Cooperative Path Following of Robotic Vehicles using an Event based Control and Communication Strategy

R. Praveen Jain, A. Pedro Aguiar, and João Borges de Sousa

Abstract— **In this paper, a novel approach to the Cooperative Path Following (CPF) control problem is proposed that aims to reduce the frequency of communication between the robotic vehicles while achieving cooperation. Specifically, a decentralized, event-based cooperative controller is designed to achieve coordination between the robotic vehicles, tasked to follow a priori specified reference geometric path while maintaining a desired formation. The event-based cooperative controller transmits the necessary information at discrete event times, that are computed based on an event triggering condition designed such that the convergence and stability properties of the consensus controller is preserved. The Input-to-State Stability framework is utilized to prove the stability and convergence of the CPF control with event-based control and communication scheme. The proposed method is experimentally validated using Autonomous Surface Vehicles communicating over Wi-Fi. It is shown that the event-based approach results in significant reduction of information exchange between the vehicles when compared to the conventional periodic transmission.**

I. Introduction

A. Motivation

One of the major critical points concerning the formation control of autonomous robotic vehicles is the communication system and its influence on overall performance of the formation. Many of the algorithms proposed in the literature do not address this issue explicitly and assume an uninterrupted, usually high-bandwidth communication between the robots to achieve cooperation. Such a requirement is in general not amenable in practice due to limited bandwidth of the underlying communication medium which usually is a shared resource. Further, this fact can lead to degradation of the Quality-of-Service of the communication medium leading to unwanted effects such as data loss and delay. Moreover, such effects could potentially compromise with the stability of the overall system. It is therefore, necessary to develop techniques that judiciously utilize the shared resources such as communication channels in a networked multi-robot system. Furthermore, such methods gain prominence in marine applications where the underwater communication is achieved through acoustic medium, which is notorious for low bandwidth and communication delays. In this paper, we propose and demonstrate a formation control approach for a group of Autonomous Surface Vehicles (ASVs) with special emphasis on techniques that reduce the frequency of transmission between the ASVs over the network. The research presented in this paper, would pave the way for developing efficient formation control solutions for marine applications, where acoustic communication medium could be used.

Among many possible alternatives to solve the formation control problem, we investigate a method termed Cooperative Path Following (CPF) problem [1]. CPF decomposes the problem into a two layered control structure, where one layer is responsible for the motion control of the individual robotic vehicle, called the Path Following (PF) controller (See [2] and references therein). The other layer, which consists of a consensus law, termed the Cooperative Controller, is responsible for achieving coordination between the robots. In this paper, we apply the CPF control strategy with particular focus on the cooperative control layer, where a decentralized, event-based cooperative controller is proposed to reduce the frequency of transmission over the network while guaranteeing the stability and convergence of the overall CPF system.

B. Related Work

CPF control of robotic vehicles involve extensive inter-vehicle information exchange which is undesirable, and perhaps not necessary in practice. Consensus algorithms [3] are at the core of cooperative control applications such as CPF, where the robots exchange information over the network, in order to agree on a certain variable of interest. However, such algorithms rely on the assumption that constant, inter-robot communication is possible. One possible strategy to overcome such issue is to adopt event-based sampling techniques namely, the event-triggered control [4] and self-triggered control [5]. Research on event-based consensus methods have been at the foreground recently with the pioneering work of [6], where both centralized and decentralized versions were discussed. In particular, the efforts were on reducing the controller updates on every agent through the use of a state dependent Event Triggering Condition (ETC) which decides the time instances when the events are generated. The proposed ETC, in order to be computed, requires a ceaseless information from the neighboring agents, resulting in a continuous communication. A time dependent ETC was proposed in [7], which aimed to reduce the frequency of communication between the agents with a limitation that the agents are aware of the second eigenvalue of the connected graph Laplacian modeling the underlying communication topology. Further work includes [8], [6] where a decentralized self-triggered control strategy was used to pre-determine

*This project has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 642153.

R. Praveen Jain, A. Pedro Aguiar and João Borges de Sousa are with the Faculty of Electrical and Computer Engineering, University of Porto, 4200-465 Porto, Portugal {praveenjain, pedro.aguiar, jtasso}@fe.up.pt

Fig. 1: Three AUVs used in experiments.

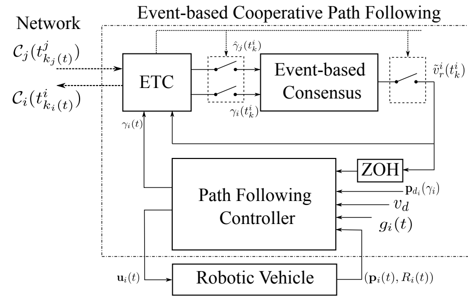


Fig. 2: Event-based CPF framework

the time instant at which the next event should be scheduled using the information available at the current time instant. In [9], a decentralized event-based communication and control method for average consensus is described, where a state dependent ETC is used to decide on the time instants at which the information needs to be transmitted. This method is suited for static consensus problems and requires modifications (change of coordinates) to adapt it to the dynamic consensus problem arising in CPF.

In the context of CPF, the work in [10] proposed a logic-based communication system, where each system has an internal filter of its own state and the states of its neighbors. Furthermore, the estimate of its owns state is synchronized with the estimate of its state contained in its neighbors. The communication logic consists of only transmitting information when the difference between the actual and the estimated state exceeds a certain threshold. As a result, communication occurs asynchronously at discrete instants of time. In a recent work [11], the authors considered a self-triggered CPF method to reduce the frequency of communication over the network. However, the proposed method was validated only in simulations and make use of a relatively complicated algorithm to determine the next event time instant. In this paper, we deviate from our previous work and present an event-based CPF method with a simplified event-based communication scheme. Further, Input-to-State Stability (ISS) framework [12] is used to provide formal guarantees of stability and convergence and experimental validation is presented using three Autonomous Underwater Vehicles (AUVs) shown in Figure 1 operating in the “ASV” mode, communicating over Wi-Fi. It is shown that the event-based approach results in significant reduction of information exchange between the vehicles, compared to the conventional periodic transmission.

