Control Theory Tech, Vol. 16, No. 2, pp. 110–121, May 2018

Control Theory and Technology

http://link.springer.com/journal/11768

Control Theory and Technology

Consensus controller for multi-UAV navigation

Patrik KOLARIC [1†], Ci CHEN [2,1], Ankur DALAL [1], Frank L. LEWIS [1,3]

1.*UTA Research Institute, University of Texas at Arlington, 7300 Jack Newell Blvd S, Fort Worth, TX, 76118, U.S.A.;*

2.*School of Automation, Guangdong University of Technology; Guangdong Key Laboratory of IoT Information Technology, Guangzhou Guangzhou 510006, China*

3.*State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang Liaoning 110819, China*

Received 20 January 2018; revised 28 March 2018; accepted 28 March 2018

Abstract

In this paper, we design consensus algorithms for multiple unmanned aerial vehicles (UAV). We mainly focus on the control design in the face of measurement noise and propose a position consensus controller based on the sliding mode control by using the distributed UAV information. Within the framework of Lyapunov theory, it is shown that all signals in the closed-loop multi-UAV systems are stabilized by the proposed algorithm, while consensus errors are uniformly ultimately bounded. Moreover, for each local UAV, we propose a mechanism to define the trustworthiness, based on which the edge weights are tuned to eliminate negative influence from stubborn agents or agents exposed to extremely noisy measurement. Finally, we develop software for a nano UAV platform, based on which we implement our algorithms to address measurement noises in UAV flight tests. The experimental results validate the effectiveness of the proposed algorithms.

Keywords: Consensus control, multi-agent system, quadrotor, Lyapunov stability, distributed system

DOI https://doi.org/10.1007/s11768-018-8013-5

1 Introduction

Cooperative control of multiple unmanned aerial vehicles (UAVs) that share a global task has recently drawn a lot of attention. The last decades have witnessed a rapid development of micro UAVs and the idea of multi-agent UAV systems performing tasks grew with it. This is due to the fact that a group of simple individual vehicles may facilitate the execution of a complex task with higher robustness to potential technical faults. Moreover, multi-agent systems also have an economical and logistical advantage, simply because the failure of a single agent does not put the whole mission into jeopardy. That makes a single agent in the system expendable and

[†] Corresponding author.

E-mail: patrik.kolaric@mavs.uta.edu.

This work was supported in part by the National Natural Science Foundation of China (No. 61633007, 61703112), in part by the China Postdoctoral Science Foundation (No. 2016M600643) and the special fund (No. 2017T100618), and in part by the Office of Naval Research (No. N00014-17-1-2239, N00014-18-1-2221).

its design requirements less rigorous. Those advantages are widely recognized in research literature [1–3].

We are interested in the consensus control of multiple UAVs, especially quadcopters. It is shown in [4] that a two-loop nonlinear scheme using input-output linearisation technique can be employed to satisfy the tracking performance. Moreover, a neural network based backstepping design for UAV is provided in [5], where an inverse kinematics solution is applied akin to that in the manipulator control such as [6]. Nonlinear models of UAV and linearization procedure is covered in [7, 8]. New control techniques have been proposed to allow multiple vehicles to work cooperatively (see [9–14]). Specific problem of UAV formation control in terms of position and attitude was addressed in [15–17]. Finally, more intricate nonlinear approach to control design for multi-agent system was covered in [18, 19].

More recently, further work is considered to quantify the performances of the whole multi-UAV system and the local UAV. Its formulation falls into a category of cooperative optimal control and game theory. In particular, reference [20] uses linear quadratic regulator based optimal control to achieve the synchronization of linear systems. Non-zero-sum games are studied for multiple players in [21–23], where the adaptive algorithm learns online the solution of coupled Riccati equations and coupled Hamilton-Jacobi equations. Even though good results were obtained, little research has been carried out to handle the measurement noise imposed on each local UAV.

Note that when the noise is injected into the measurement signals of each local UAV, the imperfect information propagates through distributed control protocols, which makes the multi-UAV system vulnerable to uncertainties and thus severely degrades the cooperative performance. Due to the fact that this paper considers actual implementation of the system in the lab environment, it is of great importance to treat the poor information quality in the control law accordingly. To address that problem, [24] proposed an adaptive stabilization control for cyber-physical systems against measurement noises. Moreover, a distributed adaptive leaderless control for multi-agent systems was proposed in [25]. However, current results in [24,25] require that the measurement noises are restricted to certain simple types. More importantly, these papers only theoretically analyzed the measurement noise problem in the cooperative control, but few results have been experimentally validated.

In this paper, we aim to design the cooperative control of multiple UAVs under measurement noises with theoretical analysis and physical experiments. To do this, we first model the dynamics of each individual UAV and its measurement noises. Based on the sliding mode control design, we propose a distributed control protocol that stabilizes all signals in the closed-loop multi-UAV systems and ensures that the consensus errors are uniformly ultimately bounded. Moreover, we use local information to define the trustworthiness [26–28], based on which we tune the edge weights to compensate for the negative effect from noisy measurements. In addition, we bridge the gap between the theoretical framework and the physical experiment and implement our results by using Crazyflie 2.0 platform. In our experiments, we utilize the motion capture system (VICON) to coordinate UAVs through the communication topology based on a master-slave model. Furthermore, a hierarchical motion control state machine is designed in a distributed fashion and implemented as a software solution based on robot operating system environment (ROS). To ensure low latency and smooth control loop, data pipeline is implemented as a multi-threaded routine specifically tailored for the high-speed control requirements of dynamically rich UAVs. Several experiments are designed to validate the effectiveness of our algorithms.

The remaining parts of this paper are outlined as follows. In Section 2, some preliminaries on graph theory are provided. In Section 3, the dynamics of UAV are detailed. In Section 4, the controller for coordinating UAVs is proposed, and its stability analysis is carried out. In Section 5, the trustworthiness based approach is proposed to tune the edge weight. The experimental environment is detailed in Section 6, and the experimental results are given in Section 7. Finally, the conclusion is contained in Section 8.

2 Preliminaries on graph theory

In this section, we briefly give some preliminaries on graph theory that will be used in our analysis and experiments.

Definition 1 Let graph G be the pair

$$G = (V, E), \tag{1}$$

where V denotes vertices (nodes) of a graph and E denotes the edges of a graph. In that context V can be represented as

$$V = \{v_1, v_2, \dots, v_N\}, \tag{2}$$

where v_i is vertex (node) i of a graph. Nodes are connected by links (edges). Edge E is defined as

$$E = (v_i, v_j), \ \ \forall i, j \in [0, N], \ i \neq j, \tag{3}$$

which is a connection between two nodes i and j.

