This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

Stochastic Event-Based Super-Twisting Formation Control for Multi-Agent System under Network Uncertainties

Anuj Nandanwar, *Student Member, IEEE*, Narendra Kumar Dhar, Dmitry Malyshev, Larisa Rybak, and Laxmidhar Behera, *Senior Member, IEEE*

***Abstract*—This paper presents a novel stochastic event-based super-twisting controller design for addressing the formation control problem in networked multi-agent system in the presence of external disturbance. The stochasticity in the system is introduced by randomness of network uncertainties i.e. losses and delays of data packets. The proposed design has three parts. The first part derives event-triggering condition for control input updates of each agent in the system. The second part guarantees desired formation in finite time by deriving an upper bound on reaching time to the designed sliding surface of each agent. The third part ensures admissibility of event based control updates of the agents such that they achieve stable desired formation. The proposed stochastic design has been validated on Pioneer P3-DX mobile robots. The results show the threefold effectiveness of stochastic design. They are a) retaining the advantages of event-triggering strategy, b) retaining the advantages of super-twisting sliding mode controller, and c) robust towards network uncertainties.**

***Index Terms*—Stochastic, packet loss, delay, event-trigger, sliding surface, and stability.**

I. Introduction

The consensus in multi-agent system (MAS) has received large attention from researchers in different areas due to its broad application in various fields, i.e., exploration, military surveillance, rescue operation, robotics, etc. [1]. As the control objectives vary with the application, the methodologies developed are also different. From the system perspective, the methodologies have been developed for integrator model [2], [3], general linear model [4], heterogeneous model [5], asynchronous-time model [6], higher-order model [7] etc. for different types of consensus i.e. leader-follower consensus [2], cyclic pursuit consensus [3], adaptive consensus [5], optimal consensus [4], asynchronous consensus [6], finite-time consensus [7] and so on. The stability of system is an important aspect which is proved asymptotically [2]–[6] and in finite time [7]. All these works share a common feature of having the continuous communication between the neighboring robots. Alongside, the time-triggered approach for the control law, increases communication and computing costs.

The event-based approach can be an alternative to reduce these cost burdens to certain extent [8]. The event-triggering condition(s) are significant in preserving the basic properties of system stability and its performance. Till now several works have been done on event-triggered control of MAS based on undirected topology [9], [10], switching topology [11], [12], sequentially connected topology [13], directed topology [14], [15], and so on. The various event-based control schemes related to consensus in MAS are consensus protocol [13], cooperative control [9], adaptive control [10], decentralized control [11], pining control [12], sampled-data control [14], distributed control [15], and so on. All the above works need perfect communication link amongst the agents which might not be possible always in real-time. In real-time applications such as search and rescue operation by multi-robot system (MRS), robots need to coordinate with each other to accomplish a given task. The exchange of information among agents may be prone to network uncertainties i.e. channel noise, delay, and packet losses etc. They may induce some complex dynamic behaviors such as divergence, oscillation, and even lead to system instability.

The event-triggered control techniques have been reported to handle network uncertainties individually [16], [17] or several at a time [18]–[22]. An analysis proposed in [16] considers time-interval between triggering instants to be larger than the transmission delay between them. However, this may be difficult to implement in practice. The packet drop-out is defined probabilistically and a periodic event-triggering mechanism for linear time-varying system has been derived in [17]. To avoid the impact of packet losses and disturbance, output-based static and dynamic event-triggering strategies with and without acknowledgement cases excluding the Zeno-behaviour have been investigated in [18]. Zhang and Yu [19] have proposed a switched linear system model for the networked control system prone to both packet-dropout and network-induced delay. A performance analysis of distributed event-triggered control for linear interconnected system affected by communication delay and packet losses is presented in [20]. A self-triggered sampling scheme has been proposed for data losses and communication delay in system. The sampling interval is dynamically adjusted without depending on the continuous/discrete measurement and on-line estimation of an event-triggered condition [21]. Zhou *et al.* [22] have proposed an active event-triggered control approach for non-linear system model in presence of packet losses, network induced delay, and quantization effects.

The deterministic event-sampled schemes [16]–[22] are ef-

This work is jointly supported by DST, India under the project DST-EE-2018309 and by RFBR, Russia under the project 18-51-45001 IND_a.

A. Nandanwar, N. K. Dhar, and L. Behera are with the Dept. of Electrical Engg., IIT Kanpur, India ({anujnand; nardhar; lbehera}@iitk.ac.in).

D. Malyshev and L. Rybak are with the Dept. of Mechanical Engg., BSTU, Russian Federation (dimon-belgorod@yandex.ru; rlbgtu@gmail.com).

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

ficient in reducing resource consumption of agents. The deterministic schemes fall short in performance when systems are affected by noise or disturbance. Therefore, stochastic event-based design for MAS become suitable and applicable. A distributed stochastic event-driven control strategy for leader-following MASs handles time-varying and fixed delays in [23]. The integer weight balancing problem for direct topology affected by delay and drop-out in communication link has been solved using event-based distributed algorithm in [24]. In [25], finite-time control problem of high-order stochastic non-linear system has been solved using back-stepping control approach. Zhu *et al.* [26] have proposed a time-varying output feedback backstepping controller for stochastic differential equation. The *K*-filters with low gain are developed to compensate the unmeasurable/unmeasured state. An adaptive output-feedback control of stochastic nonholonomic system with nonlinear parameterization has been studied in [27]. The work proposes reduced-order *K*-filters to compute unmeasurable/unmeasured state. Both the works [26], [27] consider parametric uncertainty in a delay-free system model with stochastic noise. The work in [28] proposes Razumikhin theorem for examining stability of system with stochastic delay. A key aspect of this work is the time-derivatives of Razumikhin functions are allowed to be indefinite, which reduce some restrictions existing in earlier works. The common feature of all these works [25]–[28] is that the system model contains a variable which represents random white noise calculated as derivative of Brownian motion or Wiener process.

The sliding mode control (SMC) improves the system performance in the presence of uncertainties. It is a well known robust and efficient regulation technique. However, in real-time scenario, the implementation of SMC is influenced by "chattering" phenomenon which means oscillations with finite frequency and amplitude. The chattering may cause wear and tear of moving parts, high energy loss, and low control accuracy. Besides it may excite the unmodelled dynamics and high frequency modes which are not generally considered while deriving the dynamical model of the system. The chattering can deteriorate the performance of system and may cause it to be unstable. Hence, the control law should take care of this phenomenon. The higher order SMCs (HOSMC) eliminate chattering phenomena up-to a large extent. Besides, they ensure finite-time convergence of the sliding variable and its higher derivatives with improved robustness against bounded disturbance and uncertainties. The second-order SMC (SOSMC) has proved to be very useful for such purpose. The super-twisting SMC (STSMC) is very popular among the HOSMCs [29]. The work in [30] addresses a problem on tracking control of multiple agents under cyber-physical framework using SOSMC considering packet losses constraint in communication optimization problem. The event-triggered consensus-based formation approaches for first-order MAS prone to disturbance using conventional and integral SMCs are proposed in [31] and [32], respectively. Similarly, work in [33] considers disturbance to derive dynamic event-triggering condition using STSMC. For handling time varying delays and structural uncertainties in the system, Wang *et al.* [34] proposed a time-triggered integral-SMC. Poznyak [35] investigated a stochastic version of STSMC using Lyapunov function in which the state-dependent gain variables are designed to ensure the mean-square convergence of closed-loop stochastic system in a prespecified zone.

It should be noted from the above references that the controller design problems are mostly based in continuous-time domain rather than in discrete-time domain. The work [36] studied a discrete-time version of the STSMC. The sufficient conditions for the asymptotic stability of linear discrete-time system with an upper bound on input delay by an affine state-feedback controller has been derived in [37]. Yu *et al.* in [38] proposed an event-triggered dynamic tracking controller using distributed observer based on second-order neighbors' information for leader-follower multi-agent systems subjected to packet losses and time-varying delays. The work [39] is one of the first to study the event-triggered control for stochastic system with exogenous disturbances. It is based on stabilization of stochastic nonlinear delay system with exogenous disturbance using static event-triggered feedback approach. A novel H_∞ control problem is addressed for a discrete-time stochastic system with an event-based scheme in the presence of packet losses in [40]. It is evident in a discrete-time system, the finite sampling rate usually fails to have convergence of state trajectory on the sliding surface. Consequently, a practical sliding mode approach known as quasi-sliding mode is used in which state trajectory converges within a boundary layer [41]. That and Ha [42] proposed a discrete-time SMC for linear system with time delay and disturbance in deterministic manner. In [43], Niu and Ho proposed an effective probability dependent SMC scheme for discrete-time nonlinear system subject to packet losses. A discrete-time SMC is investigated for nonlinear systems with multiple communication delays [44]. Hu *et al.* [45] addressed the robust SMC for discrete-time stochastic system with mixed time delays, uncertainties, and non-linearity. A robust discrete-time SMC (DSMC) stabilizing a networked system involving time-varying communication delay has been designed in [46]. Hu *et al.* [47] investigated the robust SMC problem for mixed delay and packet losses under certain probability. The event-driven SMC based observer and state-feedback designs for discrete-time system have been presented in [48].

However, there are some important differences between all these works and the one proposed in this paper. They are: a) The approaches in [2], [4]–[7], [9]–[15], [49], [50] work only in continuous time domain and do not consider the network uncertainties. b) The work in [3], [16]–[21], [30], [37], [42] consider network uncertainties only in deterministic manner. c) The techniques in [23], [24], [38]–[40] deal with event-triggered stochastic uncertainties using other controllers but SMC is not considered. d) The techniques in [25]–[28] deal with stochastic uncertainties using controllers other than event-triggered SMC. These works consider single agent system modeled using stochastic differential equations. e) The works in [31]–[33] deal with disturbance using event-triggered SMC but network uncertainties are not considered. f) The strategies in [9]–[13], [31], [32], [40], [48] consider a static event-triggering threshold. g) The discrete-time SMC proposed in [42]–[47] can handle packet loss, delay, and disturbance sepa-

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

rately as well as together but event-triggering is not considered. h) The work [48] handles packet losses for stochastic system using event-triggered discrete SMC scheme but delay and disturbance are not considered. i) The strategies in [31], [32], [34], and [42]–[48] use SMC but HOSMC is not considered.

In summary, this paper has following contributions.

- A novel design of real-time stochastic event-based STSMC for consensus based formation control in leader-follower framework with a directed graph topology.
- The novel upper bound on reaching time to the sliding surface of each agent.
- The admissibility guarantee of event-based control updates proved through minimum inter-event time.

The *advantages* of proposed controller are: a) it takes care of effects due to random delays and packet losses, b) its event-based nature avoids unnecessary computations and transmission of control inputs to plant, and c) it guarantees consensus based formation of agents in finite-time.

The outline of remaining paper is as follows. The preliminaries along with the problem formulation are presented in Section II. The proposed methodology is presented in Section III. Section IV demonstrates results based on real-time experiments and comparative analysis with other approaches. The conclusion is drawn in Section V.

