of the preceding cases. As one easily notice from the figure the agents are unable to achieve agreement of their outputs. This shows how crucial is satisfaction of Assumption 1. In case Assumption 1 is not satisfied, then there are subgroups in the swarm between which there is no information exchange. Therefore, not matter how successful are the local controllers to drive the agent output errors e_i in (6) to zero agreement of the outputs is not guaranteed.



Agent outputs   Average distance

Fig. 6. Results for topology which does not satisfy Assumption 1 ($N = 11$).

V. Concluding Remarks

In this paper we developed a direct adaptive fuzzy controller for achieving distributed output agreement in a class of multi-agent dynamic systems with uncertain agent dynamics. Inspired by the work in [18], we use linearly parameterized Takagi-Sugeno fuzzy systems in the adaptive part of the controller. In order to achieve robustness and to guarantee boundedness of the output errors and to achieve their asymptotic convergence to zero, we also augment the controller with bounding and sliding mode terms. Under the proposed strategy the agents are successfully able to achieve output agreement despite the model uncertainties and/or disturbances in the agent dynamics. The presented numerical simulations verify the effectiveness of the procedure.

References

[1] V. Gazi and B. Fidan, "Coordination and control of multi-agent dynamic systems: Models and approaches," in *Proceedings of the SAB06 Workshop on Swarm Robotics*, ser. Lecture Notes in Computer Science (LNCS) 4433, E. Sahin, W. M. Spears, and A. F. T. Winfield, Eds. Berlin Heidelberg: Springer-Verlag, 2007, pp. 71–102.

[2] V. Gazi and K. M. Passino, *Swarm Stability and Optimization*. Heidelberg, Germany: Springer-Verlag, January 2011.

[3] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, June 2003.

[4] W. Ren and R. W. Beard, "Consensus seeking in multi-agent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[5] V. Gazi, "Stability of a discrete-time asynchronous swarm with time-dependent communication links," *IEEE Transactions on Systems, Man, and Cybernetics : Part B*, vol. 38, no. 1, pp. 267–274, February 2008.

[6] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, February 2005.

[7] 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, September 2004.

[8] Z. Lin, B. Francis, and M. Maggiore, "Necessary and sufficient graphial conditions for formation control of unicycles," *IEEE Transactions on Automatic Control*, vol. 50, no. 1, pp. 121–127, January 2005.

[9] P. Wieland and F. Allgöwer, "An internal model principle for consensus in heterogeneous linear multi-agent systems," in *1st IFAC Workshop on Estimation and Control in Networked Systems*, Venice, Italy, September 2009, pp. 7–12.

[10] ——, "An internal model principle for synchronization," in *IEEE International Conference on Control and Automation (ICCA)*, Christchurch, New Zealand, December 2009, pp. 285–290.

[11] P. Wieland, R. Sepulchre, and F. Allgöwer, "An internal model principle is necessary and sufficient for linear output synchronization," *Automatica*, vol. 47, no. 5, pp. 1068–1074, 2011.

[12] J. Xiang, W. Wei, and Y. Li, "Synchronized output regulation of linear networked systems," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1336–1341, June 2009.

[13] H. Kim, H. Shim, and J. H. Seo, "Output consensus of heterogeneous uncertain linear multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 1, pp. 200–206, January 2011.

[14] V. Gazi, "Distributed output agreement in a class of uncertain linear heterogeneous multi-agent systems," in *IEEE Conference on Decision and Control*, 2013, preprint, submitted for publication.

[15] C. Y. Su and Y. Stepanenko, "Adaptive control of a class of nonlinear systems with fuzzy logic," *IEEE Transactions on Fuzzy Systems*, vol. 2, no. 4, pp. 285–294, November 1994.

[16] F. C. Chen and H. K. Khalil, "Adaptive control of nonlinear systems using neural networks," *International Journal of Control*, vol. 55, pp. 1299–1317, 1992.

[17] Y. Diao and K. M. Passino, "Adaptive neural/fuzzy control for interpolated nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 5, pp. 583–595, October 2002.

[18] J. T. Spooner and K. M. Passino, "Stable adaptive control using fuzzy systems and neural networks," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 339–359, August 1996.

[19] J. T. Spooner, M. Maggiore, R. Ordóñez, and K. M. Passino, *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques*. New York: John Wiley and Sons, Inc., 2002.

[20] K. M. Passino and S. Yurkovich, *Fuzzy Control*. Addison-Wesley Longman, Menlo Park, CA, 1998.

[21] S. Duran and V. Gazi, "Adaptive formation control and target tracking in a class of multi-agent systems," in *American Control Conference*, Baltimore, MD, USA, June-July 2010, pp. 75–80.

[22] S. Güler, N. Köksal, B. Fidan, and V. Gazi, "Indirect adaptive formation control with nonlinear dynamics and parametric uncertainty," in *Asian Control Conference*, Istanbul, Turkey, June 2013.

[23] A. T. Şamiloglu, V. Gazi, and A. B. Koku, "Asynchronous cyclic pursuit," in *Proc. of 9'th Conference on Simulation of Adaptive Behavior (SAB06)*, ser. Lecture Notes in Artificial Intelligence (LNAI) 4095, S. N. et al., Ed. Springer Verlag, September 2006, pp. 667–678.