If edges are not directed (which means that they are bidirectional), then the graph is said to be undirected. Each edge has a weight that can model the strength of the connection between the nodes. The number of edges that enter a node is called in-degree, while the number of edges that exit a node is called out-degree. The graph can be presented in the form of a matrix where column index corresponds to a source of an edge and row index to a sink of an edge. Matrix element a_{ij} indicates the weight of the corresponding edge in the graph. Matrix constructed of elements a_{ij} is called adjacency matrix $\mathcal{A}$. $\mathcal{A}$ is very convenient as it enables analysis of graph theory from the perspective of linear algebra. Structural properties of the graph are embedded in matrix $\mathcal{A}$ and can be extracted by examining that matrix. $d_i = \sum\limits_{i=1}^{n} a_{ij}$ is a row sum of $\mathcal{A}$ and it is introduced for convenience in the future analysis. Matrix $\mathcal{D}$ is diagonal matrix of in-degrees, $\mathcal{D} = \mathrm{diag}\{d_i\}$. Therefore, the Laplacian matrix is formulated as $\mathcal{L} = \mathcal{D} - \mathcal{A}$.

3 Quadrotor dynamics

When dealing with UAVs, states used for expressing dynamics are position ξ_i, velocity $\dot{\xi}_i$, attitude η_i and angular rate $\dot{\eta}_i$. Quadrotor dynamics are expressed as follows

$$m\ddot{\xi}_i = F_{\mathrm{g}} + F_{\mathrm{d}}(\eta_i), \tag{4}$$

$$J(\eta_i)\ddot{\eta}_i = C(\eta_i, \dot{\eta}_i) + \tau_i, \tag{5}$$

where F_{g} is a gravitational force, and F_{d} is a nonlinear input that depends on the attitude, η are rotational states (roll, pitch and yaw), and τ is the torque generated by propellers.

Note that the focus of this paper is on efficiency of the cooperative control policy rather than the design of effective dynamical controller for a single UAV. Without the loss of generality, certain simplifications are used. UAVs that we worked with are very lightweight (27 g) which permits us to linearize the dynamical system. Consensus controller will be tested only in the x-y plane. After the linearization around the equilibrium point (similar to [29]), systems (4) and (5) can be reduced to the following form:

$$\ddot{x}_i = \begin{bmatrix} \ddot{\xi}_{x,i} \\ \ddot{\xi}_{y,i} \\ \ddot{\xi}_{z,i} \\ \ddot{\eta}_{r,i} \\ \ddot{\eta}_{p,i} \\ \ddot{\eta}_{y,i} \end{bmatrix} = \begin{bmatrix} -g\eta_{p,i} \\ g\eta_{r,i} \\ -f_{T,i}/m \\ \tau_{x,i}/I_x \\ \tau_{y,i}/I_y \\ \tau_{z,i}/I_z \end{bmatrix}, \tag{6}$$

where the system state x_i is given as

$$x_i = \begin{bmatrix} \xi_{x,i} \\ \xi_{y,i} \\ \xi_{z,i} \\ \eta_{r,i} \\ \eta_{p,i} \\ \eta_{y,i} \end{bmatrix}, \tag{7}$$

where I_x, I_y and I_z denote the idealized moments of inertia for each axis and m is the mass of the UAV. After separating the input terms from the state transition matrix, we can change (6) into

$$\ddot{x}_i = Ax_i + Bv_i, \tag{8}$$

where

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & -g & 0 \\ 0 & 0 & 0 & g & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{9}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\dfrac{1}{m} & 0 & 0 & 0 \\ 0 & \dfrac{1}{I_x} & 0 & 0 \\ 0 & 0 & \dfrac{1}{I_y} & 0 \\ 0 & 0 & 0 & \dfrac{1}{I_z} \end{bmatrix}, \tag{10}$$

and the control input is

$$v_i = \begin{bmatrix} f_{T,i} \\ \tau_{x,i} \\ \tau_{y,i} \\ \tau_{z,i} \end{bmatrix}. \tag{11}$$

Considering that there are N UAVs in this paper, we present the system dynamics in a global form as

$$\ddot{x} = (I_N \otimes A)x + (I_N \otimes B)v. \tag{12}$$

To model the group of agents with distributed information we use the graph theory. In the light of that, every link of the graph represents communication capability between connected agents, while nodes represent dynamical agents. The objective of this paper is to make the follower quadrotors in the graph $\mathcal{G}$ track the desired trajectory based on the distributed information, where the trajectory is generated by the leader.

4 Position consensus controller

We introduce Δ_i as a displacement of UAV i from the center of the formation. Different combinations of Δ_i can now be used to encode various formations. x_0 is the agent leader and is directly linked to those agents that have non-zero pinning gain g_i. The reference is generated by the leader and is effectively transmitted to the whole system through the distributed communication network.

The neighbourhood error is defined as

$$e_i = \sum_{j\in N_i} a_{ij}(x_j - \Delta_j - x_i + \Delta_i) + g_i(x_0 - x_i + \Delta_i), \tag{13}$$

and neighbourhood time derivative of the error

$$\dot{e}_i = \sum_{j\in N_i} a_{ij}(\dot{x}_j - \dot{\Delta}_j - \dot{x}_i + \dot{\Delta}_i) + g_i(\dot{x}_0 - \dot{x}_i + \dot{\Delta}_i). \tag{14}$$

If we assume that the position of the leader (x_0) and the relative positions between the agents ($\Delta_i, \forall i$) are not changing during the experiment then

$$\dot{e}_i = \sum_{j\in N_i} a_{ij}(\dot{x}_j - \dot{x}_i) + g_i(-\dot{x}_i). \tag{15}$$

The alias of ideal positions displaced by Δ_i is introduced

$$x_i^{\mathrm{a}} = x_i - \Delta_i, \tag{16}$$

for $i = 1, 2, \dots, N$. From (16), position errors between the two agents are given as

$$\delta_{ij} = x_j^{\mathrm{a}} - x_i^{\mathrm{a}}. \tag{17}$$

From (17), the neighbourhood error defined in (13) now has the form

$$e_i = \sum_{j\in N_i} a_{ij}(x_j^{\mathrm{a}} - x_i^{\mathrm{a}}) + g_i(x_0 - x_i^{\mathrm{a}}), \tag{18}$$

$$\dot{e}_i = \sum_{j\in N_i} a_{ij}(\dot{x}_j - \dot{x}_i) + g_i(-\dot{x}_i). \tag{19}$$

Define $\underline{1}_N = [1, 1, \dots, 1]^{\mathrm{T}}\mathbb{R}^N$ with all N elements ones. The global forms of (18) and (19) are now expressed as

$$e = -((\mathcal{L} + \mathcal{G}) \otimes I_n)(x^{\mathrm{a}} - \underline{1}_N \otimes x_0), \tag{20}$$

$$\dot{e} = -((\mathcal{L} + \mathcal{G}) \otimes I_n)\dot{x}. \tag{21}$$

In order to further develop the discussion on that hypothesis, the second order error dynamics is first calculated

$$\ddot{e} = -((\mathcal{L} + \mathcal{G}) \otimes I_n)\ddot{x}. \tag{22}$$

Plugging (12) in (22) gives

$$\begin{aligned} \ddot{e} &= -((\mathcal{L} + \mathcal{G}) \otimes I_n)[(I_N \otimes A)x + (I_N \otimes B)v] \\ &= -((\mathcal{L} + \mathcal{G}) \otimes I_n)(A_{\mathrm{g}}x + B_{\mathrm{g}}v), \end{aligned} \tag{23}$$

where $A_{\mathrm{g}} = I_N \otimes A$, and $B_{\mathrm{g}} = I_N \otimes B$. Without the loss of generality, further simplification is introduced

$$\ddot{e} = -((\mathcal{L} + \mathcal{G}) \otimes I_n)(A_{\mathrm{g}}x + u), \tag{24}$$

where $u = B_{\mathrm{g}}v$ is defined as the global vector of inputs.