II. Preliminaries and Problem Formulation

A. System Dynamics

Let us consider a non-holonomic multi-robot system moving in a $\mathbf{X}-\mathbf{Y}$ plane. All the robot models have similar built and dynamic configuration. The kinematic model of robot i shown in Fig. 1 is expressed similarly as [32].

$$\dot{x}_i^c = v_i\cos(\theta_i), \quad \dot{y}_i^c = v_i\sin(\theta_i), \text{ and } \quad \dot{\theta}_i = \omega_i, \tag{1}$$

where $\bar{\mathrm{x}}_i = [x_i^c, y_i^c]^\top$ and θ_i denote position vector, and orientation of robot respectively. v_i and ω_i are linear and angular input velocities, respectively.

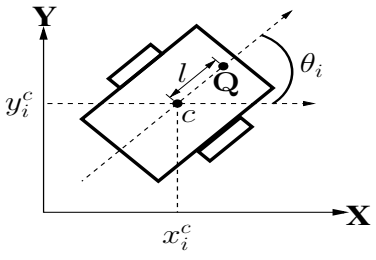


Fig. 1: A schematic of robot model.

Consider an off-axis point $\mathbf{Q}$ as the point of operation for robot i. The kinematic model (1) is transformed into single-integrator form. The off-axis point is at (x_i, y_i) where $x_i = x_i^c + l\cos\theta_i$ and $y_i = y_i^c + l\sin\theta_i$ with distance l between center of gravity (c) and $\mathbf{Q}$. Therefore, robot dynamics about the off-axis point can be expressed as

$$\begin{aligned} \dot{x}_i &= v_i\cos\theta_i - l\omega_i\sin\theta_i \quad \text{and} \\ \dot{y}_i &= v_i\sin\theta_i + l\omega_i\cos\theta_i. \end{aligned} \tag{2}$$

On rearranging (2), we get

$$\begin{aligned} v_i &= \dot{x}_i\cos\theta_i + \dot{y}_i\sin\theta_i \quad \text{and} \\ \omega_i &= (-1/l)(\dot{x}_i\sin\theta_i - \dot{y}_i\cos\theta_i). \end{aligned} \tag{3}$$

The linear and angular velocities are represented as

$$\begin{aligned} v_i &= u_{ix}\cos\theta_i + u_{iy}\sin\theta_i \quad \text{and} \\ \omega_i &= (-1/l)(u_{ix}\sin\theta_i - u_{iy}\cos\theta_i), \end{aligned} \tag{4}$$

where $\mathrm{u}_i = [u_{ix}, u_{iy}]^\top$ is control input of robot i. On formulating (2) as single-integrator form using (4)

$$\dot{x}_i = u_{ix} \quad \text{and} \quad \dot{y}_i = u_{iy}. \tag{5}$$

The robot's movement is prone to external disturbances (d_{ix}, d_{iy}). The dynamics (5) after incorporating disturbances,

$$\dot{x}_i = u_{ix} + d_{ix} \quad \text{and} \quad \dot{y}_i = u_{iy} + d_{iy}. \tag{6}$$

The dynamics (6) can thereby be expressed simply in the form of single integrator model as

$$\dot{\mathrm{x}}_i = \mathrm{u}_i + \mathrm{d}_i, \tag{7}$$

where $\mathrm{x}_i = [x_i, y_i]^\top$ and $\mathrm{d}_i = [d_{ix}, d_{iy}]^\top$ are position and lumped disturbances of robot i, respectively.

B. Multi-Agent System Framework

There are M agents in MAS framework. The interaction among agents in distributed MAS is modeled as directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ where $\mathcal{V} = 1, 2, \ldots, M$ is the set of nodes. $A = [a_{ij}]$ is a weighted adjacency matrix where $a_{ij} = 1$ if and only if edge $(j, i) \in \mathcal{E}$ (edge matrix) otherwise $a_{ij} = 0$. In addition, there is no self-loop i.e. $a_{ii} = 0$ for all $i \in \mathcal{V}$. The neighborhood set of agent i denoted as $\mathcal{M}_i$ includes agents from which it receives information. L and B are Laplacian and weighted connection matrix between leader and follower robots, respectively.

Notations: For any vector $\varepsilon = [\varepsilon_1, \varepsilon_2, ..., \varepsilon_n]^\top$, a) $\mathrm{sgn}(\varepsilon) = [\mathrm{sgn}(\varepsilon_1), \mathrm{sgn}(\varepsilon_2), ..., \mathrm{sgn}(\varepsilon_n)]^\top$, b) $|\varepsilon|^\eta = [|\varepsilon_1|^\eta, ..., |\varepsilon_n|^\eta]^\top$, and c) $|\varepsilon|^\eta\mathrm{sgn}(\varepsilon) = [|\varepsilon_1|^\eta\mathrm{sgn}(\varepsilon_1), ..., |\varepsilon_n|^\eta\mathrm{sgn}(\varepsilon_n)]^\top$ where parameter $\eta \in \mathcal{R}$. The norm used in the paper is defined as $\|\varepsilon\| = \sqrt{\varepsilon^\top\varepsilon}$. The expected value of parameters are based on random packet loss and delay and is denoted by $\underset{\delta_i,\rho_i}{\mathrm{E}}\{..\}$ which can be simply represented as $\mathrm{E}\{..\}$ $(\forall i,\, 1 \le i \le M)$.

Remark 1: An edge $(\mathcal{V}_i, \mathcal{V}_i)$ is a self-loop indicates that the agent i can obtain its own absolute value. The topology graph used here is simple as it contains neither a self-loop nor a repeated edge. For multi-agent consensus model, this means a follower does not show any self-reliability while following the leader, instead it updates its heading by its average heading summed up with the headings of neighbors [49]. □

C. System Dynamics and Control

The leader and follower robot dynamics prone to network uncertainties and external disturbances in MAS framework are:

$$\begin{aligned} \text{Follower:} \quad & \dot{\mathrm{x}}_i(t) = \rho_i^{\mathrm{u}}\mathrm{u}_i(t-\delta_i^{\mathrm{u}}) + \mathrm{d}_i(t), \quad i = 1\ldots p \\ \text{Leader:} \quad & \dot{\mathrm{x}}_0(t) = \rho_0^{\mathrm{u}}\mathrm{u}_0(t-\delta_0^{\mathrm{u}}), \end{aligned} \tag{8}$$

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

where $\mathrm{x}_i(t) \in \mathcal{R}^n, \mathrm{u}_i(t) \in \mathcal{R}^n$, and $\mathrm{d}_i(t) \in \mathcal{R}^n$ represent position, control input, and bounded disturbance input, respectively for the i^{th} follower robot. $\mathrm{x}_0(t) \in \mathcal{R}^n$ and $\mathrm{u}_0(t) \in \mathcal{R}^n$ are leader robot's position and control input, respectively. ρ_i^{u} and ρ_0^{u} are random packet loss coefficients for control input of follower and leader robots, respectively. δ_i^{u} and δ_0^{u} are random delays of follower and leader robot, respectively. Similarly, ρ_i^{x} and ρ_0^{x} are random packet loss coefficients and δ_i^{x} and δ_0^{x} are random delays of follower and leader robot states respectively. A simple discrete-time form of leader-follower robot dynamics (8) is given by

$$\begin{aligned} \text{Follower:} \quad \mathrm{x}_i(k+1) =& \mathrm{x}_{i(\delta_i,\rho_i)}(k) + \Delta T \mathrm{u}_i(k) + \mathrm{d}_i(k) \\ \text{Leader:} \quad \mathrm{x}_0(k+1) =& \mathrm{x}_{0(\delta_0,\rho_0)}(k) + \Delta T \mathrm{u}_0(k), \end{aligned} \tag{9}$$

where ΔT is sampling interval of the system. The states $\mathrm{x}_{i(\delta_i,\rho_i)}$ and $\mathrm{x}_{0(\delta_0,\rho_0)}$ are due to random delays and packet losses on their respective current and previous states and previous control inputs. They are represented as

$$\begin{aligned} \mathrm{x}_{i(\delta_i,\rho_i)}(k) =& \{\rho_i^{\mathrm{x}}\mathrm{x}_i(k), \rho_i^{\mathrm{x}}\mathrm{x}_i(k-1), \ldots, \rho_i^{\mathrm{x}}\mathrm{x}_i(k-\delta_i^{\mathrm{x}}), \\ & \rho_i^{\mathrm{u}}\mathrm{u}_i(k-1), \rho_i^{\mathrm{u}}\mathrm{u}_i(k-2), \ldots, \rho_i^{\mathrm{u}}\mathrm{u}_i(k-\delta_i^{\mathrm{u}})\} \\ \mathrm{x}_{0(\delta_0,\rho_0)}(k) =& \{\rho_0^{\mathrm{x}}\mathrm{x}_0(k), \rho_0^{\mathrm{x}}\mathrm{x}_0(k-1), \ldots, \rho_0^{\mathrm{x}}\mathrm{x}_0(k-\delta_0^{\mathrm{x}}), \\ & \rho_0^{\mathrm{u}}\mathrm{u}_0(k-1), \rho_0^{\mathrm{u}}\mathrm{u}_0(k-2), \ldots, \rho_0^{\mathrm{u}}\mathrm{u}_0(k-\delta_0^{\mathrm{u}})\}. \end{aligned} \tag{10}$$

A stochastic model for (9) is defined below taking into account expected value of states in (10) due to network uncertainties

$$\begin{aligned} \mathrm{E}\{\mathrm{x}_{i,k+1}\} =& \mathrm{E}\{\mathrm{x}_{i,k}\} + \Delta T \mathrm{u}_{i,k} + \mathrm{d}_{i,k} \\ \mathrm{E}\{\mathrm{x}_{0,k+1}\} =& \mathrm{E}\{\mathrm{x}_{0,k}\} + \Delta T \mathrm{u}_{0,k}, \end{aligned} \tag{11}$$

where $\mathrm{E}\{\mathrm{x}_{i,k}\}$ represents expected value of $\mathrm{x}_{i(\delta_i,\rho_i)}$. Let

$$\begin{aligned} \mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\} =& \mathrm{E}\{\mathrm{x}_{i,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_i \\ \text{and} \quad \tilde{\mathrm{u}}_{i,k} =& \mathrm{u}_{i,k} - \mathrm{u}_{0,k}, \end{aligned} \tag{12}$$

where $\mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\}$ and $\tilde{\mathrm{u}}_{i,k}$ are deviations in position and control input of follower robot i from leader robot. Γ_i is the desired deviation in position of follower i from the leader. The relative dynamics using (11) can be written as

$$\mathrm{E}\{\tilde{\mathrm{x}}_{i,k+1}\} = \mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\} + \Delta T \tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k}. \tag{13}$$