C. Contributions

In this paper, a novel approach to the CPF problem using an event-based cooperative controller is proposed that leads to significant reduction in the inter-vehicle communication when compared with the traditional periodic transmission methods. Using some of the ideas of the event-triggered consensus controller presented in [8] (that assumes uninterrupted communication), we derive a simpler event-based communication strategy with formal convergence guarantees in the presence of a known exogenous input acting on the dynamics of the states that needs to be coordinated, and apply the proposed event-based control and communication method to the CPF problem, where we demonstrate its efficacy through experimental results. The experimental results are backed with the stability and convergence analysis.

D. Notation and organization of the paper

The Euclidean norm and the induced matrix norm is denoted by $\|.\|$. Set of N robotic vehicles is defined as $\mathcal{I} = \{1, 2, \cdots, N\}$. The Laplacian matrix of the undirected graph modeling the communication topology of the robots is denoted by L and $\lambda_2(L)$ denotes the second eigenvalue of L. $\mathbf{1}_N$ and $\mathbf{0}_N$ are the N-dimensional column vector with all entries 1 and 0, respectively. The set of non-negative integers is denoted by $\mathbb{Z}_{\geq 0}$. The set of neighbors of robot i is given by $\mathcal{N}_i$ and its cardinality is given by $|\mathcal{N}_i|$. The i^{th} element of a vector is denoted by $[\ .\ ]_i$. A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. Additionally it is said to belong to class $\mathcal{K}_\infty$ if $a = \infty$ and $\alpha(r) \to \infty$ as $r \to \infty$. A continuous function $\beta : [0, a) \times [0, \infty) \to [0, \infty)$ is said to belong to class $\mathcal{KL}$, if for each fixed s, the mapping $\beta(r, s)$ belongs to class $\mathcal{K}$ with respect to r and, for each fixed r, the mapping $\beta(r, s)$ is decreasing with respect to s and $\beta(r, s) \to 0$ as $s \to \infty$. The system $\dot{x} = f(t, x, u)$ where $f : [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is piecewise continuous in t and locally Lipschitz in x and u, is said to be Input-to-State Stable (ISS) if there exist a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ function γ such that for any initial time t_0, initial state $x(t_0)$ and any bounded input $u(t)$, the solution $x(t)$ exists for all $t \geq t_0$ and satisfies

$$\|x(t)\| \leq \beta(\|x(t_0)\|, t - t_0) + \gamma\left(\sup_{t_0 \leq \tau \leq t} \|u(\tau)\|\right) \tag{1}$$

Additionally, it admits a smooth ISS-Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$ such that $\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|)$ and $\dot{V} \leq -\alpha_3(\|x\|) + \alpha_4(\|u\|)$ holds for any x, where $\alpha_i \in \mathcal{K}_\infty$ for $(i = 1, \cdots, 4)$.

The rest of the paper is organized as follows, Section II formulates the research problem addressed in this paper and briefly describes the subsystems of an Event-based CPF framework. Section III solves the Path Following problem. The event-based cooperative control method is presented in Section IV. In particular, the main conceptual result of the paper i.e., the event-based CPF is provided. The experimental results are discussed in Section V followed by conclusions in Section VI.

II. Problem Formulation

The Event-based CPF strategy illustrated in Figure 2 for a generic vehicle i, consists of a interconnection of two control subsystems namely, the *Event-based Cooperative Controller*, tasked to achieve cooperation between the robotic vehicles in an event-based manner, and the *Path Following* controller that is responsible to guide the robots through a desired geometric path with a desired speed [10], [13].

To this end, consider N robotic vehicles described by the following kinematic model,

$$\begin{aligned} \dot{\mathbf{p}}_i(t) &= R_i(t)\mathbf{v}_i(t) + \mathbf{w}_i(t) \\ \dot{R}_i(t) &= R_i(t)S(\boldsymbol{\omega}_i) \end{aligned} \tag{2}$$

where $(\mathbf{p}_i(t), R_i(t)) \in SE(n)$ denotes the position and orientation of the robotic vehicle, where $n = 2$ for 2D and $n = 3$ for 3D space, $(\mathbf{v}_i(t), S(\boldsymbol{\omega}_i(t))) \in se(n)$ denotes the linear and angular input velocities, expressed in the Body frame $\{B\}$, and $\mathbf{w}_i(t) \in \mathbb{R}^n$ denotes disturbances acting on the vehicle expressed in the inertial frame. The orientation of the robot is expressed by the Rotation matrix $R_i(t)$ from the body frame to the inertial frame. $S(\boldsymbol{\omega}_i(t))$ is a skew symmetric matrix associated with the input angular velocity $\boldsymbol{\omega}_i$. For simplicity, we will consider the robotic vehicles at kinematic level and assume that each vehicle includes an implemented inner-loop controller to address the actuator dynamics, leveraging the focus to the outer-loop controller that generates linear and angular velocity references. For under-actuated vehicles and $n = 2$, which is the case of the ASVs considered in this paper, the linear input velocity vector reduces to $\mathbf{v}_i(t) = [v_{f_i}, 0]^T$. The control inputs for the vehicle are $\mathbf{u}_i(t) = [v_{f_i}, \boldsymbol{\omega}_i]^T$.

The Path Following problem can be stated as follows.

Problem 1 (Path Following). Given a reference geometric path $\mathbf{p}_{d_i}(\gamma_i) : \mathbb{R} \to \mathbb{R}^n$ parameterized by the path variable $\gamma_i \in \mathbb{R}$ and a desired common speed for the path variable $v_d \in \mathbb{R}$, for each vehicle $i \in \mathcal{I}$, the path following problem is to design a feedback control law $\mathbf{u}_i(t)$ such that the path following error, $\|\mathbf{p}_i(t) - \mathbf{p}_{d_i}(\gamma_i)\|$ converges to an arbitrary small neighborhood of the origin as $t \to \infty$. Furthermore, the robotic vehicle must satisfy the desired speed assignment, $\|\dot{\gamma}_i - v_d\| \to 0$ as $t \to \infty$. □

The path variable γ_i can be viewed as a virtual point moving along the path $\mathbf{p}_{d_i}$ and the objective of the path following controller is to drive the robotic vehicle around an arbitrary small neighborhood of the virtual point. In order to achieve the coordination objective, the following condition is imposed for the dynamics of the path variable γ_i,

$$\dot{\gamma}_i = v_d + \tilde{v}_r^i(t) + g_i(t) \tag{3}$$

where $\tilde{v}_r^i(t)$ is the cooperative control actuation signal obtained from the cooperative control system and the real function $g_i(t)$ referred to as the path following error correction term, is a known, continuous, and uniformly bounded signal such that $\|g_i(t)\| \leq \mu$. This signal can be seen as an exogenous input to the dynamics of the path variable whose role is to assist the robotic vehicle in the path following. This is achieved through actuating in the evolution of the virtual point γ_i, according to the path following error $\|\mathbf{p}_i(t) - \mathbf{p}_{d_i}(\gamma_i)\|$, i.e, if the robotic vehicle lags or leads the virtual point. This allows the path following system to deal with the situations where the robotic vehicle is not able to catch up to the virtual point due to disturbances, brief actuator faults, etc. Clearly, one would expect that the signal $g_i(t)$ vanishes when the path following error goes to zero (or around the neighborhood of zero).