To drive both position and velocity to zero, we use the sliding mode control and define the sliding mode error as

$$r = \dot{e} + \Lambda e, \tag{25}$$

where Λ is positive definite. From [30], e is bounded as long as r is bounded. Let $\lambda = \mathrm{diag}\{\lambda_i\}$ be N dimensional diagonal matrix with λ_i on the diagonal, each corresponding to one agent. Then $\Lambda = \lambda \otimes I_n$. Taking the time derivative of (25) yields

$$\begin{aligned} \dot{r} &= \ddot{e} + \Lambda\dot{e} \\ &= -((\mathcal{L} + \mathcal{G}) \otimes I_n)(A_{\mathrm{g}}x + u) \\ &\quad - \Lambda((\mathcal{L} + \mathcal{G}) \otimes I_n)\dot{x}. \end{aligned} \tag{26}$$

By using Kronecker rule $(A\otimes B)(C\otimes D) = (AC)\otimes(BD)$ (under the assumption that dimensions of A, B, C, D allow multiplications AC and BD), we introduce the following for convenience

$$\begin{aligned}
&Z = ((\mathcal{L}+\mathcal{G})\otimes I_n) \qquad (27)\\
&Z^A = ((\mathcal{L}+\mathcal{G})\otimes I_n)A_{\mathrm{g}} = ((\mathcal{L}+\mathcal{G})\otimes A)\\
&Z_\lambda = (\lambda\otimes I_n)((\mathcal{L}+\mathcal{G})\otimes I_n) = ((\lambda(\mathcal{L}+\mathcal{G}))\otimes I_n).
\end{aligned}$$

Rewrite (26) as

$$\dot r = -Z^A x - Z_\lambda \dot x - Zu. \qquad (28)$$

Based on the undirected graph topology, we make some assumptions useful for the control design.

Assumption 1 $\mathcal{L}$ is irreducibly diagonally dominant matrix.

Assumption 2 $\mathcal{G}$ has at least one diagonal entry.

It is not restrictive to make Assumptions 1 and 2, since many practical multi-agent systems fall under that category.

We introduce the control law locally and globally as

$$u_i = u_{1i} + K_i r_i, \qquad (29)$$

$$u = u_1 + (I_N\otimes K_i)r = u_1 + Kr, \qquad (30)$$

where matrix $K_i = K_j, \forall i, j$ is used for control design. Assume that u_1 cancels first two terms in (28). u_1 is specified later in the theorem. Under that assumption rewrite (28) in the global form

$$\dot r = -((\mathcal{L}+\mathcal{G})\otimes K_i)r = -ZKr. \qquad (31)$$

Lemma 1 Let Assumptions 1 and 2 hold. If we define

$$W = \mathrm{diag}\{w_i\} = \mathrm{diag}\Big\{\frac{1}{q}_i\Big\}, \qquad (32)$$

where

$$q = (\mathcal{L}+\mathcal{G})^{-1}\underline{1}_N, \qquad (33)$$

then, both W and $Q = W(\mathcal{L}+\mathcal{G})+(\mathcal{L}+G)^{\mathrm{T}}W$ are positive definite. Additionally, if Q is positive definite then $Q\otimes I_n$ is also positive definite. $Q = W(\mathcal{L}+\mathcal{G}) + (\mathcal{L}+G)^{\mathrm{T}}W$.

Proof The first assertion follows the results in [9, 31]. For the second assertion, we obtain that eigenvalues of a matrix generated by Kronecker product are cross-products of all possible combinations of eigenvalues from matrix Q and I_n. Both Q and I_n are positive definite and corresponding eigenvalues all positive. This completes the proof. □

Assumption 3 Let $\bar x$ be the measurement of ideal state vector x

$$\begin{cases} \bar x = x + \mu_1,\\ \dot{\bar x} = \dot x + \mu_2,\\ \bar x_0 = x_0 + \mu_0, \end{cases} \qquad (34)$$

then define the bound on sensor noise

$$\begin{cases} \|\mu_1\| < \mu_{1,\mathrm{B}},\\ \|\mu_2\| < \mu_{2,\mathrm{B}},\\ \|\mu_0\| < \mu_{0,\mathrm{B}}, \end{cases} \qquad (35)$$

where $\mu_{0,\mathrm{B}}$ denotes the measurement error of leader's position.

Sliding mode error injected with noises is

$$\bar r_i = \dot{\bar e}_i + \lambda_i \bar e_i, \qquad (36)$$

where

$$\bar e_i = \sum_{j\in N_i} a_{ij}(\bar x_j - \Delta_j - \bar x_i + \Delta_i) + g_i(\bar x_0 - \bar x_i + \Delta_i), \qquad (37)$$

$$\dot{\bar e}_i = \sum_{j\in N_i} a_{ij}(\dot{\bar x}_j - \dot{\bar x}_i) + g_i(-\dot{\bar x}_i). \qquad (38)$$

The global form of noisy error is

$$\bar r = \dot{\bar e} + \Lambda \bar e, \qquad (39)$$

where

$$\begin{aligned}
\bar e &= -((\mathcal{L}+\mathcal{G})\otimes I_n)(\bar x^{\mathrm{a}} - \underline{1}_N\otimes \bar x_0)\\
&= -((\mathcal{L}+\mathcal{G})\otimes I_n)(x^{\mathrm{a}} + \mu_1 - \underline{1}_N\otimes x_0 - \underline{1}_N\otimes \mu_0)\\
&= -((\mathcal{L}+\mathcal{G})\otimes I_n)(\mu_1 - \underline{1}_N\otimes \mu_0) + e,\\
&= -Z(\mu_1 - \underline{1}_N\otimes\mu_0) + e, \qquad (40)\\
\dot{\bar e} &= -Z\mu_2 + \dot e, \qquad (41)\\
\ddot{\bar e} &= -(A_{\mathrm{g}}\bar x + u) = -ZA_{\mathrm{g}}\mu_1 + \ddot e. \qquad (42)
\end{aligned}$$

Extracting the ideal sliding mode error from the measured error gives

$$\bar r = r - \Lambda Z(\mu_1 - \underline{1}_N\otimes\mu_0) - Z\mu_2. \qquad (43)$$

Substituting (42) and (41) into (26) yields

$$\dot{\bar r} = \dot r - ZA_{\mathrm{g}}\mu_1 - \Lambda Z\mu_2. \qquad (44)$$

Definition 2 The signal $z(t)$ is said to be uniformly ultimately bounded (UUB) with the ultimate bound b, if given positive constants b and c, for every $d \in (0, c)$, there exists $T(d, b)$, independent of t_0, such that

$$\|z(t_0)\| \leqslant d \Rightarrow \|z(t)\| \leqslant b, \ \ \forall t \geqslant t_0 + T. \tag{45}$$

Here, we propose our distributed consensus control for UAVs in the following theorem.