The following sliding mode surface is defined for follower agent i in the MAS framework to design controller,

$$\mathrm{E}\{\mathrm{S}_{i,k}\} = \mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\} - \mathrm{E}\{q_{i,k}^{\alpha}\}, \quad i = 1, \ldots, M-1. \tag{14}$$

where $\mathrm{S}_{i,k} = [s_{1,k}^i, s_{2,k}^i, \ldots s_{n,k}^i]^\top$. With the onset of sliding phase, $\mathrm{S}_{i,k} = 0$. To achieve desired consensus, we have considered a super-twisting control law

$$\begin{aligned} \tilde{\mathrm{u}}_{i,k} =& \mathrm{E}\{q_{i,k}^{\alpha}\} - \mathrm{K}_1 \mathrm{E}\{|\mathrm{S}_{i,k}|^{\frac{1}{2}} \mathrm{sgn}(\mathrm{S}_{i,k})\} + \mathrm{E}\{\varrho_{i,k}\} \\ \text{with } \mathrm{E}\{\varrho_{i,k+1}\} =& \mathrm{E}\{\varrho_{i,k}\} - \Delta T\, \mathrm{K}_2 \{\mathrm{sgn}(\mathrm{E}\{\mathrm{S}_{i,k}\})\}, \end{aligned} \tag{15}$$

where

$$\begin{aligned} \mathrm{E}\{q_{i,k}\} =& -\frac{\gamma_i}{m_i+1} \sum_{j \in M} a_{ij} \{(\mathrm{E}\{\mathrm{x}_{i,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_i) \\ & - (\mathrm{E}\{\mathrm{x}_{j,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_j)\} \\ & + b_i \big(\mathrm{E}\{\mathrm{x}_{i,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_i\big) \\ =& -\frac{\gamma_i}{m_i+1} \sum_{j \in M} a_{ij} \big(\mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\} - \mathrm{E}\{\tilde{\mathrm{x}}_{j,k}\}\big) + b_i \mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\}. \end{aligned} \tag{16}$$

The consensus component $q_{i,k}$ is defined based on tracking error of follower i and neighboring agents with respect to the leader. In (15) and (16), $\gamma_i > 0$, $\alpha \in (0,1)$, and m_i $(1 \leq m_i < M)$ is the count of neighboring agents of i^{th} member. The selection of gain parameters follows the constraint that they should be positive, i.e., $k_{ij} > 0, \forall i, j$, when system is not perturbed. On the other hand, if system is perturbed then it is not possible to retain the system performance only by selecting the gains to be positive. To handle this situation, the gain parameter should be chosen in following manner, i.e., $\mathrm{K}_1 = \mathrm{diag}\{k_{11}, k_{12}, \ldots k_{1n}\} = 1.5\sqrt{\bar{\Omega}}$ and $\mathrm{K}_2 = \mathrm{diag}\{k_{21}, k_{22}, \ldots k_{2n}\} = 1.1\bar{\Omega}$ where $\bar{\Omega}$ is as per Assumption 1 mentioned later in this section. The choice of such gain parameters can be found in [36].

Remark 2. Let us assume there is a self-loop ($a_{ii} \neq 0$) for agent i in the topology. Hence, $\mathrm{E}\{q_{i,k}\}$ in (16) is modified as

$$\begin{aligned} \mathrm{E}\{q_{i,k}\} =& -\frac{\gamma_i}{m_i+1} \sum_{j=1}^{M} a_{ij} \{(\mathrm{E}\{\mathrm{x}_{i,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_i) \\ & - \mathrm{E}\{\mathrm{x}_{j,k}\} + \mathrm{E}\{\mathrm{x}_{0,k}\} - \Gamma_j\} + b_i \big(\mathrm{E}\{\mathrm{x}_{i,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_i\big) \\ & + a_{ii} \underbrace{(\mathrm{E}\{\mathrm{x}_{i,k}\} - \mathrm{E}\{\mathrm{x}_{0,k}\} + \Gamma_i - \mathrm{E}\{\mathrm{x}_{i,k}\} + \mathrm{E}\{\mathrm{x}_{0,k}\} - \Gamma_i)}_{=0}. \end{aligned} \tag{17}$$

In (17), the relative tracking error component due to self-loop turns out to be 0. Thus, analytically consensus component does not need self-loop in its topology. □

Remark 3.. The STSMC mitigates chattering phenomenon to a large extent. The term $\mathrm{K}_1 \mathrm{E}\{|\mathrm{S}_{i,k}|^{\frac{1}{2}} \mathrm{sgn}(\mathrm{S}_{i,k})\}$ is primarily responsible for reaching the sliding manifold in finite time while the term $\mathrm{E}\{\varrho_{i,k}\}$ attenuates the chattering effect significantly but not completely. □

Remark 4. In this work, data packet dropout (θ) occurs as per the following probability distribution:

$$\Pr\{\theta = 1\} = p, \ \Pr\{\theta = 0\} = 1 - p, \tag{18}$$

where $0 \leq p < 1$ is probability for loss of a data packet. □

To devise the event-triggering approach, a novel measurement error termed as event-trigger error ($e_{i,k}$) for agent i is introduced here. The event-trigger error is defined as

$$e_{i,k} = \mathrm{E}\{\mathrm{S}_{i,k}\} - \mathrm{E}\{\mathrm{S}_{i,k_\epsilon^i}\} = \begin{bmatrix} \mathrm{E}\{s_{1,k}^i\} - \mathrm{E}\{s_{1,k_\epsilon^i}^i\} \\ \vdots \\ \mathrm{E}\{s_{n,k}^i\} - \mathrm{E}\{s_{n,k_\epsilon^i}^i\} \end{bmatrix}, \tag{19}$$

where $\mathrm{S}_{i,k}$ and $\mathrm{S}_{i,k_\epsilon^i}$ are sliding surfaces at current instant (k) and last event-instant (k_ϵ^i) respectively.

Remark 5. The superscript ($i = 1, .., M$) and subscript ($\epsilon = 1, 2, ..$) in k_ϵ^i are agent and event number respectively. The latest event-triggering instant for agent i is represented by $k_\epsilon^i (k_\epsilon^i \in k)$ whereas k represents regular (discrete) sampling instants of system operation. □

The control law $\tilde{\mathrm{u}}_{i,k}$ in (15) is updated only at events otherwise

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

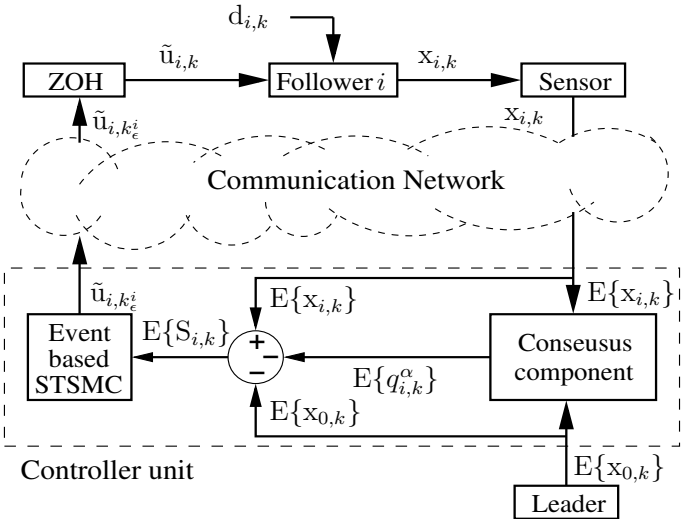


Fig. 2: A schematic diagram of the proposed approach.

it remains same. Hence, $\forall k\,(k^i_\epsilon \le k < k^i_{\epsilon+1})$ it is

$$\tilde{\mathrm{u}}_{i,k} = \mathrm{E}\{q^\alpha_{i,k^i_\epsilon}\} - \mathrm{K}_1\mathrm{E}\{|\mathrm{S}_{i,k^i_\epsilon}|^{\frac{1}{2}}\mathrm{sgn}(\mathrm{S}_{i,k^i_\epsilon})\} + \mathrm{E}\{\varrho_{i,k^i_\epsilon}\}. \quad (20)$$

Using (14), the sliding surface dynamics is expressed as

$$\mathrm{E}\{\mathrm{S}_{i,k+1}\} = \mathrm{E}\{\tilde{\mathrm{x}}_{i,k+1}\} - \mathrm{E}\{q^\alpha_{i,k+1}\}. \quad (21)$$

The schematic diagram of proposed approach is shown for a follower robot i in Fig 2. The sensor data from robot reaches the controller via a wireless communication network prone to packet losses and delay. The consensus component $q^\alpha_{i,k}$ is calculated based on position of leader and follower robots as in (16). The sliding surface is computed using consensus component, and deviation of leader and follower position as in (14). The control law as in (15) depends on sliding surface. Its update is event-based. The presence of a zero-order-hold component holds the last transmitted data until next transmission occurs. The two assumptions mentioned below are considered while devising the proposed approach.

Assumption 1. The unmodeled dynamics/disturbance lumped together as $\mathrm{d}_{i,k}$ and its change $\Delta\mathrm{d}_{i,k}(\Delta\mathrm{d}_{i,k} = \frac{\mathrm{d}_{i,k+1}-\mathrm{d}_{i,k}}{\Delta T})$ are bounded, i.e., $\|\mathrm{d}_{i,k}\| < \Omega$ and $sup\|\Delta\mathrm{d}_{i,k}\| < \bar{\Omega}$ where Ω and $\bar{\Omega} \in \mathcal{R}_{\ge 0}$.

Assumption 2. The deviation of component q^α_i between the last event and current time instant ($q^\alpha_{i,k^i_\epsilon} - q^\alpha_{i,k}$) is upper bounded as $\left\|q^\alpha_{i,k^i_\epsilon} - q^\alpha_{i,k}\right\| \le \Phi_i$ where $\Phi_i \in \mathcal{R}_{\ge 0}$.

D. Problems

This work contributes through solutions to three problems.

Problem 1: Given a MAS (8), find update instants k^i_ϵ for each agent i such that the event-based controller renders system stable in the presence of bounded external disturbance.

Problem 2: Find the upper bound on reaching time to the sliding surface for each agent?

Problem 3: Find the lower bound on admissibility or inter-execution time for update of control input of each agent?

The solutions of these problems are presented in next section.

III. Event-Triggered Super Twisting Formation Control for MAS

The proposed methodology has three parts. The first one derives the event-triggering condition for agents. The triggering condition is then used to derive the upper bound on reaching time to the sliding surface and lower bound on inter-execution time to prevent the zeno effect (i.e. accumulation of infinite events) in the next two subsections. This methodology ensures consensus based formation among agents and their stability.

A. Event-Triggering Condition

The Theorem 1 derives triggering condition for update of each agent's control input.