Consequently, the cooperative control system is required to achieve i) synchronization (in a practical sense) of the path variables γ_i for all $i \in \mathcal{I}$, and ii) respond to the disturbances on any of the robots by modifying the speed of the path variables (and hence the individual vehicles) such that they remain synchronized. Assuming that the path following controller achieves the desired objectives on each robotic vehicle, the Event-based Cooperative Control problem is stated as follows.

Problem 2 (Event-based Cooperative Control). Given the path variables γ_i, $i \in \mathcal{I}$, for the N robotic vehicles, and an imposed communication topology between the vehicles, the objective of the cooperative control system is to design a decentralized cooperative control actuation signal $\tilde{v}_r^i$ together with a decentralized Event-Triggering Condition (ETC) such that: 1) the position of the virtual points (denoted by γ_i) along the reference path is synchronized in a practical sense, that is, $\|\gamma_i - \gamma_j\|$ converges to a neighborhood of zero for all $i, j \in \mathcal{I}$ as $t \to \infty$; and 2) there is no data transmissions between the robots while the ETC is valid. Once it breaks, an event is generated by the system, leading to information transmission and an update of the cooperative control actuation signal. □

III. Path Following Control

A. Controller design

We consider the controller presented in [2], but at the kinematic level, see also [14] for this case. Let $\mathbf{e}_i = R_i^T(\mathbf{p}_i - \mathbf{p}_{d_i}(\gamma_i)) + \boldsymbol{\epsilon}$, be an error variable where $\boldsymbol{\epsilon}$ is a given small vector. The error dynamics of the path following system is given by

$$\begin{aligned} \dot{\mathbf{e}}_i &= \dot{R}_i^T(\mathbf{p}_i - \mathbf{p}_{d_i}(\gamma_i)) + R_i^T(\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_{d_i}(\gamma_i)) \\ &= -S(\boldsymbol{\omega}_i)\mathbf{e}_i + \Delta\mathbf{u}_i - R_i^T\frac{\partial \mathbf{p}_{d_i}(\gamma_i)}{\partial \gamma_i}\dot{\gamma}_i + R_i^T\mathbf{w}_i \end{aligned} \tag{4}$$

where

$$\Delta = \begin{bmatrix} 1 & -\epsilon_2 \\ 0 & \epsilon_1 \end{bmatrix} \quad \text{or} \quad \Delta = \begin{bmatrix} 1 & 0 & \epsilon_3 & -\epsilon_2 \\ 0 & -\epsilon_3 & 0 & \epsilon_1 \\ 0 & \epsilon_2 & -\epsilon_1 & 0 \end{bmatrix}$$

for the case of horizontal plane (2D), or for the general case (3D), respectively. We consider a realistic situation where $\mathbf{e}_i(t)$ is not precisely known, instead only an estimate of $\mathbf{e}_i(t)$, denoted as $\hat{\mathbf{e}}_i(t)$ is known. Let $\tilde{\mathbf{e}}_i = \hat{\mathbf{e}}_i - \mathbf{e}_i$ be the estimation error, and consider that $\boldsymbol{\epsilon}$ is selected such that Δ is full rank and the term $\|\frac{\partial \mathbf{p}_{d_i}}{\partial \gamma_i}\|$ is bounded. Then, the following result holds.

Proposition 1 (Path Following)**.** *Given the error dynamics for the path following system described by* (4)*, the control law*

$$\mathbf{u}_i = \Delta^+ \left(-K_{p_i}\hat{\mathbf{e}}_i + R_i^T \frac{\partial \mathbf{p}_{d_i}(\gamma_i)}{\partial \gamma_i} v_d \right) \quad (5)$$

where Δ^+ is the Moore-Penrose pseudo inverse, K_{p_i} is a known positive definite gain matrix, makes the origin $\mathbf{e}_i = 0$ of the closed-loop system Input-to-State Stable (ISS) with respect to the estimation error $\tilde{\mathbf{e}}_i(t)$, the cooperative control actuation signal $\tilde{v}_r^i(t)$, the path following error correction signal $g_i(t)$ and the unknown bounded disturbance $\mathbf{w}_i(t)$.

Proof. The result can be concluded by following the same arguments in [11] with the Lyapunov function $V_{\mathrm{PF}_i}(\mathbf{e}_i) = \frac{1}{2}\mathbf{e}_i^T\mathbf{e}_i$. □

Remark 1. Note that Proposition 1 implies convergence of the error $\mathbf{e}_i(t)$ to a small neighborhood of zero, whose size depends on the bound of the exogenous input and disturbance signals $\left(\tilde{\mathbf{e}}_i(t), \tilde{v}_r^i(t), g_i(t), \mathbf{w}_i(t)\right)$. The error converges to zero if these disturbances vanishes to zero.

Remark 2. If $\boldsymbol{\epsilon} = 0$, we do not have direct control over the attitude of the robotic vehicle i.e., the control input $\boldsymbol{\omega}_i$ does not appear in the error dynamics (4). Hence, it is necessary to have the condition $\boldsymbol{\epsilon} \neq 0$ satisfied for this controller.