Theorem 1 Let Assumptions 1–3 hold. Define the sliding mode error dynamics as (26). Select the control policy for the local agent as

$$u_i = K_i \bar{r}_i - A\bar{x}_i - \lambda_i \dot{\bar{x}}_i. \tag{46}$$

Assume that $\lambda_i = \lambda_j, \forall i, j$, (46). Consider the error dynamics (31) and design matrices $\mathcal{Q}$ and $\mathcal{R}$. Pick following control gain

$$K_i = \mathcal{R}^{-1}\mathcal{P}, \tag{47}$$

$$K = (I_N \otimes K_i), \tag{48}$$

where $\mathcal{P}$ is the unique positive definite solution of control algebraic Riccati equation

$$0 = \mathcal{Q} - \mathcal{P}\mathcal{R}^{-1}\mathcal{P}. \tag{49}$$

The control law (46) with the gain (47) guarantees asymptotic stability for (31). Moreover it stabilizes the system (12) and makes the ideal tracking error (13) UUB.

Proof Let eigenvalues of $(\mathcal{L} + \mathcal{G})$ be $\nu_i = \alpha_i + j\beta_i$. Matrix $\mathcal{L} + \mathcal{G}$ has $\alpha_i > 0$ because of Assumption 1 and 2. System (31) is asymptotically stable if matrices $-\nu_i K_i$ are asymptotically stable (see [9]). Since $\mathcal{P} > 0$, $\mathcal{R} > 0$ and $\alpha_i > 0$ it follows that matrix $-\nu_i K_i$ is Hurwitz and (31) is asymptotically stable. From Lyapunov theory, the stability condition for (31) is equivalent to

$$(-ZK)^{\mathrm{T}}P + P(-ZK) = Q, \tag{50}$$

where $Q < 0$ and $P > 0$, $P^{\mathrm{T}} = P$, and K from (48). (50) is equivalent to

$$PZK = -\frac{1}{2}\mathcal{Q}. \tag{51}$$

To prove stability in the presence of noisy measurements, consider following Lyapunov function candidate for multi-agent systems in global form

$$V = \frac{1}{2}r^{\mathrm{T}}Pr. \tag{52}$$

Differentiating (52) with respect to (26) yields

$$\begin{aligned}\dot{V} =& r^{\mathrm{T}}P\dot{r}\\ =& r^{\mathrm{T}}P[((\mathcal{L} + \mathcal{G}) \otimes I_n)(-\Lambda\dot{x} - A_{\mathrm{g}}x - u)]\\ =& r^{\mathrm{T}}P[((\mathcal{L} + \mathcal{G}) \otimes I_n)(-\Lambda\dot{x} - A_{\mathrm{g}}x - K\bar{r}\\ &+ A_{\mathrm{g}}\bar{x} + \Lambda\dot{\bar{x}}]\\ =& r^{\mathrm{T}}P\Lambda((\mathcal{L} + \mathcal{G}) \otimes I_n)\mu_2\\ &+ r^{\mathrm{T}}P((\mathcal{L} + \mathcal{G}) \otimes I_n)A_{\mathrm{g}}\mu_1\\ &- r^{\mathrm{T}}P((\mathcal{L} + \mathcal{G}) \otimes I_n)K\bar{r}.\end{aligned} \tag{53}$$

For brevity, we use (27). Combined with (43) gives

$$\begin{aligned}\dot{V} =& r^{\mathrm{T}}PZ_{\lambda}\mu_2 \qquad (54)\\ &+ r^{\mathrm{T}}PZ^{A}\mu_1\\ &- r^{\mathrm{T}}PZK[r - Z_{\lambda}(\mu_1 - \underline{1}_N \otimes \mu_0) - Z\mu_2]\\ =& - r^{\mathrm{T}}PZKr + r^{\mathrm{T}}P(Z_{\lambda} + ZKZ)\mu_2\\ &+ r^{\mathrm{T}}P(Z^{A} + ZKZ_{\lambda})\mu_1\\ &- r^{\mathrm{T}}PZKZ_{\lambda}(\underline{1}_N \otimes \mu_0).\end{aligned} \tag{55}$$

Note that the upper bound on $\dot{V}$ can be found through the norm algebra and the bounds that are previously assumed to limit the measurement noise vectors. To be specific, from (54), we have

$$\begin{aligned}\dot{V} \leqslant& - \min\{\|PZK\| \cdot \|r\|^2\}\\ &+ \max\{\|r\| \cdot \|P(Z^{A} + ZKZ_{\lambda})\| \cdot \|\mu_1\|\}\\ &+ \max\{\|r\| \cdot \|P(Z_{\lambda} + ZKZ)\| \cdot \|\mu_2\|\}\\ &+ \max\{\|r\| \cdot \|PZKZ_{\lambda}\| \cdot \|\underline{1}_N \otimes \mu_0\|\},\end{aligned} \tag{56}$$

based on which (56) is changed into

$$\begin{aligned}\dot{V} \leqslant& - \sigma_{\min}\{PZK\} \cdot \|r\|^2\\ &+ \|r\| \cdot \sigma_{\max}\{P(Z^{A} + ZKZ_{\lambda})\} \cdot \mu_{1,\mathrm{B}}\\ &+ \|r\| \cdot \sigma_{\max}\{P(Z_{\lambda} + ZKZ)\} \cdot \mu_{2,\mathrm{B}}\\ &+ \|r\| \cdot \sigma_{\max}\{PZKZ_{\lambda}\} \cdot \mu_{1,0,\mathrm{B}},\end{aligned} \tag{57}$$

where $\sigma_{\max}(M)$ and $\sigma_{\min}(M)$ denote maximum and minimum eigenvalue of matrix M. From (57), it is clear that $\dot{V}$ is always negative for

$$\begin{aligned}\|r\| >& \frac{\sigma_{\max}\{P(Z^{A} + ZKZ_{\lambda})\}}{\sigma_{\min}\{PZK\}}\mu_{1,\mathrm{B}}\\ &+ \frac{\sigma_{\max}\{P(Z_{\lambda} + ZKZ)\}}{\sigma_{\min}\{PZK\}}\mu_{2,\mathrm{B}}\\ &+ \frac{\sigma_{\max}\{PZKZ_{\lambda}\}}{\sigma_{\min}\{PZK\}}\mu_{1,0,\mathrm{B}}.\end{aligned} \tag{58}$$

Moreover, if r is outside the bounding set given in (58), our controller will eventually drag the error inside the set. In other words, the sliding mode error r is bounded to a set which makes all the signals in the closed-loop multi-agent system UUB. □

The controller drags all aliases x_i^{a} for $i = 1,\dots,N$ to the center of the formation. If all aliases are in the center of the formation, no action from the controller is required. Neighbourhood errors quantify the spatial disharmony of the formation. The controller determines an action u_i of the agent i such that it recovers consensus. Λ is a tuning parameter. As given, controller resembles well known standard PD controller.