Theorem 1. Consider the MAS (11) with event-based control law for each agent in (20). If the condition

$$\left\|e_{i,k}\right\| > \Big\|\frac{1}{2}(\psi^\top - \beta^\top)^{-1}(2\psi^\top\beta - \psi^\top\psi - \beta^\top\beta) - \mathrm{E}\{\mathrm{S}_{i,k^i_\epsilon}\}\Big\|, \quad (22)$$

where $\psi = \mathrm{E}\{q^\alpha_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k}$, and $\beta = \mathrm{E}\{q^\alpha_{i,k+1}\}$ is satisfied then an event is triggered and control input for the agent i is updated at that instant. The gradual events leading to control input updates of each agent result in desired formation of agents and ensure stability in the presence of disturbance.

Proof: Consider a Lypunov function for agent i as

$$V_{i,k} = \mathrm{E}\{\mathrm{S}_{i,k}\}^T\mathrm{E}\{\mathrm{S}_{i,k}\}. \quad (23)$$

The first difference of Lyapunov function in (23) is

$$\Delta V_{i,k} = \mathrm{E}\{\mathrm{S}_{i,k+1}\}^T\mathrm{E}\{\mathrm{S}_{i,k+1}\} - \mathrm{E}\{\mathrm{S}_{i,k}\}^T\mathrm{E}\{\mathrm{S}_{i,k}\}. \quad (24)$$

Using (13) and (14), sliding surface dynamics in (21) becomes

$$\begin{aligned}\mathrm{E}\{\mathrm{S}_{i,k+1}\} &= \mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\}\\ &= \mathrm{E}\{\mathrm{S}_{i,k}\} + \mathrm{E}\{q^\alpha_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\}. \quad (25)\end{aligned}$$

On substituting (25) in (24),

$$\begin{aligned}\Delta V_{i,k} =& \Big(\mathrm{E}\{\mathrm{S}_{i,k}\} + \mathrm{E}\{q^\alpha_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\}\Big)^\top\\ &\Big(\mathrm{E}\{\mathrm{S}_{i,k}\} + \mathrm{E}\{q^\alpha_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\}\Big)\\ &- \mathrm{E}\{\mathrm{S}_{i,k}\}^\top\mathrm{E}\{\mathrm{S}_{i,k}\}. \quad (26)\end{aligned}$$

Let $\mathrm{E}\{q^\alpha_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} = \psi$ and $\mathrm{E}\{q^\alpha_{i,k+1}\} = \beta$. Hence, (26) can be rewritten as

$$\begin{aligned}\Delta V_{i,k} =& \Big(\mathrm{E}\{\mathrm{S}_{i,k}\} + \psi - \beta\Big)^\top\Big(\mathrm{E}\{\mathrm{S}_{i,k}\} + \psi - \beta\Big)\\ &- \mathrm{E}\{\mathrm{S}_{i,k}\}^\top\mathrm{E}\{\mathrm{S}_{i,k}\}\\ =& 2(\psi^\top - \beta^\top)\mathrm{E}\{\mathrm{S}_{i,k}\} + \psi^\top\psi - 2\psi^\top\beta + \beta^\top\beta. \quad (27)\end{aligned}$$

For sliding dynamics to be stable $\Delta V_{i,k} \le 0$. From (27)

$$2(\psi^\top - \beta^\top)\mathrm{E}\{\mathrm{S}_{i,k}\} + \psi^\top\psi - 2\psi^\top\beta + \beta^\top\beta \le 0. \quad (28)$$

On using the event-trigger error from (19) in (28)

$$e_{i,k} + \mathrm{E}\{\mathrm{S}_{i,k^i_\epsilon}\} \le \frac{1}{2}(\psi^\top - \beta^\top)^{-1}(2\psi^\top\beta - \psi^\top\psi - \beta^\top\beta). \quad (29)$$

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

On further simplifying (29) using Frobenius norm

$$\left\|e_{i,k}\right\| \leq \left\| \frac{1}{2}(\psi^\top - \beta^\top)^{-1}(2\psi^\top\beta - \psi^\top\psi - \beta^\top\beta) - \mathrm{E}\{\mathrm{S}_{i,k^i_\epsilon}\} \right\|. \quad (30)$$

Based on (30) one can ensure sliding dynamics to be stable (i.e. $\Delta V_{i,k} \leq 0$) until the norm of event-trigger error value $(e_{i,k})$ is less than or equal to the value of expression on right-hand side. If this condition is violated then it means that stability of agent i tends to become uncertain. The triggering of event leads to update of control input based on law in (20). The event-triggering condition can be therefore written as

$$\left\|e_{i,k}\right\| > \left\| \frac{1}{2}(\psi^\top - \beta^\top)^{-1}(2\psi^\top\beta - \psi^\top\psi - \beta^\top\beta) - \mathrm{E}\{\mathrm{S}_{i,k^i_\epsilon}\} \right\|. \quad (31)$$

The condition (31) is necessary for update of event-based control policy $(\tilde{\mathrm{u}}_{i,k})$ to drive the system towards stability. This completes the proof of Theorem 1. □

Remark 6. The proposed event-triggering approach captures the essence of using a small amount of information to stabilize the system. However, this small amount of information may lead to low convergence speed but the super-twisting SMC considered here provides an at par convergence speed as that of the time-triggered controller. It also provides a very high formation accuracy. For smooth real-time implementation of (31), the '$>$' is replaced by '$\geq$'. This condition can be evaluated easily as it is based on parameters such as consensus term $(q^\alpha_{i,k})$, control law $(\tilde{\mathrm{u}}_{i,k})$, sliding surface $(\mathrm{S}_{i,k^i_\epsilon})$, and disturbance $(\mathrm{d}_{i,k})$ which are either computable or readily available. Practically, it may be difficult to have exact knowledge of disturbance every time. However, this approach considers bounded disturbance to evaluate the triggering condition. □

Remark 7. This work does not consider following cases: (i) uncertainties in the form of system noise or channel noise and (ii) asynchronous or event-based data transmission from sensor to controller. The case (i) if considered, then the expected values of states and other system parameters would vary based on the nature of noise i.e. distribution function etc. This will also affect quality and value of control inputs to plant. The case (ii) when considered will require triggering conditions for data transmission from sensor to controller in addition to those for control updates. The event-based scenario will provide lesser state information to controller which will further change the frequency of control updates. □

B. Finite Reaching Time to Discrete Sliding Surface

For real-time applications, consensus should be achieved in finite time. This requires reaching time to the sliding phase be finite. In the sliding phase, the describing point of an agent moves along the chosen surface. Theorem 2 below presents an upper bound on finite reaching time.

Theorem 2. Consider the MAS dynamics (11) with the triggering condition (22) which leads to update of control input based on law (20). The event-based controller allows the sliding phase to start in finite time where the reaching time T_r to attain this phase is upper bounded by

$$T_r = \frac{\left\|\mathrm{S}_{i,k_0}\right\|}{S^\Delta_i}. \quad (32)$$

Proof: From (21), the sliding surface dynamics is

$$\mathrm{E}\{\mathrm{S}_{i,k+1}\} = \mathrm{E}\{\tilde{\mathrm{x}}_{i,k+1}\} - \mathrm{E}\{q^\alpha_{i,k+1}\}. \quad (33)$$

The equation (33) can be further written using (13) as

$$\mathrm{E}\{\mathrm{S}_{i,k+1}\} = \mathrm{E}\{\tilde{\mathrm{x}}_{i,k}\} + \Delta T\{\tilde{\mathrm{u}}_{i,k}\} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\}. \quad (34)$$

Using (14) and (34), the change in sliding surface value $(\Delta \mathrm{S}_{i,k} = \mathrm{E}\{\mathrm{S}_{i,k+1}\} - \mathrm{E}\{\mathrm{S}_{i,k}\})$ is

$$\Delta \mathrm{S}_{i,k} = \Delta T \tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\} + \mathrm{E}\{q^\alpha_{i,k}\}. \quad (35)$$

The sliding surface variable for agent i when it starts operation at $k = k_0$ is S_{i,k_0}. Once, it attains the sliding phase at $k = k_f$, $S_{i,k_f} = 0$. Using the Assumptions 1 and 2, $\Delta \mathrm{S}_{i,k}$ from (35) is upper bounded by

$$\left\|\Delta \mathrm{S}_{i,k}\right\| \leq \Delta T \left\|\tilde{\mathrm{u}}_{i,k}\right\| + \Omega + \Phi_i = S^\Delta_i. \quad (36)$$

Let the time taken by agent i to achieve the sliding phase $(\mathrm{S}_{i,k_f} = 0)$ is $T_r = k_f - k_0$. The maximum change in magnitude of sliding surface from S_{i,k_0} to S_{i,k_f} is

$$S^\Delta_i(k_f - k_0) = \left\|\mathrm{S}_{i,k_f} - \mathrm{S}_{i,k_0}\right\| = \left\|\mathrm{S}_{i,k_0}\right\|. \quad (37)$$

From (37), the upper bound on finite reaching time for agent i to the sliding surface is

$$T_r = \frac{\left\|\mathrm{S}_{i,k_0}\right\|}{S^\Delta_i}.$$

C. Admissibility of Event-Based Control

The admissibility of event-based control is necessary to avoid piling of triggering instants (zeno effect) which can make the system unstable. The Theorem 3 provides a lower bound for inter-event execution duration to avoid such scenario.

Theorem 3. For the system (11), control input is updated when an event is triggered. The admissibility of such control update is guaranteed if the inter-event execution time T_e ($T_e = k^i_{\epsilon+1} - k^i_\epsilon$) is lower bounded by

$$T^{min}_e(k) \geq \min_{k\in\mathbb{N}} \frac{\mu_{min} - \left\|e_{i,k^i_\epsilon}\right\|}{2\Phi_i + \Omega}. \quad (38)$$

Proof: Let the sliding surface value at latest event instant $k = k^i_\epsilon$ be represented as $\bar{\mathrm{S}}_{i,k} = \mathrm{S}_{i,k^i_\epsilon}$. The event-error dynamics $\forall k,\ k^i_\epsilon \leq k < k^i_{\epsilon+1}$ using (19) can be written as

$$\begin{aligned} e_{i,k+1} &= \mathrm{E}\{\mathrm{S}_{i,k+1}\} - \mathrm{E}\{\bar{\mathrm{S}}_{i,k+1}\} \\ &= \mathrm{E}\{\mathrm{S}_{i,k}\} + \mathrm{E}\{q^\alpha_{i,k}\} + \Delta T\tilde{\mathrm{u}}_{i,k} + \mathrm{d}_{i,k} - \mathrm{E}\{q^\alpha_{i,k+1}\} \\ &\quad - \mathrm{E}\{\bar{\mathrm{S}}_{i,k}\} - \mathrm{E}\{\bar{q}^\alpha_{i,k}\} - \Delta T\bar{\tilde{\mathrm{u}}}_{i,k} - \bar{\mathrm{d}}_{i,k} \\ &\quad + \mathrm{E}\{\bar{q}^\alpha_{i,k+1}\}. \end{aligned} \quad (39)$$