B. Path following error correction signal $g_i(t)$

As explained earlier, the objective of the function $g_i(t)$ is to assist the robotic vehicle in the path following. To this end, we consider the variation in the squared norm of the path following error $\mathbf{e}_i$ with respect to γ_i. Note that $\mathbf{e}_i$ is a funcion of γ_i, hence

$$\eta_i = \frac{\partial (1/2)\mathbf{e}_i^T\mathbf{e}_i}{\partial \gamma_i} = -\mathbf{e}_i^T \left(R_i^T \frac{\partial \mathbf{p}_{d_i}(\gamma_i)}{\partial \gamma_i} \right) \quad (6)$$

Then selecting $g_i(t) = -k_\eta \mathrm{sat}(\eta_i)$ for any $k_\eta > 0$ serves to modify the evolution of the path variable according to the path following error $\mathbf{e}_i(t)$. In the implementation, the vector $\left(R_i^T \frac{\partial \mathbf{p}_{d_i}(\gamma_i)}{\partial \gamma_i} \right)$ is normalized and we ensure that the bound $\|g_i(t)\| \leq \mu$ is selected such that $\mu \geq v_d$. This provides flexibility to the virtual point to move freely along the path.

IV. Event-based Cooperative Path Following

A. Event-based cooperative control

Consider the dynamics of the path variable of N robotic vehicles that satisfies (3), rewritten as $\dot{\gamma}_i(t) = \nu_i(t) \quad \forall\ i \in \mathcal{I}$, where $\nu_i(t) = v_d + \tilde{v}_r^i(t) + g_i(t)$. Let t_k^i for all $k \in \mathbb{Z}_{\geq 0}$ denote the time instants, also referred to as event time, at which agent i updates its cooperative control actuation signal $\tilde{v}_r^i(t) = \tilde{v}_r^i(t_k^i)$ for all $t \in [t_k^i, t_{k+1}^i)$ and transmits the necessary information over the network to its neighbors. Furthermore, let $\hat{\gamma}_j^i$ denote the estimate of the path variable of agent j computed on agent i for all $j \in \mathcal{N}_i$. The estimation error of the path variable j from the perspective of agent i is then denoted as $\tilde{\gamma}_j^i := \gamma_j - \hat{\gamma}_j^i$. Then, the event-based cooperative control system on every agent i, given all the necessary information from its neighbors, performs the following tasks: i) estimate the path variable of its neighbor $\hat{\gamma}_j^i$ for all $j \in \mathcal{N}_i$, and ii) use the estimates $\hat{\gamma}_j^i$ to determine the event time t_k^i, at which the cooperative control input $\tilde{v}_r^i(t_k^i)$ is computed and the information is transmitted such that the convergence and stability property of the overall CPF system is preserved. In order to achieve the desired objectives of the cooperative controller, we make the following standing assumption on the system.

Assumption 1. The i^{th} robot communicates only with its fixed neighbors $j \in \mathcal{N}_i$. Furthermore, it is able to transmit successfully at its event time t_k^i, and vice versa. Such an assumption of fixed communication topology is viable for multi-robot systems executing cooperative maneuvers wherein the geometric shape of the formation is static.

It is well known that the distributed consensus law, $\tilde{v}_r^i(t) = -\sum_{j\in\mathcal{N}_i}(\gamma_i(t) - \gamma_j(t))$ has been at the core of multi-agent coordination applications. The implementation of such a control law requires constant measurement of the states of the agents neighbor. Certainly, such a strategy in many cases is practically not feasible and as we show in this paper, is not necessary. In order to meet the objective of reducing the frequency of communication among the agents, we consider the piecewise constant, event-based control law [8]

$$\tilde{v}_r^i(t) = -k_i \sum_{j\in\mathcal{N}_i} \left(\hat{\gamma}_i^i(t_k^i) - \hat{\gamma}_j^i(t_k^i) \right) \quad \forall t \in [t_k^i, t_{k+1}^i) \quad (7)$$

where $k_i > 0$ is the consensus gain. Note that in this case the estimates of the path variables computed locally by the agent i, are used to compute (7) instead of the actual values γ_i. Define the path variable measurement error,

$$\xi_i(t) := \sum_{j\in\mathcal{N}_i} \left(\hat{\gamma}_i^i(t_k^i) - \hat{\gamma}_j^i(t_k^i) \right) - \sum_{j\in\mathcal{N}_i} \left(\hat{\gamma}_i^i(t) - \hat{\gamma}_j^i(t) \right) \quad (8)$$

and note that, by definition $\xi_i(t_k^i) = 0$. Then, the cooperative control actuation signal can be written as $\tilde{v}_r^i(t) = -k_i \left(\sum_{j\in\mathcal{N}_i} \left(\hat{\gamma}_i^i(t) - \hat{\gamma}_j^i(t) \right) + \xi_i(t) \right)$. Using $\hat{\gamma}_i^i(t) = \gamma_i(t)$ and $\hat{\gamma}_j^i(t) = \gamma_j(t) - \tilde{\gamma}_j^i(t)$ we have, $\tilde{v}_r^i(t) = -k_i \left(\sum_{j\in\mathcal{N}_i} (\gamma_i(t) - \gamma_j(t)) + \xi_i(t) + \sum_{j\in\mathcal{N}_i} \tilde{\gamma}_j^i(t) \right)$. Let $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \cdots, \gamma_N]^T$ and $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_N]^T$ denote the path variable and the path variable measurement error stacked together as a column vector. Then, the dynamics of the path variable can be collectively written as

$$\dot{\boldsymbol{\gamma}} = v_d \mathbf{1}_N - K_c L \boldsymbol{\gamma} - K_c \breve{\boldsymbol{\gamma}} - K_c \boldsymbol{\xi} + \mathbf{g} \quad (9)$$

where $K_c = \mathrm{diag}(k_{c_1}, \cdots, k_{c_N})$, $\breve{\boldsymbol{\gamma}} \in \mathbb{R}^N$ such that $[\breve{\gamma}]_i = \sum_{j\in\mathcal{N}_i} \tilde{\gamma}_j^i(t)$, and $\mathbf{g} = [g_1, \cdots, g_N]^T$. In order to prove ISS of the event-based consensus controller (7), introduce the disagreement vector [3] as, $\boldsymbol{\delta} := \boldsymbol{\gamma} - \alpha \mathbf{1}_N$, where $\alpha = (1/N)\mathbf{1}_N^T\boldsymbol{\gamma}$ is the average of path variables of all the robotic vehicles, $L\boldsymbol{\gamma} = L\boldsymbol{\delta}$ and $\mathbf{1}_N^T\boldsymbol{\delta} = 0$. We now state the following result on event-based cooperative control, wherein the Event Triggering Condition (ETC) is designed and ISS property is proven for the proposed ETC.