5 Modifying edge weights based on trust

In this section, it is shown how UAVs in the graph can distributively modify the graph weights based on their measure of trustworthiness of the neighbors. The purpose is to maintain the consensus by eliminating negative influence of stubborn agents or extremely noisy measurement imposed on the local agent.

The trust that agent i has for agent j can be calculated as deviation from the intended average state of neighboring agents

$$\tau_{ij} = \frac{\theta_{ij}}{\|\bar{x}_j^{\mathrm{a}} - \dfrac{1}{N_i+1}(\bar{x}_i^{\mathrm{a}} + \sum\limits_{j\in N_i} \bar{x}_j^{\mathrm{a}})\| + \theta_{ij}}, \tag{59}$$

where θ_{ij} is a certain positive constant that serves as a threshold value. From (59), when the local agent j approaches the average of the neighbourhood agents, the trust value τ_{ij} approaches one; otherwise, τ_{ij} approaches zero. With this trust metric, we further propose a mechanism to modify the graph weights. To this end, we use the differential equation

$$\dot{d}_{ij} = -\eta d_{ij} + \eta\tau_{ij}, \tag{60}$$

where η is a certain positive constant. To take into account the trust value calculated for neighbors, the controller (46) is now revisited by weighting all edge weights a_{ij} by d_{ij}. That is all occurrences of a_{ij} in (46) are replaced by $d_{ij}a_{ij}$.

6 Experimental environment

Fig. 1 shows an entire system designed at UTA Research Institute for testing distributed controllers on UAVs. Basic four elements are VICON, Crazyflie 2.0 UAVs, and the workstation that runs master program. Master is implemented as ROS package and it handles consensus controller calculation as well as local manipulation over UAVs.

As already mentioned, Crazyflie 2.0 is used as UAV and it communicates with master workstation using radio transceiver. In order to have reliable feedback information, we use motion capture system VICON. Data is transmitted to master through wireless local area network. Such communication architecture enables consensus controller to send the control commands to UAVs at 200 Hz, which is more than sufficient.

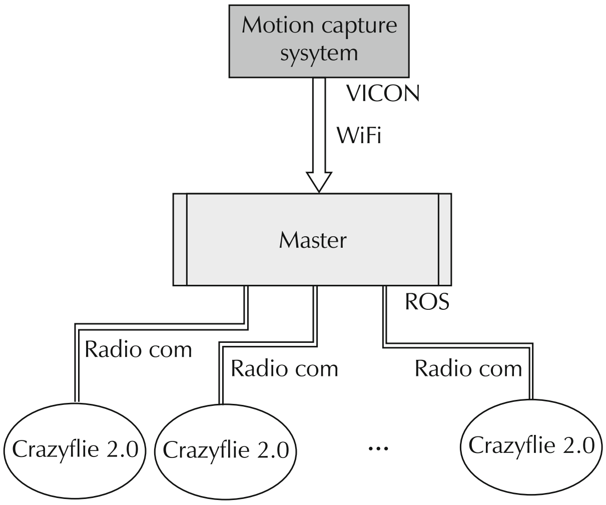

Fig. 1 Motion capture and communication topology based on master-slave model architecture.

6.1 Crazyflie 2.0

Crazyflie 2.0 is a commercially available UAV suitable for research and development. It is created by Bitcraze AB. We use their hardware and modify the firmware to implement our control laws. IMU data collection, four motors with corresponding PWM modules and communication with radio (nRF51822) is all handled by microcontroller (STM32F405). Attitude controller is implemented on firmware side rather than on master/server. It uses inputs from outer control loop (commanded velocities in all three dimensions and commanded yaw rate) and IMU data to stabilize UAV and to drive it to the desired position. Desired values are received from nRF51822, previously sent from the master. The output of attitude controller are signals that determine PWM cycle duty and eventually produce propeller thrusts and torques. For more hardware and construction details, as well as for a simplified mathematical model of Crazyiflie 2.0 refer to the official website [32].

6.2 Master workstation

Fig. 2 shows the software architecture that is executed on the master workstation. The master controller is a ROS node that manages general control of all UAVs. Each UAV is controlled by individual outer loop controller.

The master controller is, in fact, a mission planner. It commands different tasks to UAVs by communicating with corresponding position controller. It also listens to external user input to allow manual task switching, manual take off, manual landing, and manual emergency landing. State machine seems to be a plausible design solution for master controller, considering its current function of switching states. As Fig. 3 clearly shows, there are three different states in state machine. In waypoint tracking state, master controller simply sets waypoints to position controllers. Consensus controller is slightly more complex, as generalized formation error for each UAV has to be recalculated at every iteration. The idle state is active when all Crazyflies are off. It is important to notice that user has the control over whole state machine which is a security requirement.

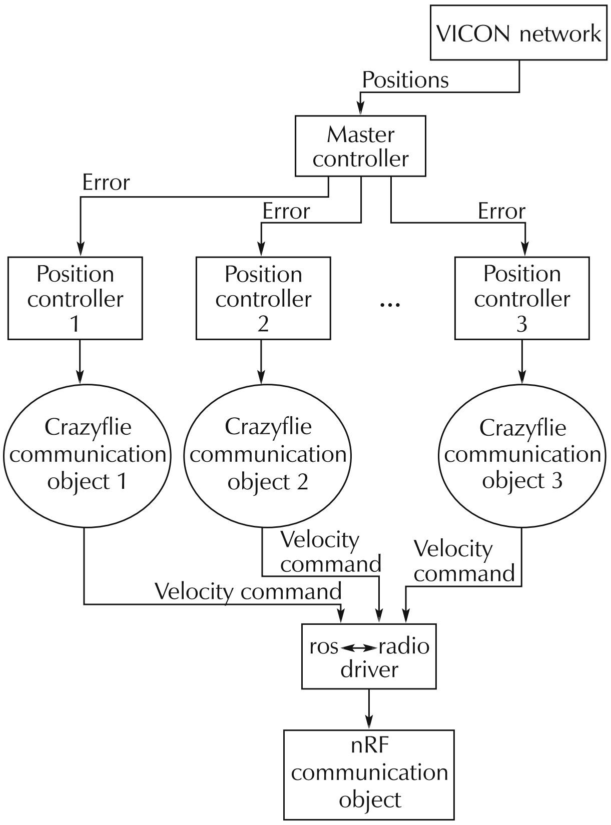

Fig. 2 Distributed multi-threaded software architecture with data pipeline.