Using the event-trigger error from (19) in (39)

$$\begin{aligned} e_{i,k+1} = &e_{i,k} + (\mathrm{E}\{q^\alpha_{i,k}\} - \mathrm{E}\{\bar{q}^\alpha_{i,k}\}) + \Delta T(\tilde{\mathrm{u}}_{i,k} - \bar{\tilde{\mathrm{u}}}_{i,k}) \\ &+ (\mathrm{d}_{i,k} - \bar{\mathrm{d}}_{i,k}) + (\mathrm{E}\{\bar{q}^\alpha_{i,k+1}\} - \mathrm{E}\{q^\alpha_{i,k+1}\}). \end{aligned} \quad (40)$$

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

The control inputs $\tilde{\mathrm{u}}_{i,k}$ and $\bar{\tilde{\mathrm{u}}}_{i,k}$ for $\mathrm{S}_{i,k+1}$ and $\bar{\mathrm{S}}_{i,k+1}$, respectively are same until the next event occurs. Therefore

$$\Delta e_{i,k} = e_{i,k+1} - e_{i,k} = (\mathrm{E}\{q^{\alpha}_{i,k}\} - \mathrm{E}\{\bar{q}^{\alpha}_{i,k}\}) + (\mathrm{d}_{i,k} - \bar{\mathrm{d}}_{i,k}) + (\mathrm{E}\{\bar{q}^{\alpha}_{i,k+1}\} - \mathrm{E}\{q^{\alpha}_{i,k+1}\}). \quad (41)$$

Using the Assumptions 1 and 2 in (41)

$$\left\|\Delta e_{i,k}\right\| \leq 2\Phi_i + \Omega. \quad (42)$$

Using (42), the event-trigger error magnitude $\forall k(k^i_{\epsilon} \leq k < k^i_{\epsilon+1})$ can be computed as

$$\left\|e_{i,k^i_{\epsilon}}\right\| \leq \left\|e_{i,k}\right\| \leq \left\|e_{i,k^i_{\epsilon}}\right\| + (k - k^i_{\epsilon})(2\Phi_i + \Omega). \quad (43)$$

For $k^i_{\epsilon} < k < k^i_{\epsilon+1}$, inter-event time T_e is the time taken by $e_{i,k}$ to reach minimum threshold $(\mu_{min})\forall k \in \mathbb{N}$ in (22).

$$\mu_{min} = \left\|\frac{1}{2}(\psi^T - \beta^T)^{-1}(2\psi^T\beta - \psi^T\psi - \beta^T\beta) - \mathrm{E}\{\mathrm{S}_{i,k^i_{\epsilon}}\}\right\|. \quad (44)$$

From (43) and (44), the event is triggered only when

$$\left\|e_{i,k^i_{\epsilon}}\right\| + (k - k^i_{\epsilon})(2\Phi_i + \Omega) \geq \mu_{min}$$
$$\Rightarrow (k - k^i_{\epsilon})(2\Phi_i + \Omega) \geq \mu_{min} - \left\|e_{i,k^i_{\epsilon}}\right\| \quad (45)$$

From (45), the lower bound on inter-event time is

$$T^{min}_e(k) = \min_{k\in\mathbb{N}}\{T_e(k)\} \geq \min_{k\in\mathbb{N}} \frac{\mu_{min} - \left\|e_{i,k^i_{\epsilon}}\right\|}{2\Phi_i + \Omega}.$$

Remark 8. The minimum and maximum bounds on disturbance $\mathrm{d}_{i,k}$ are 0 and Ω, respectively. Similarly, these bounds on disturbance $(\bar{\mathrm{d}}_{i,k})$ at event instant $(k = k^i_{\epsilon})$ are also same. Hence, the difference $\left\|\mathrm{d}_{i,k} - \bar{\mathrm{d}}_{i,k}\right\|$ is always bounded by Ω.

Based on (43) and (45), it can be said that the event-trigger error less than $\left\|e_{i,k^i_{\epsilon}}\right\| + (k - k^i_{\epsilon})(2\Phi_i + \Omega)$ will not trigger an event which guarantees $\mu_{min} - \left\|e_{i,k^i_{\epsilon}}\right\| > 0$. □

IV. Case Study

This section presents results based on real-time experiments and simulations for the proposed approach using non-holonomic multi-robot system.

In event-driven designs, two approaches are generally followed. In the first approach, the first instant of operation may not necessarily be the first triggering instant. It depends on triggering condition. If the condition is satisfied at the first instant, then it will also be the first triggering instant otherwise the triggering instant will be different. The system will be driven by initial control input. In the second approach, the first instant is always considered as first triggering instant. No validation of triggering condition is required at this instant. However, the next triggering instant will depend on triggering condition. In our case study, for all experiments, we have considered the first instant of operation as the triggering instant (based on second approach).

A. Real-Time Experiment

The real-time experiments have been performed using three Pioneer P3-DX (two-wheeled differential driven) mobile robots. In this setup one robot is assigned as a leader and the other two robots as followers. The robots have sonar and position encoder. A unique IP address assigned to each robot assists in communicating with each other. The robot operating system (ROS) provides complete control over sensors and actuator of P3-DX mobile robot.

The weighted communication graph G in Fig. 3 has connectivity weights considered as 1. The Laplacian matrix $L = [0\ 0; -1\ 1]$ and connection between leader to follower $B = [1\ 0; 0\ 1]$ are considered.

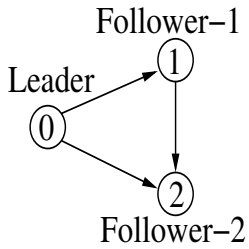


Fig. 3: A MAS digraph for three robot system.

The leader robot is capable of moving in any direction in obstacle-free environment. To have consensus during formation, follower robots should follow leader trajectory with desired separation. The initial position of leader is $\mathrm{x}_0 = [0; 0]$ whereas followers are initially at $\mathrm{x}_1 = [-2; 0.5]$ and $\mathrm{x}_2 = [-2; -0.5]$. The system parameters are as follows: $l = 0.21m, \mathrm{K}_1 = \mathrm{diag}\{1.25, 1.25\}, \mathrm{K}_2 = \mathrm{diag}\{0.75, 0.75\}, \Gamma_1 = (0.75m, -0.75m), \Gamma_2 = (0.75m, 0.75m)$, and $\alpha = 5/7$. The sampling time of operation is 0.01s. As per Assumption 1, values of disturbance bound (Ω) and its change $(\bar{\Omega})$ are considered as 2 and 0.4, respectively.

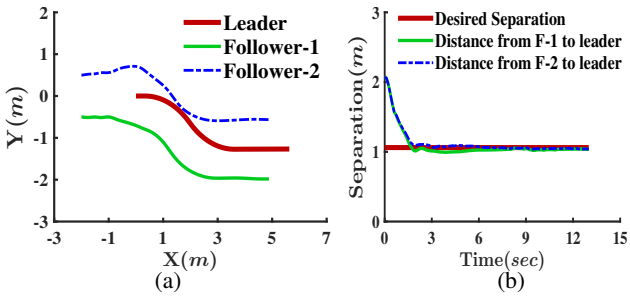


Fig. 4: (a) Trajectory of robots during desired formation and (b) Relative distance of followers from leader.

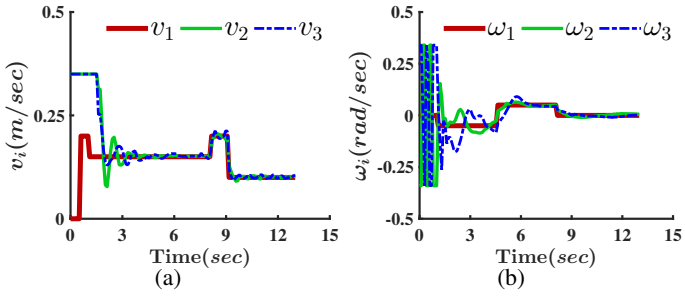
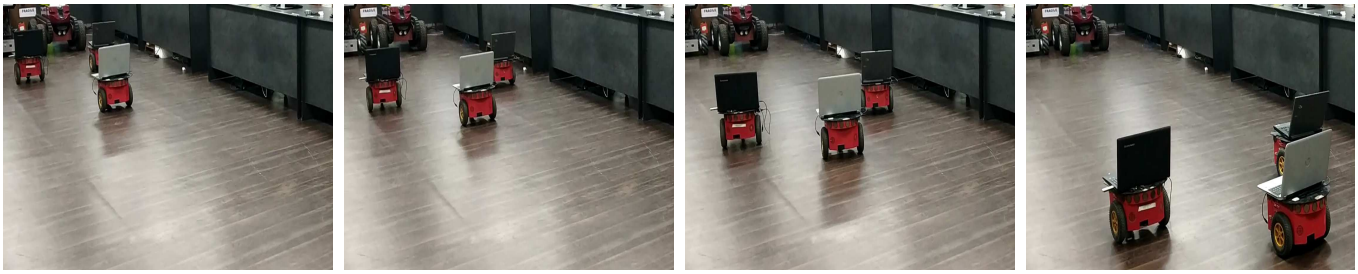


Fig. 5: (a) Linear velocity and (b) Angular velocity of robots.

The data communicated from robots are prone to delay and packet losses. The time-varying delay δ^{x}_i satisfies $0 \leq \delta^{\mathrm{x}}_i \leq 2$.

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems



(a) At initial position (b) At $t = 5$s (c) At $t = 9$s (d) At $t = 14$s

Fig. 6: Video image sequences (a)-(d) of the robots during formation.

The packet losses are modeled using Bernoulli distribution with probability $p = 0.15$.

The analyses without packet losses and delay are shown in Figs. 4-5. The trajectories of leader and followers are shown in Fig. 4a and their relative distances in Fig. 4b. From these figures it can be noticed that the distance between agents attain the desired value as time progresses. This implies that the agents achieve the desired formation in finite time. The linear (v) and angular (ω) velocities are shown in Fig. 5. The robustness of proposed approach has been verified for three different cases. The initial and other system parameters are same as mentioned above for all the three cases.

i) Case-1: The *packet losses and delay are considered only for position (state) information of follower robots*. The images at different time instants from the video of experiment have been shown in Fig. 6. The results during formation are presented in Figs. 7-10. The Fig. 7a clearly shows that the followers are able to achieve a consensus with respect to the leader. The separation between agents are large initially in Fig. 7b but gradually the event-based controller drives the agents to desired separation and ensures the same throughout the formation. The Fig. 8 shows that the velocity of followers try to emulate the leader's velocity. The proposed approach is robust as the follower robots adapt to the leader's velocity and the maneuvers despite the delays and packet losses.

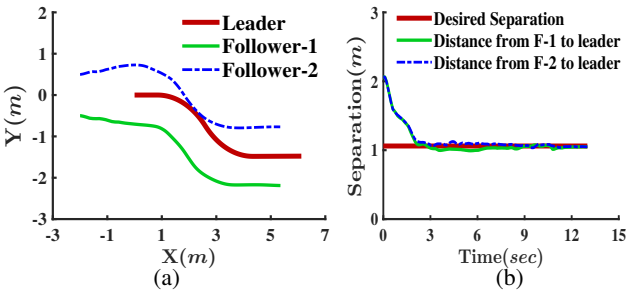


Fig. 7: (a) Trajectory of robots during desired formation and (b) Relative distance of follower robots from the leader robot.