Theorem 1 (Event-based Cooperative Control)**.** *Given the dynamics of the path variable* (3)*, let each robotic vehicle i* *transmit the information packet*

$$\mathcal{C}_i(t_k^i) := \left(t_k^i, \gamma_i(t_k^i), \nu_i(t_k^i)\right) \tag{10}$$

at its event time t_k^i to the neighboring agents $j \in \mathcal{N}_i$. Then, the event based cooperative control law (7)*, under the Assumption 1, makes the system ISS with respect to the path variable measurement error $\xi_i(t)$, path variable estimation error $\tilde{\gamma}_i(t)$ and the path following error correction term $g_i(t)$ provided the Event Triggering Condition (ETC)*

$$\xi_i^2 \leq \frac{\sigma_i}{8}\left(\sum_{j\in\mathcal{N}_i}\left(\hat{\gamma}_i^i(t) - \hat{\gamma}_j^i(t)\right)\right)^2 \tag{11}$$

where

$$\begin{aligned}\hat{\gamma}_i^i(t) &= \gamma_i(t)\\ \hat{\gamma}_j^i(t) &= \gamma_j(t_{k_j(t)}^j) + (t - t_{k_j(t)}^j)\nu_j(t_{k_j(t)}^j)\end{aligned} \tag{12}$$

is satisfied for all $0 < \sigma_i < 1$. In (12)*, $t_{k_j(t)}^j$ denotes the most recent event time on agent $j \in \mathcal{N}_i$. The next event time t_{k+1}^i is defined as*

$$t_{k+1}^i = t_k^i + \min\{\tau_k^i, \tau_{\mathrm{ub}}\} \tag{13}$$

with $\tau_k^i = \min\left\{t - t_k^i > 0 : \xi_i^2 \geq \frac{\sigma_i}{8}\left[\sum_{j\in\mathcal{N}_i}\left(\hat{\gamma}_i^i - \hat{\gamma}_j^i\right)\right]^2\right\}$ *and $\tau_{\mathrm{ub}} > 0$ is an upper bound to the inter-event time specified as a design parameter.*

Proof. We show the result through the following steps

Step 1: ISS Lyapunov function – Consider the ISS Lyapunov function

$$V_{\mathrm{CC}}(\boldsymbol{\delta}) = \frac{1}{2}\boldsymbol{\delta}^T L\boldsymbol{\delta}$$

Taking the time derivative and using (9) yields

$$\dot{V}_{\mathrm{CC}}(\boldsymbol{\delta}) = -\boldsymbol{\delta}^T LK_c L\boldsymbol{\delta} - \boldsymbol{\delta}^T LK_c\tilde{\boldsymbol{\gamma}} - \boldsymbol{\delta}^T LK_c\boldsymbol{\xi} + \boldsymbol{\delta}^T L\mathbf{g}$$

Let $\mathbf{z} = L\boldsymbol{\delta}$ and $z_i = [L\boldsymbol{\delta}]_i$, then the time derivative of the ISS Lyapunov function can be written as

$$\dot{V}_{\mathrm{CC}} = -\sum_{i=1}^N k_i z_i^2 - \sum_{i=1}^N\sum_{j\in\mathcal{N}_i} k_i z_i\tilde{\gamma}_j^i - \sum_{i=1}^N k_i z_i\xi_i + \sum_{i=1}^N z_i g_i$$

Note that the path variable estimation error $\tilde{\gamma}_i^j = \gamma_i - \hat{\gamma}_i^j$ is identical to all the agents $j \in \mathcal{N}_i$. This holds as all agents $j \in \mathcal{N}_i$ use the same information $\mathcal{C}_i(t_k^i)$ obtained from agent i at its event time t_k^i to estimate $\hat{\gamma}_i^j$ using (12). Using the fact that the communication topology is undirected, the term $\sum_{j\in\mathcal{N}_i}\tilde{\gamma}_j^i$ can be written as $\sum_{j\in\mathcal{N}_i}\tilde{\gamma}_i^j \leq |\mathcal{N}_i|\tilde{\gamma}_i$. Therefore,

$$\dot{V}_{\mathrm{CC}} \leq -\sum_{i=1}^N k_i z_i^2 - \sum_{i=1}^N |\mathcal{N}_i| k_i z_i\tilde{\gamma}_i - \sum_{i=1}^N k_i z_i\xi_i + \sum_{i=1}^N z_i g_i$$

Using young's inequality $|xy| \leq (1/2a_i)x^2 + (a_i/2)y^2$ for the terms $|\xi_i z_i|$, $|z_i\tilde{\gamma}_i|$ and $|z_i g_i|$ with coefficients a_i equal to $1/4$, $4|\mathcal{N}_i|$, $2/k_i$ respectively

$$\dot{V}_{\mathrm{CC}} \leq -\sum_{i=1}^N \frac{k_i}{2}z_i^2 + \sum_{i=1}^N 2k_i\xi_i^2 + \sum_{i=1}^N 2|\mathcal{N}_i|^2 k_i\tilde{\gamma}_i^2 + \sum_{i=1}^N \frac{1}{k_i}g_i^2 \tag{14}$$

Algorithm 1 Event-based Control and Communication for Agent i

Input: Information received: $\mathcal{C}_j(t_{k_j(t)}^j)\ \forall\ j \in \mathcal{N}_i$ and γ_i.
Output: Event-based control input: $\nu_i(t)$ and communication packet $\mathcal{C}_i(t_k^i)$.

```
Initialization:
Define k ← 0; t_k^i ← t; ṽ_r^i(t_k^i) ← 0; γ_i(t_k^i) ← γ_i.
Transmit C_i(t_k^i)
for t = hτ_s where h ∈ Z_{≥0} do
    Estimate γ̂_j^i(t) for all j ∈ N_i using (12).
    Compute ETC using (11).
    if ξ_i^2 > (σ_i/8)[Lγ̂^i]_i^2 and |ξ_i^2 − (σ_i/8)[Lγ̂^i]_i^2| > ϱ then
        k ← k + 1; t_k^i ← t; γ_i(t_k^i) = γ_i(t); g_i(t_k^i) = g_i(t)
        Compute ṽ_r^i(t_k^i) using (7)
        Reset ξ_i(t_k^i) to zero
        Transmit C_i(t_k^i)
    end if
    Output ν_i(t) ← v_d + ṽ_r^i(t_k^i) + g_i(t)
end for
```

Clearly, we have ISS of the disagreement vector $\boldsymbol{\delta}(t)$ provided that the signals $\xi_i(t)$, $\tilde{\gamma}_i(t)$ and $g_i(t)$ are bounded for all $i \in \mathcal{I}$. This indeed is true as discussed next.