The described procedure can be broken down into the following main steps:

Step 1 Check the link quality. If link quality is low proceed to Step 7. Otherwise continue to Step 2.

Step 2 Collect data from VICON, distribute information to agents according to adjacency matrix.

Step 3 Update trust values and apply them to modify adjacency matrix.

Step 4 Call local distributed controllers. Send calculated roll, pitch, yaw and thrust commands to each agent.

Step 5 Calculate PWM locally on each agent and generate forces.

Step 6 If no user interrupt, go back to Step 1. Otherwise go to Step 7.

Step 7 Start landing sequence. Land.

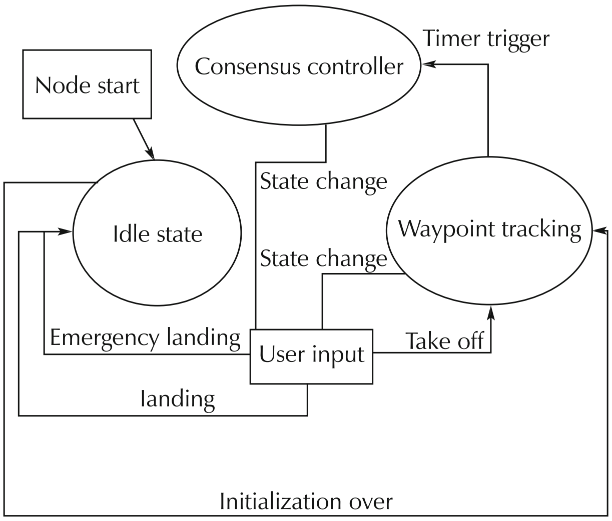

Fig. 3 Distributed hierarchical motion control state machine.

7 Flight tests

Three flight tests shown in Fig. 4 are carried out in this section, where different adjacency matrices are picked. For each experiment, we show two plots, x position and y position for all three UAVs. The desired reference is set $(0, 0, 0.7)$ and disturbance is applied to one of three UAVs to test the behaviour of the group. To be specific, we push the UAV away from the point of global equilibrium imposed by consensus controller. Displacements from the desired reference are $\Delta_1 = (-0.5, 0.5, 0.0)$, $\Delta_2 = (0.5, 0.5, 0.0)$, and $\Delta_3 = (-0.5, -0.5, 0.0)$. Note that in the experiment, we define the virtual input u as

$$u = (I_N \otimes B)v = B_g v. \tag{61}$$

Technical control vector v can now be calculated from virtual inputs u through

$$u = (I_N \otimes B)v, \tag{62}$$

$$v_i = B^{\mathrm{T}}(BB^{\mathrm{T}})^{-1}u_i. \tag{63}$$

In fact, the ability to calculate v is not only a matter of convenience, but rather a necessity, since v is the command vector that is transmitted to UAV agents. u_i is a distributed control signal (46). In the first 5 seconds, altitude control is applied to take off UAVs, and after that our consensus controller is activated.

Fig. 4 Photograph of the experiment during consensus.

7.1 Experiment 1

In this case, we define the adjacency matrix as

$$\mathcal{A} = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 3 \\ 2 & 3 & 0 \end{bmatrix} \tag{64}$$

and the pinning gains as $g_2 = 3$ and $g_1 = g_3 = 0$. After applying our controller, the experimental results are presented in Figs. 5 and 6, where UAVs 1–3 are labelled in red, blue, and green. Here, UAV 1 is the agent that is injected with disturbances. In this experiment, UAV 1 moves away from the consensus due to disturbances, while UAVs 2 and 3 try to follow. From (64), UAV 3 strictly follows movements of agent 1 in order to keep consensus, since it has direct link to agent 1. Note that UAV 2 is less responsive to position change of agent 1 due to the fact that agent 2 is connected to the leader.

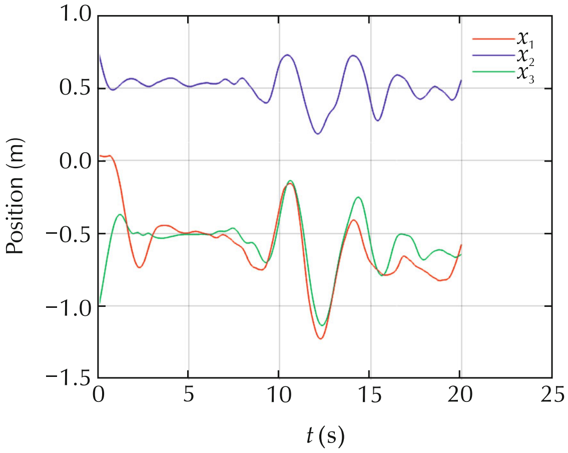

Fig. 5 Experiment 1, positions x of UAVs.

Also notice that UAV 1 is first sent way out of desired position at around 9th second. At the very peak, when UAV 1 is more than 0.5 m away from desired position, UAV 3 does not follow UAV 1 to the extreme deviation because it has lost trust in UAV 1 (a_{13} was close to 0). The fact that trajectories of UAVs 1 and 3 have slightly different shapes at that point confirms that (Fig. 6). As UAV 1 returns and distance between UAVs 1 and 3 is back as expected, trust increases and consensus is re-established.

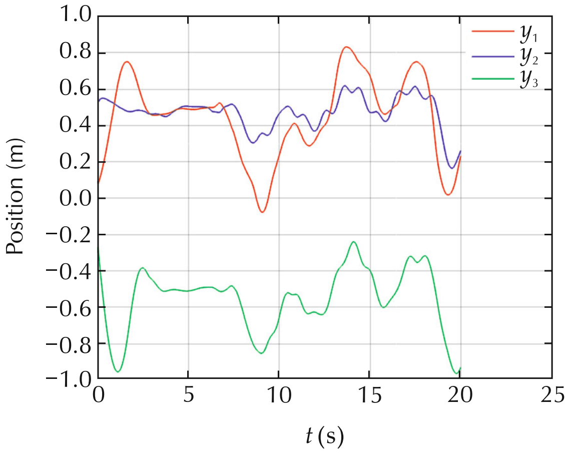

Fig. 6 Experiment 1, positions y of UAVs.

7.2 Experiment 2

Here, we define the adjacency matrix as

$$\mathcal{A} = \begin{bmatrix} 0 & 2 & 2 \\ 2 & 0 & 3 \\ 2 & 3 & 0 \end{bmatrix} \tag{65}$$

and the pinning gains as $g_2 = 3$ and $g_1 = g_3 = 0$. Adding a direct link between agents 1 and 2 makes UAV 2 a bit more responsive. Formation control in Figs. 7 and 8 is better compared to that in the previous experiment.