The Fig. 9 shows the sliding surfaces ($\|S\|$) of the followers. The results indicate that the describing point of agents reach the sliding surfaces approximately in 1.8s. The event-trigger error and its threshold for follower robots 1 and 2 are depicted in Fig. 10. The respective control inputs are updated whenever the event-trigger error exceeds its threshold. The expression of upper bound on finite reaching time in (32) evaluates to 15.94s. The reaching time during real-time experiment (i.e.1.8s) is well below the upper bound. The admissibility of control updates prevent the zeno phenomenon during operation. The least inter-event time is 0.01s. The first sampling instant is considered as the first event instant without validating the condition (22). However, the second event occurs at $t = 0.02$s for each agent. This is based on the triggering condition. The values of event-triggering error (LHS value) in (22) are 0.2496 and 0.25 for Follower-1 and Follower-2, respectively, whereas the event-trigger error threshold (RHS value) for Follower-1 and Follower-2 are 0.1824, and 0.1828, respectively.

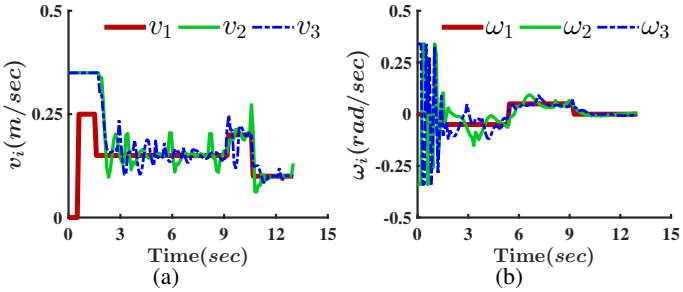


Fig. 8: (a) Linear velocity and (b) Angular velocity of robots.

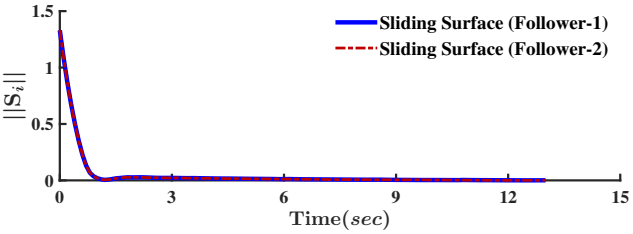


Fig. 9: Sliding surfaces of two follower robots.

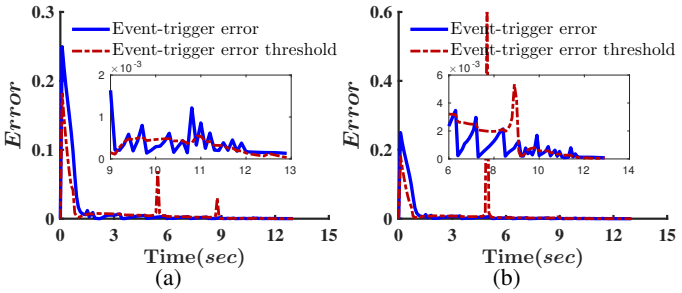


Fig. 10: Event-trigger error and threshold of (a) Follower robot-1 and (b) Follower robot-2.

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

ii) Case-2: The *packet losses and delays are considered only for position (state) information of leader and follower robots*. Figure 11 ensures that consensus based formation is achieved even when state information is prone to delay and packet losses. The effects of delay and packet losses on robot velocities can be noticed in Fig. 12. As packet losses and delay occur, controller tries to overcome their effect by changing magnitudes of respective velocities drastically.

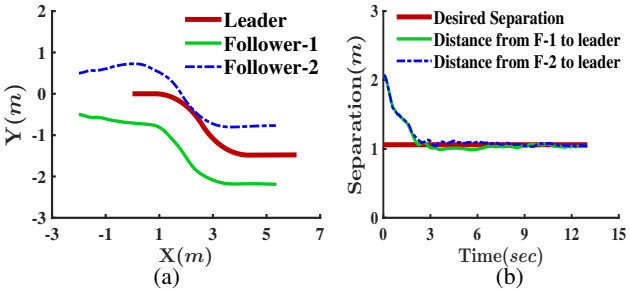


Fig. 11: (a) Trajectory of robots during desired formation and (b) Relative distance of follower robots from the leader.

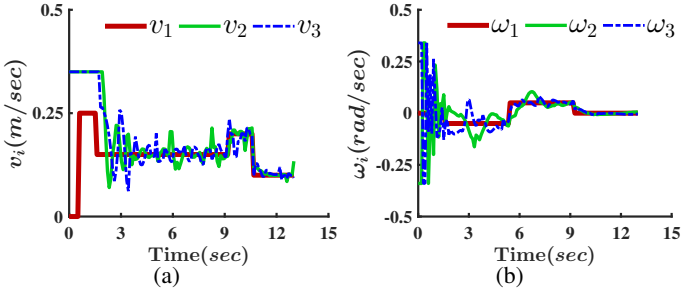


Fig. 12: (a) Linear velocity and (b) Angular velocity of robots.

iii) Case-3: This case considers *packet losses and delays for state and control input data of leader and follower robots*. The effect of network uncertainties is the worst in this case amongst the three cases. However, the proposed approach ensures formation which can be observed in Fig. 13. The changes in linear and angular velocities are much drastic than the Case-2. This can be seen in Fig. 14. The robot velocities change sharply when packet losses or delays or both occur.

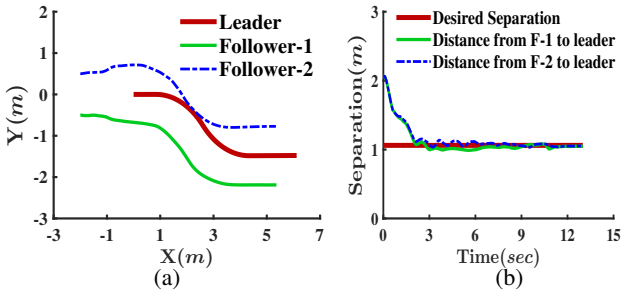


Fig. 13: (a) Trajectory of robots during desired formation and (b) Relative distance of follower robots from the leader robot.

The experimental validations of proposed approach are shown in [51]. The consensus in all these experiments is achieved approximately in $3-4$s. This confirms efficacy of methodology.

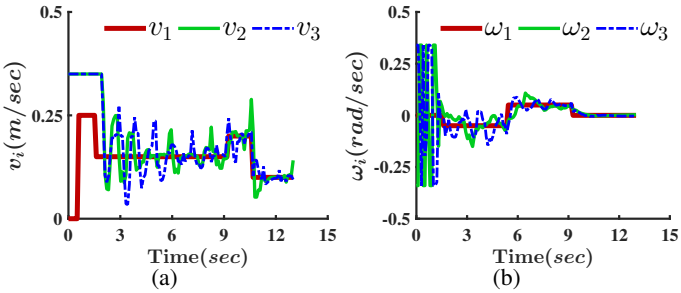


Fig. 14: (a) Linear velocity and (b) Angular velocity of robots.

B. Comparative Analysis

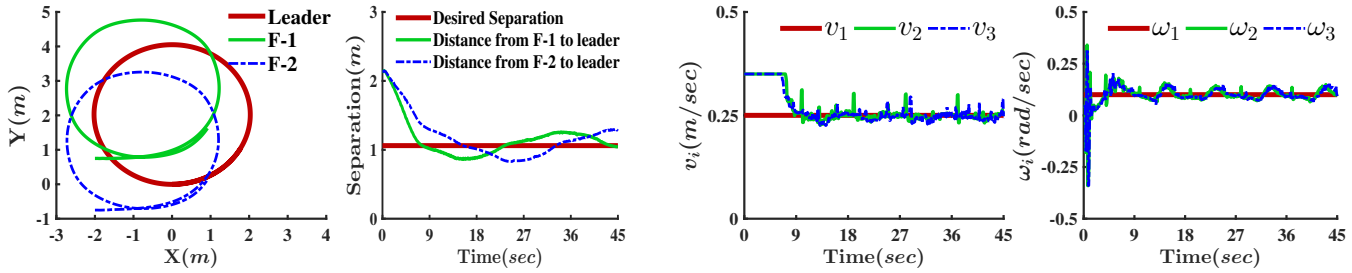
1) Comparison with approach in [25]: The comparative analysis of our methodology and that in [25] has been performed in Gazebo environment with three Pioneer P3-DX robots [30], [33]. The robots (one leader and two followers) try to achieve the desired consensus based formation of "circle". The initial position of leader is $\mathrm{x}_0 = [0; 0]$ where as the followers are initially at $\mathrm{x}_1 = [-2; 0.75]$ and $\mathrm{x}_2 = [-2; -0.75]$. The desired separation between leader and followers 1 and 2 should be $\Gamma_1 = (0.75m, -0.75m), \Gamma_2 = (0.75m, 0.75m)$, respectively. The linear (v_1) and angular (ω_1) velocities of leader are 0.25 m/sec and 0.1 rad/sec, respectively. The packet losses (ρ_i) are modeled using Bernoulli distribution with probability $p = 0.2$. The packet delay and other parameters are considered same as previous real-time experiments. The packet losses and delay are considered for follower robots' position (state) information only.

The results based on our methodology are shown in Fig. 15. The leader and follower trajectories are shown in Fig. 15a. The leader traces a circle-shaped trajectory whereas the followers follow leader's trajectory with desired separation. In the initial stage, the leader and followers are not in consensus. However, they achieve consensus and thereby desired formation after some time. Figure 15b verifies the distance between robots at desired separation. The desired formation is achieved in finite time in the presence of disturbance and network uncertainties. The Figs. 15c and 15d show the velocities of robots where followers try to emulate the leader's velocity. It can be noticed from the velocity profiles of followers that they have reduced chattering. The reason for this behaviour can be attributed to the use of the second order SMC which reduces chattering in the control inputs. The average triggering for follower-1 and follower-2 are 45.78% and 50.44%, respectively. These results prove the effective trajectory tracking even in the presence of uncertainty with less number of control inputs.