Step 2: Boundedness of $\xi_i(t)$ – The boundedness of the path variable measurement signal $\xi_i(t)$ can be ensured using the ETC (11). The ETC (11) satisfies $\xi_i^2 \leq \frac{\sigma_i}{4}z_i^2 + \frac{\sigma_i}{4}\left(\sum_{j\in\mathcal{N}_i}\tilde{\gamma}_j^i\right)^2$, where we have used the relation $(a+b)^2 \leq 2a^2 + 2b^2$. Substituting the above equation in the time derivative of the ISS Lyapunov function (14) with $\sum_{j\in\mathcal{N}_i}\tilde{\gamma}_j^i \leq |\mathcal{N}_i|\tilde{\gamma}_i$, $(\gamma_i - \gamma_j) = (\delta_i - \delta_j)$, and $\lambda_2(L)\boldsymbol{\delta}^T L\boldsymbol{\delta} \leq \boldsymbol{\delta}^T LL\boldsymbol{\delta} \leq \lambda_N(L)\boldsymbol{\delta}^T L\boldsymbol{\delta}$, we obtain

$$\dot{V}_{\mathrm{CC}} \leq -\alpha_{\mathrm{CC}}V_{\mathrm{CC}} + \rho_1\|\tilde{\boldsymbol{\gamma}}\|^2 + \rho_2\|\mathbf{g}\|^2 \tag{15}$$

where $\alpha_{\mathrm{CC}} = \frac{k}{2}(1-\sigma)\lambda_2(L)$ with $k_i = k$ and $\sigma_i = \sigma$, $\rho_1 = |\mathcal{N}_{\max}|^2 k\left(2 + \frac{\sigma}{2}\right)$ where $\mathcal{N}_{\max} = \max_{i\in\mathcal{I}}|\mathcal{N}_i|$ and $\rho_2 = 1/k$.

Step 3: Boundedness of $\tilde{\gamma}_i(t)$ and $g_i(t)$ – The signal $g_i(t)$ is uniformly bounded by design. Hence, boundedness of $\tilde{\gamma}_i(t)$ remains to be proven. The time derivative of the path variable estimation error satisfies $\dot{\tilde{\gamma}}_i^j = \nu_i(t) - \nu_i(t_k^i) = g_i(t) - g_i(t_k^i)$. Using $\|g_i(t) - g_i(t_k^i)\| \leq 2\mu$, the path variable estimation error signal is always bounded as,

$$\|\tilde{\gamma}_i\| \leq (t - t_k^i)2\mu \quad \forall t \in [t_k^i, t_{k+1}^i) \tag{16}$$

Note that the superscript j is dropped to indicate that $\tilde{\gamma}_i^j = \tilde{\gamma}_i$ for all $j \in \mathcal{N}_i$ and $\tilde{\gamma}_i$ is zero at event times of agent i. The signal $\tilde{\gamma}_i$ is bounded if the inter-event time $\tau_k^i = t_{k+1}^i - t_k^i$ is upper bounded. This is guaranteed by choosing the next event time t_{k+1}^i using (13). Consequently, the event based cooperative control system is ISS with respect to the path variable estimation error $\tilde{\gamma}_i(t)$ and the path following error correction term $g_i(t)$. □

The algorithm for event-based control and communication strategy adopted is given in Algorithm 1. In order to implement the event-based consensus controller, the following modification is made to the ETC, $\xi_i^2 > (\sigma_i/8)[L\hat{\boldsymbol{\gamma}}^i]_i^2$ and $|\xi_i^2 - (\sigma_i/8)[L\hat{\boldsymbol{\gamma}}^i]_i^2| > \varrho$, with a small parameter ϱ (in our case $\varrho = 10^{-3}$). The second condition allows some slack for the ETC in order to prevent event-generation due to the numerical issues when the consensus has been achieved. The approach presented in this paper differs from the previous work [11] in two aspects. First, it is reactive to disturbances acting on the vehicle due to introduction of the signal $g_i(t)$ and repeated monitoring of the ETC. Second, an agent communicates the necessary data in a single packet at its event time when compared to the self-triggered approach that needed $(|\mathcal{N}_i| + 2)$ packets of data to be exchanged.

B. Event-based CPF

The main conceptual result of this paper can be summarized as follows.

Theorem 2 (Event-based CPF)**.** *The decentralized event-based control and communication method given by* (7) *and the ETC* (11)*, along with the path following controller* (5)*, collectively termed as Event-based CPF controller makes the system* (2) *and* (3) *ISS with respect to the path following estimation error* $\tilde{\mathbf{e}}_i(t)$*, unknown disturbance acting on the robotic vehicle* $\mathbf{w}_i(t)$*, the path following error correction term* $g_i(t)$ *and the path variable estimation error* $\tilde{\gamma}_i(t)$*.*

Proof. The proof directly follows from the results of Proposition 1 and Theorem 1 that are two ISS subsystems of Event-based CPF. The result is a direct consequence of the fact that the cascade connection of two ISS systems result in an ISS system [12], [15]. □

V. Experimental Results

The Event-based CPF method was experimentally validated using three AUVs (see Figure 1), operating at the surface and communicating with one another over Wi-Fi using UDP. The Event-based CPF was implemented in C++ using the DUNE toolchain that is a runtime environment for unmanned systems on-board software. The vehicles were constantly monitored using Neptus Command and Control framework[1]. The event-based CPF method was tested for two test cases: straight line formation and circular formation. The path variable in straight line formation corresponds to the along path distance, whereas they denote the arc for a circular formation. Although the experiments were conducted for 2D case, the results are applicable to 3D tests. The event-based control and communication algorithm was executed with a sampling frequency of 100 Hz using $\sigma_i = 0.8$ for all $i \in \mathcal{I}$. The event-based consensus controller gain was chosen as $k_i = 0.5$. The values of $\boldsymbol{\epsilon}$ in the path following controller was chosen as $[0.3,\ 0]^T$ for all the vehicles. A constant desired speed assignment of $v_d = 1$ m/s and $v_d = 0.035$ rad/s was provided respectively, for the linear and circular formations.