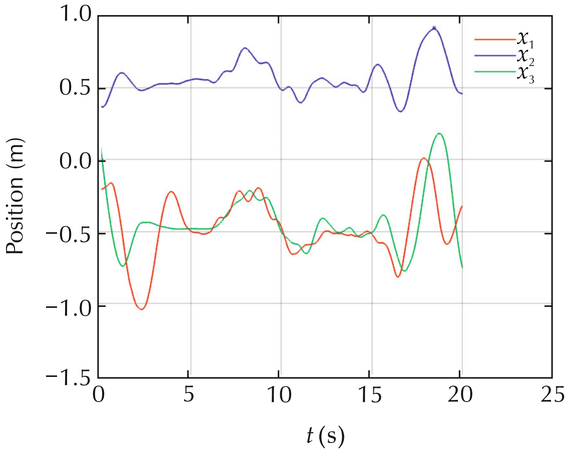

Fig. 7 Experiment 2, positions x of UAVs.

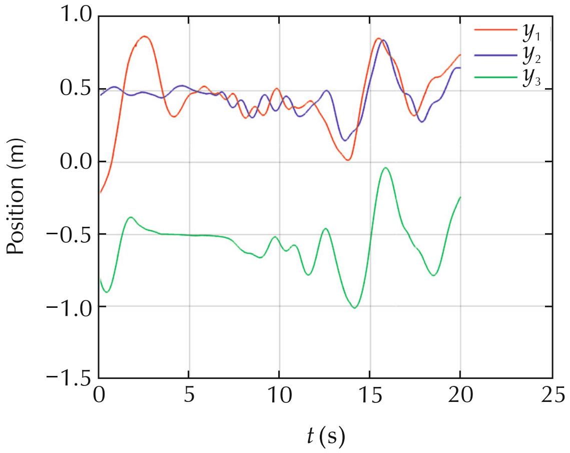

Fig. 8 Experiment 2, positions y of UAVs.

7.3 Experiment 3

In the final experiment, the adjacency matrix is designed as

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \tag{66}$$

and pinning gains are given as $g_1 = g_2 = g_3 = 1$. Compared to the previous experiments, (66) gives the best performance, as shown in Figs. 9 and 10. Such results are expected because the graph is dense and weights are not giving preference to any neighbouring agent.

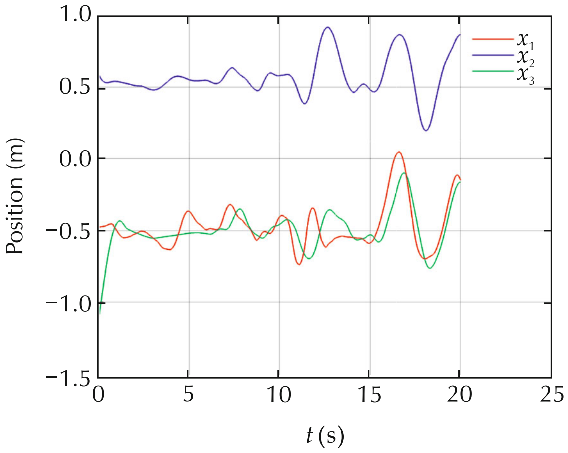

Fig. 9 Experiment 3, positions x of UAVs.

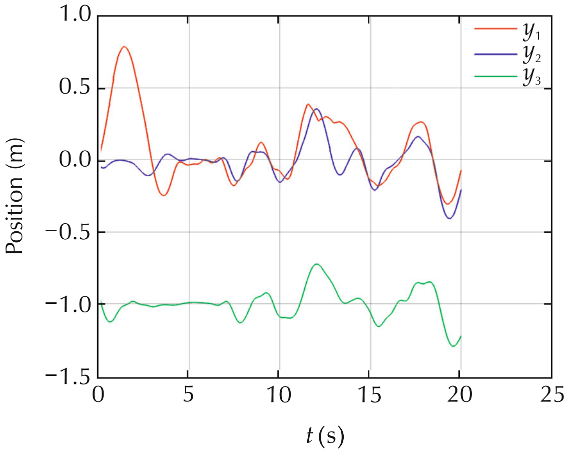

Fig. 10 Experiment 3, positions y of UAVs.

8 Conclusions

Consensus control design introduced in this paper is effective in keeping a group of UAVs in the formation. Confirmation is given by using Lyapunov theory to prove the stability of the group and later in the experimental environment on the system of multi UAVs.

It is shown that the robustness of consensus depends highly on the graph topology. In that aspect, fully connected graphs will provide better basis for firm consensus control. Unfortunately, it is often the case that practical multi-agent network topology is inherent to the system to which control is applied. In other words, while designing controller, graph is predefined and cannot be changed in favor of better control. However, one thing we can do is to update the values of the links that are present based on the trustworthiness of a neighbouring agent. Our approach achieves that.

Due to certain practical assumptions about dynamics of the nano UAV and by restricting experiments to a slow flight, we were able to make some reasonable simplifications on the dynamical model of the UAV. That allowed for more transparent design of the consensus controller in the light of measurement noises imposed on the system. Future work should extend current design to the domain of non-linear control, allowing for dynamically rich experiments and treat the measurement noise with advanced tools such as distributed Kalman filter.

References

[1] U.S. Air Force Research Laboratories. *UAV Cooperative Decision and Control; Challenges and Practical Approaches*. T. Shima, S. Rasmussen (eds.). Philadelphia: SIAM. 2009.

[2] R. M. Murray. Recent research in cooperative control of multivehicle systems. *Journal of Dynamic Systems Measurement and Control – Transactions of the ASME*, 2007, 129(5): 571 – 583.

[3] Z. Qu. *Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles*. London: Springer, 2009.

[4] A. Das, K. Subbarao, F. L. Lewis. Dynamic inversion with zero-dynamics stabilisation for quadrotor control. *IET Control Theory & Applications*, 2009, 3(3): 303 – 314.

[5] A. Das, F. L. Lewis, S. Subbarao. Backstepping approach for controlling a quadrotor using neural network. *Journal of Intelligent and Robotics Systems*, 2009, 56(1): 127 – 151.

[6] F. L. Lewis, S. Jagannathan, A. Yesildirak. *Neural Network Control of Robot Manipulators and Non-linear Systems*. Boca Raton: CRC Press, 1998.

[7] P. Wang, Z. Man, Z. Cao. Dynamics modelling and linear control of quadcopter. *International Conference on Advanced Mechatronic Systems*, Melbourne: IEEE, 2016: 498 – 503.

[8] G. V. Raffo, M. G. Ortega, F. R. Rubio. An integral predictive/nonlinear H_∞ control structure for a quadrotor helicopter. *Automatica*, 2010, 46(1): 29 – 39.

[9] F. L. Lewis, H. Zhang, K. Hengster-Movric, et al. *Cooperative Control of Multi-agent Systems: Optimal and Adaptive Design Approaches*. London: Springer, 2013.

[10] R. Olfati-Saber, J. A. Fax, R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 2007, 95(1): 215 – 233.

[11] W. Ren, R. W. Beard, E. M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems*, 2007, 27(2): 71 – 82.