To compare the effectiveness of our approach, we have simulated the approach in [25]. The approach in [25] discusses first-order stochastic nonlinear system in strict-feedback form with finite time stability and closed loop solution through backstepping controller. The kinematic model of mobile robot used here is not in strict-feedback form. The model is first transformed to strict-feedback form using a chained model transformation [50]. The backstepping controller is then used in time-triggered manner. On comparing the trajectories of robots in Figs. 15a and 16a, the proposed approach performs better than [25]. The uncertainties influence the tra-

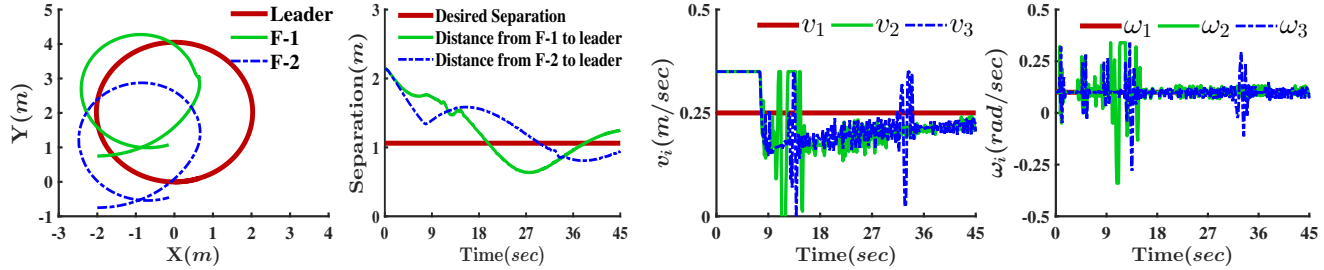
2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems



(a) (b) (c) (d)

Fig. 15: Result based on proposed approach : (a) Trajectory of robots during desired formation (b) Relative distance of followers from leader (c) Linear velocity and (d) Angular velocity of robots.



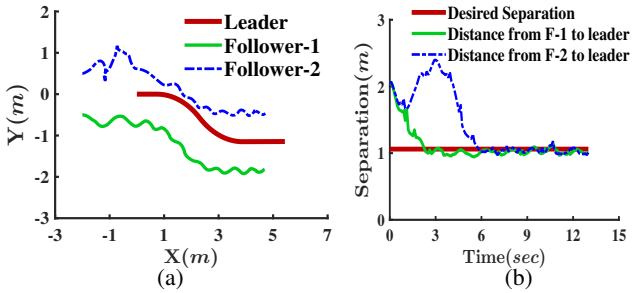
(a) (b) (c) (d)

Fig. 16: Result based on [25] : (a) Trajectory of robots during desired formation (b) Relative distance of followers from leader (c) Linear velocity and (d) Angular velocity of robots.

jectories of follower robots 1 and 2 at time $t = 11.2sec$ and $t = 8.1sec$, respectively. The Fig. 16b shows that the desired separations are not completely maintained throughout the robot trajectories. To check the consistency of both controllers, a performance criteria (average tracking error norm) is considered. The proposed approach has an average tracking error of $0.0140m$ and $0.0138m$ for follower-1 and follower-2, respectively. The average tracking error of follower-1 and follower-2 using backstepping controller [25] are $0.4915m$ and $0.44m$, respectively. This indicates that the proposed approach delivers better tracking accuracy.

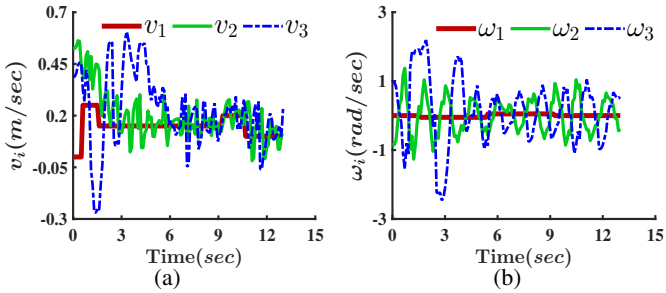
The linear and angular velocities in Figs. 16c and 16d show that the velocities of followers cannot emulate the leader velocities when prone to system and network uncertainties. We can draw a conclusion here for the comparison with [25]. Though the results shown in 16a-16d are through time-triggered implementation, results shown in 15a-15d obtained through event-triggered implementation are better. The results for [25] may be even worse for event-triggered implementation. The videos of simulations can be found in a video on You Tube at [51].

2) Comparison with approach in [32]: An another comparison but in real-time of our approach with the one presented in [32] for **Case-1** as mentioned earlier is done. The control law used in [32] is defined as $\mathrm{u}_i(t) = q_i^{\alpha}(t) - \mathrm{K}_1 \mathrm{sign}(\mathrm{S}_i(t)) - \mathrm{K}_2 \mathrm{S}_i(t)$. The triggering function used is $f_i(t) = \left\| e_i(t) \right\| - \rho_i$ where event-trigger error is $e_i(t) = \mathrm{u}_i(t_k^i) - \mathrm{u}_i(t)$. The other parameters are $\alpha \in (0.5, 1)$ and ρ (a positive constant). The initial position of leader is $\mathrm{x}_0 = [0\,;\,0]^T$ whereas followers are initially at $\mathrm{x}_1 = [-2\,;\,0.5]^T$ and $\mathrm{x}_2 = [-2\,;\,-0.5]^T$. The data delay and losses are considered same as that for the proposed approach.



(a) (b)

Fig. 17: Results based on [32]: (a) Trajectory of robots and (b) Relative distance of followers from leader.



(a) (b)

Fig. 18: Velocities based on approach in [32]: (a) Linear velocity and (b) Angular velocity.

On comparing the trajectories of robots in Figs. 7a and 17a, the proposed methodology provides a smoother trajectory than [32]. In Fig. 18, we can observe that the system model when affected by network uncertainties, linear and angular velocities are not smooth for [32] whereas our approach provides a smooth velocity response as shown in Fig. 8.

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

Table I presents average percentages of events triggered during the operation. Though minimum inter-event time is same for both the approaches, the average triggers are lesser as compared to the approach [32]. This confirms that the methodology presented in this paper performs well in presence of network uncertainties.

TABLE I: Comparison of all the approaches

	Proposed Approach		Approach [32]	
Parameter	F-1	F-2	F-1	F-2
Avg. events triggered (in %)	33.84	33.07	58.00	44.66
Min. inter-event time (in s)	0.01	0.01	0.01	0.01

Note. F-1: Follower-1 and F-2: Follower-2.

On the basis of theoretical and experimental validations, we conclude that the proposed event-driven STSMC approach is a good choice to handle consensus based formation problem in the presence of disturbance and network uncertainties.

V. Conclusion

A stochastic finite-time event-triggered second order sliding mode control design for MAS subjected to network uncertainties and external disturbances to achieve consensus based formation has been discussed in this paper. An event-triggering condition for each agent has been derived using discrete-time sliding surface. The closed-loop finite-time stability has been ensured using Lyapunov stability theory. The zeno behavior during formation is tackled by satisfying the admissibility criteria. To show efficacy of the proposed methodology, simulation and real-time experiments using three robots have been carried out. The experimental results demonstrate the superiority of proposed method over traditional time-triggered approach and the existing event-based approach. The count of control updates to achieve desired formation is significantly reduced because of lesser number of events. Consequently, the reduced use of computational and communication resources has been guaranteed during operation.

However, this work does not consider uncertainties in the form of system noise, channel noise etc. They may deteriorate the system performance from desired objective and may even cause the system to be unstable. The proposed work also does not consider the asynchronous or event-based data transmission from the sensor to controller. The future work, therefore, will consider the above mentioned limitations in the proposed methodology.

References

[1] W. Ren, R. W. Beard, and E. M. Atkins, "A Survey of Consensus Problems in Multi-Agent Coordination," in *Proc. Amer. Contr. Conf.*, Dec. 2005, pp. 1859–1864.

[2] H. Sun, Y. Liu, F. Li, and X. Niu, "Distributed LQR Optimal Protocol for Leader-Following Consensus," *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3532–3546, Sept. 2019.

[3] S. De, S. R. Sahoo, and P. Wahi, "Communication-Delay-Dependent Rendezvous With Possible Negative Controller Gain in Cyclic Pursuit," *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 3, pp. 1069–1079, Sept. 2020.

[4] E. Semsar-Kazerooni and K. Khorasani, "Optimal Consensus Seeking in a Network of Multiagent Systems: An LMI Approach," *IEEE Trans. Syst.,Man, Cybern. B, Cybern.*, vol. 40, no. 2, pp. 540–547, Apr. 2010.

[5] F. Xiao and T. Chen, "Adaptive Consensus in Leader-Following Networks of Heterogeneous Linear Systems," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1169–1176, Sept. 2018.

[6] M. Zhu and S. Martínez, "On the Convergence Time of Asynchronous Distributed Quantized Averaging Algorithms," *IEEE Trans. Autom. Control*, vol. 56, no. 2, pp. 386–390, Feb. 2011.

[7] H. Du, G. Wen, G. Chen, J. Cao, and F. E. Alsaadi, "A Distributed Finite-Time Consensus Algorithm for Higher-Order Leaderless and Leader-Following Multiagent Systems," *IEEE Trans. Syst., Man, Cybern., Syst*, vol. 47, no. 7, pp. 1625–1634, Jul. 2017.

[8] N. K. Dhar, N. K. Verma, and L. Behera, "Adaptive Critic-Based Event-Triggered Control for HVAC System," *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 178–188, Jan. 2018.

[9] X. Chen, F. Hao, and B. Ma, "Periodic Event-Triggered Cooperative Control of Multiple Non-holonomic Wheeled Mobile Robots," *IET Control Theory Appl.*, vol. 11, no. 6, pp. 890–899, Apr. 2017.

[10] Y.-Y. Qian, L. Liu, and G. Feng, "Output Consensus of Heterogeneous Linear Multi-Agent Systems with Adaptive Event-Triggered Control," *IEEE Trans. Autom. Control*, vol. 64, no. 6, pp. 2606–2613, Jun. 2019.

[11] T.-H. Cheng, Z. Kan, J. R. Klotz, J. M. Shea, and W. E. Dixon, "Event-Triggered Control of Multiagent Systems for Fixed and Time-Varying Network Topologies," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 5365–5371, Oct. 2017.

[12] A. Adaldo, F. Alderisio, D. Liuzza, G. Shi, D. V. Dimarogonas, M. Di Bernardo, and K. H. Johansson, "Event-Triggered Pinning Control of Switching Networks," *IEEE Trans. Control Netw. Syst.*, vol. 2, no. 2, pp. 204–213, Jun. 2015.

[13] Y. Cui, Y. Liu, W. Zhang, and F. E. Alsaadi, "Event-Based Consensus for a Class of Nonlinear Multi-Agent Systems With Sequentially Connected Topology," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 10, pp. 3506–3518, Oct. 2018.

[14] C. Peng, J. Zhang, and Q.-L. Han, "Consensus of Multiagent Systems With Nonlinear Dynamics Using an Integrated Sampled-Data-Based Event-Triggered Communication Scheme," *IEEE Trans. Syst., Man, Cybern.,Syst.*, vol. 49, no. 3, pp. 589–599, Mar. 2019.

[15] C. Du, X. Liu, W. Ren, P. Lu, and H. Liu, "Finite-Time Consensus for Linear Multiagent Systems via Event-Triggered Strategy without Continuous Communication," *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 1, pp. 19–29, Mar. 2020.

[16] V. Dolk, D. P. Borgers, and W. Heemels, "Output-Based and Decentralized Dynamic Event-Triggered Control With Guaranteed $\mathcal{L}_p$-Gain Performance and Zeno-Freeness," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 34–49, Jan. 2017.