[1]The toolchains DUNE/NEPTUS and the AUVs were developed at the Laboratòrio de Sistemas e Tecnologia Subaqutica (LSTS) in University of Porto. See https://github.com/LSTS/dune and https://github.com/LSTS/neptus

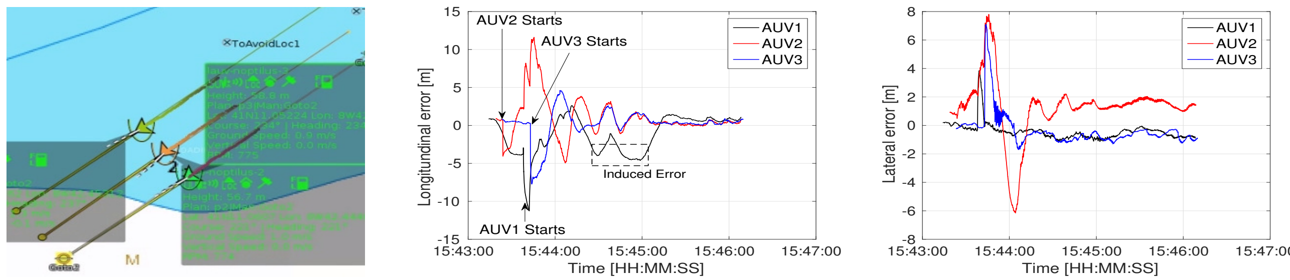
TABLE I: Event time for Straight line formation

	AUV-1	AUV-2	AUV-3
Duration [s]	216.49	237.48	211.49
Max τ_k [s]	12.28	20.93	11.17
Min τ_k [s]	0.01	0.01	0.01
Num Events	474	378	428
Periodic	21649	23748	21149
% Comms	2.189	1.592	2.023

TABLE II: Event time for Circular formation

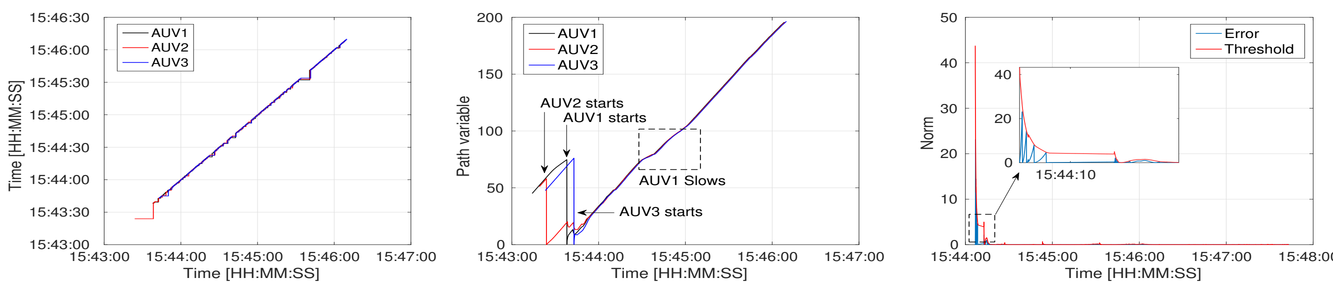
	AUV-1	AUV-2	AUV-3
Duration [s]	617.96	643.48	648.17
Max τ_k [s]	160.24	32.10	78.80
Min τ_k [s]	0.70	0.03	0.61
Num Events	31	36	51
Periodic	61796	64348	64817
% Comms	0.050	0.055	0.078

In order to demonstrate the utility of the function $g_i(t)$, AUV-1 was actuated in open loop with a reduced speed of 0.6 [m/s] for a brief duration. The function $g_i(t)$ was selected as $-k_{\eta_i}\left(\tanh(\eta_i + 5) + \tanh(\eta_i - 5)\right))$ with $k_{\eta_i} = 2v_d$. This particular choice of a stair-case like shaped signal with a dead-zone is motivated by the fact that the actuation in path variable is not required when the path following errors are close to zero. Figure 3a shows the snapshot of the Neptus console while the vehicles execute the CPF in a straight line. The associated longitudinal and lateral errors in the vehicle positions are shown in Figure 3b and 3c respectively. The longitudinal and the lateral errors in absence of disturbances should converge to zero. However, due to the external disturbances such as currents, and un-modeled dynamics of the propellers and the attitude controller of the AUV, the error converges to a region around zero. The size of the region is dictated by the magnitude of the disturbances acting on the vehicle. Additionally, the artificially induced error in AUV-1 leads to a path following error as highlighted in Figure 3b. The main objective of the proposed event-based control and communication method is to reduce the frequency of communication between the robotic vehicles. Figures 4a and 5d show the generated event times for the AUVs in a linear and circular formation respectively. As it can be seen from the figure, there are time intervals during which no events are generated and hence no communication takes place. Table I and II shows the quantitative data for the event time in the case of straight line CPF and circular CPF respectively. In order to appreciate the results, we provide the explanation of the data for the AUV-3 in a straight line CPF. From Table I, it can be seen that the total mission duration was 211.49 seconds. In a periodic transmission with sampling time of 0.01 [s], a total of 21149 packets of data would be sent out by the AUV. With the event-based strategy, the AUV communicates 428 times in the entire mission. Considering periodic transmission as 100% communication, the AUV-3 communicates only 2.02% of the mission duration. The maximum and the minimum inter-event time for the AUV-3 is 11.17 and 0.01 seconds respectively. Similar results can be noted for other AUVs for the case of both straight line and circular formation. Clearly, with the proposed method,