[12] Randal W. Beard, Timothy W. McLain. *Small Unmanned Aircraft: Theory and Practice*. Princeton: Princeton University Press, 2012.

[13] S. Khoo, L. Xie, Z. Man. Robust finite-time consensus tracking algorithm for multirobot systems. *IEEE Transactions on Mechatronics*, 2009, 14(2): 219 – 228.

[14] X. Xu, L. Liu, G. Feng. Consensus of single integrator multi-agent systems with directed topology and communication delays. *Control Theory and Technology*, 2016, 14(1): 21 – 27.

[15] Y. Kuriki, T. Namerikawa. Consensus-based cooperative formation control with collision avoidance for a multi-UAV system. *American Control Conference*, Portland: IEEE, 2014: 2077 – 2082.

[16] M. Yan, X. Zhu, X. Zhang, et al.Consensus-based three-dimensional multi-UAV formation control strategy with high precision. *Frontiers of Information Technology & Electronic Engineering*, 2017, 18(7): 968 – 977.

[17] J. Wang, X. Nian, H. Wang. Consensus and formation control of discrete-time multi-agent systems. *Journal of Central South University of Technology*, 2011, 18(4): 1161 – 1168.

[18] A. Das, F. L. Lewis. Cooperative adaptive control for synchronization of second-order systems with unknown nonlinearities. *International Journal of Robust and Nonlinear Control*, 2011, 21(13): 1509 – 1524.

[19] J. Thunberg, W. Song, Y. Hong, et al. Distributed attitude synchronization using backstepping and sliding mode control. *Control Theory and Technology*, 2014, 12(1): 48 – 55.

[20] H. Zhang, F. L. Lewis, A. Das. Optimal design for synchronization of cooperative systems: state feedback, observer and output feedback. *IEEE Transactions on Automatic Control*, 2011, 56(8): 1948 – 1952.

[21] D. Vrabie, F. L. Lewis. Adaptive dynamic programming for online solution of a zero-sum differential game. *Journal of Control Theory and Applications*, 2011, 9(3): 353 – 360.

[22] K. G. Vamvoudakis, F. L. Lewis. Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton-Jacobi equations. *Automatica*, 2011, 47(8): 1556 – 1569.

[23] K. G. Vamvoudakis, F. L. Lewis. Online solution of nonlinear two-player zero-sum games using synchronous policy iteration. *International Journal of Robust and Nonlinear Control*, 2012, 22(13): 1460 – 1483.

[24] T. Yucelen, W. M. Haddad, E. M. Feron. Adaptive control architectures for mitigating sensor attacks in cyber-physical systems. *American Control Conference*, Boston: IEEE, 2016: 1165 – 1170.

[25] E. Arabi, T. Yucelen, W. M. Haddad. Mitigating the effects of sensor uncertainties in networked multiagent systems. *American Control Conference*, Boston: IEEE, 2016: 5545 – 5550.

[26] J. S. Baras, T. Jiang. Trust evaluation in anarchy: a case study on autonomous networks. *Proceedings of IEEE INFOCOM*, Barcelona: IEEE, 2006: 835 – 846.

[27] J. S. Baras, T. Jiang. Cooperation, trust and games in wireless networks. *Symposium on Systems, Control, and Networks*, Berkeley: Birkhauser, 2005: 183 – 202.

[28] T. Jiang, J. S. Baras. *Graph Algebraic Interpretation of Trust Establishment in Autonomic Networks*. Maryland: University of Maryland, 2010.

[29] Y. Kuriki, T. Namerikawa. Formation control of UAVs with a fourth-order flight dynamics. *The 52nd IEEE Conference on Decision and Control*, Florence: IEEE, 2013: 6706 – 6711.

[30] J.-J. E. Slotine, W. Li. *Applied Nonlinear Control*. Englewood Cliffs: Prentice Hall, 1991.

[31] Z. Qu. *Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles*. New York: Springer, 2009.

[32] Bitcraze. https://www.bitcraze.io/.

Patrik KOLARIC has received his bachelor's and master's degree at Faculty of Electrical Engineering and Computing, Zagreb, Croatia in 2014 and 2016 respectively. He is pursuing Ph.D. under Dr. Frank L. Lewis's supervision at University of Texas at Arlington. He is research assistant in Autonomous System Lab at University of Texas at Arlington Research Institute where he works on wide range of practical topics related to robotics. His areas of interests include intelligent control, machine learning and sensor networking. E-mail: patrik.kolaric@mavs.uta.edu.

Ankur V. DALAL has received his bachelor's degree from India. Along with pursuing his Masters in Mechanical engineering at University of Texas at Arlington, he has been working as a research intern at University of Texas Arlington Research Institute and robotics engineer at drone delivery company, WAEC LLC. His areas of interest include in robotics and controls. E-mail: ankur.dalal@mavs.uta.edu.

Ci CHEN received the B.Sc. degree and Ph.D. with the School of Automation, Guangdong University of Technology, Guangzhou, China, in 2011 and 2016, respectively. He was a research assistant in School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, from October 2015 to January 2016. Currently, he is with School of Automation, Guangdong University of Technology, and also with The University of Texas at Arlington. E-mail: gdutcc@gmail.com.

Frank L. LEWIS Member, National Academy of Inventors. Fellow IEEE, Fellow IFAC, Fellow AAAS, Fellow U.K. Institute of Measurement & Control, PE Texas, U.K. Chartered Engineer. UTA Distinguished Scholar Professor, UTA Distinguished Teaching Professor, and Moncrief-O'Donnell Chair at the University of Texas at Arlington Research Institute. Qian Ren Thousand Talents Consulting Professor, Northeastern University, Shenyang, China. China Liaoning Friendship Award. He obtained the Bachelor's Degree in Physics/EE and the MSEE at Rice University, the MS in Aeronautical Engineering from Univ. W. Florida, and the Ph.D. at Ga. Tech. He works in feedback control, intelligent systems, cooperative control systems, and nonlinear systems. He is author of 7 U.S. patents, numerous journal special issues, 374 journal papers, and 20 books, including Optimal Control, Aircraft Control, Optimal Estimation, and Robot Manipulator Control which are used as university textbooks worldwide. He received the Fulbright Research Award, NSF Research Initiation Grant, ASEE Terman Award, Int. Neural Network Soc. Gabor Award, U.K. Inst Measurement & Control Honeywell Field Engineering Medal, IEEE Computational Intelligence Society Neural Networks Pioneer Award, AIAA Intelligent Systems Award. Received Outstanding Service Award from Dallas IEEE Section, selected as Engineer of the year by Ft. Worth IEEE Section. Was listed in Ft. Worth Business Press Top 200 Leaders in Manufacturing. Texas Regents Outstanding Teaching Award 2013. He is Distinguished Visiting Professor at Nanjing University of Science & Technology and Project 111 Professor at Northeastern University in Shenyang, China. Founding Member of the Board of Governors of the Mediterranean Control Association. E-mail: lewis@uta.edu.