[17] Q. Ling, "Periodic Event-Triggered Quantization Policy Design for a Scalar LTI System With i.i.d. Feedback Dropouts," *IEEE Trans. Autom. Control*, vol. 64, no. 1, pp. 343–350, Jan. 2019.

[18] V. Dolk and M. Heemels, "Event-triggered control systems under packet losses," *Automatica*, vol. 80, pp. 143–155, Jun. 2017.

[19] W.-A. Zhang and L. Yu, "Modelling and Control of Networked Control Systems With both Network-Induced Delay and Packet-Dropout," *Automatica*, vol. 44, no. 12, pp. 3206–3210, Dec. 2008.

[20] M. Guinaldo, D. Lehmann, J. Sánchez, S. Dormido, and K. H. Johansson, "Distributed event-triggered control with network delays and packet losses," in *Proc. 51st IEEE Conf. Decision Control*, Dec. 2012, pp. 1–6.

[21] C. Peng and Q.-L. Han, "On Designing a Novel Self-Triggered Sampling Scheme for Networked Control Systems With Data Losses and Communication Delays," *IEEE Trans. Ind. Electron.*, vol. 63, no. 2, pp. 1239–1248, Feb. 2016.

[22] T. Zhou, Z. Zuo, and Y. Wang, "Active Event-Triggered Control for Nonlinear Networked Control Systems With Communication Constraints," *IEEE Trans. Cybern.*, Apr. 2019, doi:10.1109/TCYB.2019.2907619.

[23] X. Tan, J. Cao, X. Li, and A. Alsaedi, "Leader-following mean square consensus of stochastic multi-agent systems with input delay via event-triggered control," *IET Control Theory Appl.*, vol. 12, no. 2, pp. 299–309, Jan. 2018.

[24] A. I. Rikos and C. N. Hadjicostis, "Distributed Integer Weight Balancing in the Presence of Time Delays in Directed Graphs," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1300–1309, Sept. 2018.

[25] H. Wang and Q. Zhu, "Finite-time stabilization of high-order stochastic nonlinear systems in strict-feedback form," *Automatica*, vol. 54, pp. 284–291, Apr. 2015.

[26] Q. Zhu and H. Wang, "Output feedback stabilization of stochastic feedforward systems with unknown control coefficients and unknown output function," *Automatica*, vol. 87, pp. 166–175, Jan. 2018.

[27] H. Wang and Q. Zhu, "Adaptive output feedback control of stochastic nonholonomic systems with nonlinear parameterization," *Automatica*, vol. 98, pp. 247–255, Dec. 2018.

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCNS.2021.3089142, IEEE Transactions on Control of Network Systems

[28] W. Hu, Q. Zhu, and H. R. Karimi, "Some improved razumikhin stability criteria for impulsive stochastic delay differential systems," *IEEE Trans. Autom. Control*, vol. 64, no. 12, pp. 5207–5213, Dec. 2019.

[29] A. Levant, "Sliding Order and Sliding Accuracy in Sliding Mode Control," *Int. J. Contr.*, vol. 58, no. 6, pp. 1247–1263, Dec. 1993.

[30] A. Nandanwar, R. R. Nair, and L. Behera, "Fuzzy Inferencing-Based Path Planning with a Cyber-Physical Framework and Adaptive Second-Order SMC for Routing and Mobility Control in a Robotic Network," *IET Cyber-Syst. Robot.*, vol. 2, no. 3, pp. 149–160, Sept. 2020.

[31] A. Sinha and R. K. Mishra, "Consensus in first order nonlinear heterogeneous multi-agent systems with event-based sliding mode control," *Int. J. Contr.*, vol. 93, no. 4, pp. 858–871, 2020.

[32] R. R. Nair, L. Behera, and S. Kumar, "Event-triggered finite-time integral sliding mode controller for consensus-based formation of multirobot systems with disturbances," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 1, pp. 39–47, Jan. 2019.

[33] A. Nandanwar, N. K. Dhar, D. Malyshev, L. Rybak, and L. Behera, "Finite-time robust admissible consensus control of multirobot system under dynamic events," *IEEE Syst. J.*, vol. 15, no. 1, pp. 780–790, Mar. 2021.

[34] S.-G. Wang, L. Bai, and M. Chen, "Robust Sliding Mode Control of General Time-Varying Delay Stochastic Systems with Structural Uncertainties," *Control Theory Technol.*, vol. 12, no. 4, pp. 357–367, 2014.

[35] A. Poznyak, "Stochastic super-twist sliding mode controller," *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1538–1544, May 2018.

[36] S. Koch and M. Reichhartinger, "Discrete-time equivalents of the super-twisting algorithm," *Automatica*, vol. 107, pp. 190–199, Sept. 2019.

[37] J. Xiong and J. Lam, "Stabilization of Networked Control Systems With a Logic ZOH," *IEEE Trans. Autom. Control*, vol. 54, no. 2, pp. 358–363, Feb. 2009.

[38] M. Yu, C. Yan, D. Xie, and G. Xie, "Event-triggered Tracking Consensus with Packet Losses and Time-varying Delays," *IEEE/CAA J. Autom. Sinica*, vol. 3, no. 2, pp. 165–173, Apr. 2016.

[39] Q. Zhu, "Stabilization of stochastic nonlinear delay systems with exogenous disturbances and the event-triggered feedback control," *IEEE Trans. Autom. Control*, vol. 64, no. 9, pp. 3764–3771, Sept. 2019.

[40] Z. Hu, P. Shi, J. Zhang, and F. Deng, "Control of discrete-time stochastic systems with packet loss by event-triggered approach," *IEEE Trans. Syst., Man, Cybern.,Syst.*, vol. 51, no. 2, pp. 755–764, Feb. 2021.

[41] W. Gao, Y. Wang, and A. Homaifa, "Discrete-time variable structure control systems," *IEEE Trans. Ind. Electron.*, vol. 42, no. 2, pp. 117–122, Apr. 1995.

[42] N. D. That and Q. P. Ha, "Discrete-Time Sliding Mode Control with State Bounding for Linear Systems With Time-Varying Delay and Unmatched Disturbances," *IET Control Theory Appl.*, vol. 9, no. 11, pp. 1700–1708, Jul. 2015.

[43] Y. Niu and D. W. Ho, "Design of Sliding Mode Control Subject to Packet Losses," *IEEE Trans. Autom. Control*, vol. 55, no. 11, pp. 2623–2628, Nov. 2010.

[44] L. Ma, Z. Wang, Y. Niu, Y. Bo, and Z. Guo, "Sliding Mode Control for a Class of Nonlinear Discrete-Time Networked Systems with Multiple Stochastic Communication Delays," *Int. J. Syst. Sci*, vol. 42, no. 4, pp. 661–672, 2011.

[45] J. Hu, Z. Wang, H. Gao, and L. K. Stergioulas, "Robust Sliding Mode Control for Discrete Stochastic Systems With Mixed Time Delays, Randomly Occurring Uncertainties, and Randomly Occurring Nonlinearities," *IEEE Trans. Ind. Electron.*, vol. 59, no. 7, pp. 3008–3015, Jul. 2012.

[46] A. Argha, L. Li, S. W. Su, and H. Nguyen, "Discrete-Time Sliding Mode Control for Networked Systems with Random Communication Delays," in *Proc. Amer. Control Conf.*, 2015, pp. 6016–6021.

[47] J. Hu, H. Zhang, X. Yu, H. Liu, and D. Chen, "Design of Sliding-Mode-Based Control for Nonlinear Systems With Mixed-Delays and Packet Losses Under Uncertain Missing Probability," *IEEE Trans. Syst. Man Cybern. Syst.*, Jun. 2019, doi:10.1109/TSMC.2019.2919513.

[48] D. Yao, B. Zhang, P. Li, and H. Li, "Event-Triggered Sliding Mode Control of Discrete-Time Markov Jump Systems," *IEEE Trans. Syst., Man, Cybern.,Syst.*, vol. 49, no. 10, pp. 2016–2025, Oct. 2019.

[49] J. Shao, W. X. Zheng, T.-Z. Huang, and A. N. Bishop, "On Leader–Follower Consensus With Switching Topologies: An Analysis Inspired by Pigeon Hierarchies," *IEEE Trans. Autom. Control*, vol. 63, no. 10, pp. 3588–3593, Oct. 2018.

[50] B. Ning and Q.-L. Han, "Prescribed Finite-Time Consensus Tracking for Multiagent Systems With Nonholonomic Chained-Form Dynamics," *IEEE Trans. Autom. Control*, vol. 64, no. 4, pp. 1686–1693, Apr. 2019.

[51] Stochastic Event-Based Super-Twisting Formation Control for Multi-Agent System under Network Uncertainties. [Online]. Available: https://youtu.be/cXIXSx608B0



Anuj Nandanwar received the master's degree in Electronics and Telecommunication from CSVTU, India. He is currently working toward the Ph.D. degree in control and automation in the Department of Electrical Engineering, Indian Institute of Technology Kanpur, Kanpur India. His primary research interests include cyber-physical systems, cooperative control of multi-robot systems, sliding mode control, intelligent control, event-trigger control and fault-tolerance controller.



Narendra Kumar Dhar received the Ph.D. degree from the Department of Electrical Engineering,IIT Kanpur, Kanpur, India, in 2020. He is currently working as a Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA. His main research interests include intelligent control, cyber–physical systems, connected and autonomous vehicles, optimal control, neural networks, and soft computing.



Dmitry Malyshev graduated from Belgorod State Technological University named after V.G. Shukhov in 2015, specializing in mechanical engineering technology. Since 2015, he has been working at the Center for High Technologies at BSTU named after V.G. Shukhov. His research interests include a wide range of issues related to numerical optimization and simulation methods as applied to robots and manipulators of various structures.



Larisa Rybak graduated with honors from the Kharkov Aviation Institute in 1985. She received the Ph.D. degree in the theory of mechanisms and machines from the Moscow Academy of Instrument Engineering in 1992 and Ph.D. Hab degree in 1998. Since 2004 she has been working in the BSTU named after V.G. Shukhov as a professor and head of the Scientific Laboratory of Mechatronics and Robotics. Her research interests include mechanics of machines and robots, intelligent control systems for machines and robots, mathematical simulation, optimization and computational methods.



Laxmidhar Behera is currently working as a Professor with IIT Kanpur, Kanpur, India, having research and teaching experience of more than 24 years. He has also worked as a Visiting Researcher/Professor with FHG, Germany, and ETH Zurich, Switzerland. His primary research interests include the convergence of machine learning, control theory, robotic vision, and heterogeneous robotic platforms.Dr. Behera has established industrial collaboration with TCS, Renault Nissan, BEL, Bengaluru, and ADNOC, Abu Dhabi. He is a fellow of the INAE.

2325-5870 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on July 01,2021 at 10:31:15 UTC from IEEE Xplore. Restrictions apply.