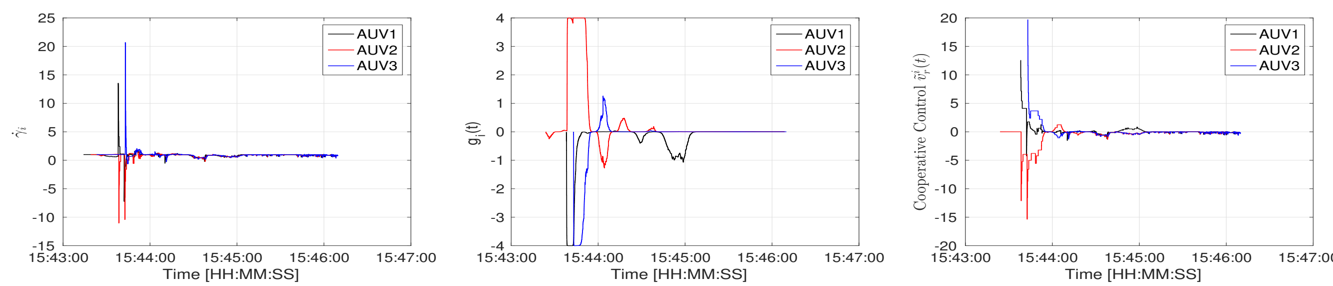


(a) Snapshot of Neptus console (b) Longitudinal error in vehicle position (c) Lateral error in vehicle position

Fig. 3: Vehicle positions and path following error for the AUVs in straight line formation



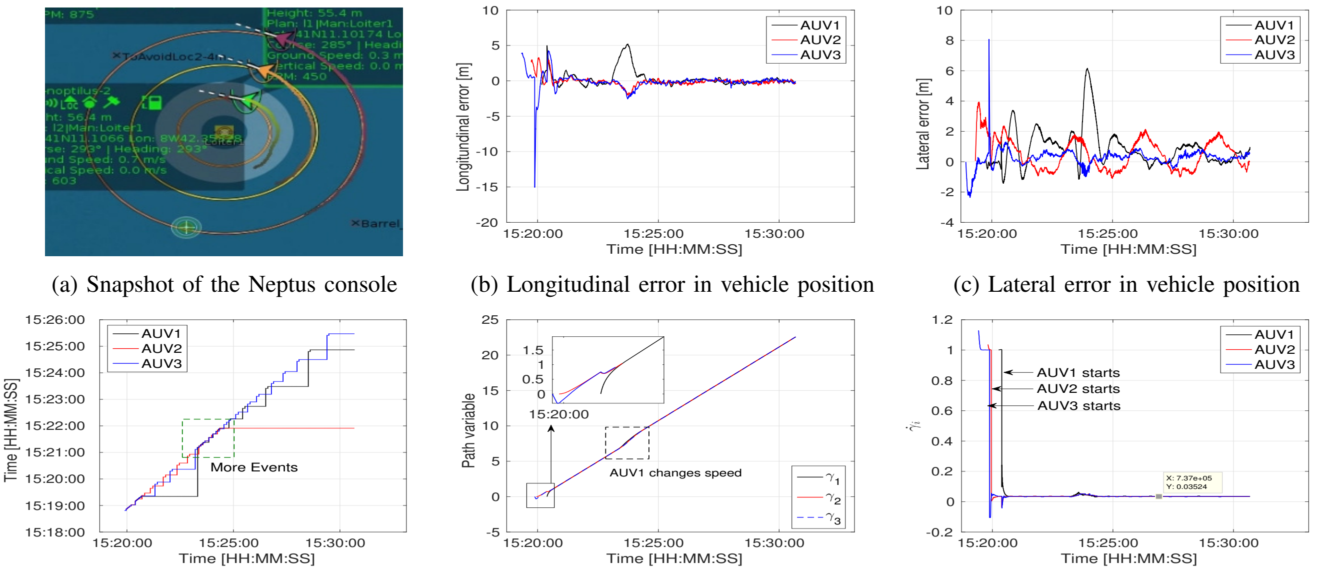
(a) Event-times (b) Path variable γ_i for all $i \in \mathcal{I}$ (c) Norm of ETC for AUV-1



(d) Time derivative of the path variable (e) $g_i(t)$ of AUV's (f) Cooperative control signal $\tilde{v}_r^i$

Fig. 4: Results of Event-based CPF for three AUVs in straight line formation

we are able to reduce the frequency of communication by a significant amount. Note that these results assume that all the transmissions by the AUVs have been successful. This need not be true in practice, where it is possible that the transmitted data packets might not be received by the robots neighbor. Notice that in the experiments, the robots do not start cooperating with one another instantaneously or at the same time instant. The event-based CPF is initiated only when the robots reach the starting way point. Therefore, the event-based consensus starts executing on every robot at different time instants (indicated in Figure 3b and 4b). Figure 4b shows the evolution of the path variables and also indicates the time instants at which specific vehicles starts cooperating. Clearly, consensus is achieved and the path variable evolves according to the desired speed assignment as seen in Figure 4d. Note that the time instant at which the AUV-1 slows down results in correction action due to the term $g_i(t)$ as shown in Figure 4e. The other vehicles in the network respond to the change in the speed of the AUV-1 through the use of the cooperative control actuation signal (Figure 4f) while maintaining consensus. Note that the actuation of the path variables due to $g_i(t)$ is zero for AUV-2 and AUV-3 when AUV-1 reduces its speed. This reinforces the fact that $g_i(t)$ is responsible to assist individual robots in path following, while the coordination is achieved through cooperative control actuation signal $\tilde{v}_r^i(t)$. The coordination between the vehicles in presence of disturbances on any of the vehicles comes at an increased communication cost as can be noted in Figure 4a and 5d. This is due to the fact that the path variables are actuated by different $g_i(t)$ for all $i \in \mathcal{I}$ and frequent communication is required to maintain consensus (coordination). Figures 4c shows the plot of norm of the measurement error variable $\xi_i(t)$ associated with the event-based consensus method (in blue) versus the norm of ETC when it holds with equality (in red) for AUV-1. An event is generated when the norm of the error variable hits the threshold, thereby triggering the control update and information transmission. The results of the event-based CPF for circular formation are shown in Figure 5a - 5f. The norm of the measurement error variables and the ETC for circular



(a) Snapshot of the Neptus console (b) Longitudinal error in vehicle position (c) Lateral error in vehicle position

(d) Event time (e) Path variables γ_i for all $i \in \mathcal{I}$ (f) Time derivative of path variables

Fig. 5: Results of Event-based CPF for three AUVs in circular formation.

formation is not provided due to the limitations of space.

VI. Conclusion

In this paper, a novel approach to the CPF problem was presented using event-based control and communication strategy. A decentralized event-based control and communication scheme was proposed which aims to reduce the frequency of communication between the robotic vehicles through a suitable defined ETC and communication of computed cooperative control input in addition to the path variables. Stability and convergence guarantees were provided using ISS framework and the approach was experimentally validated using three AUVs in the presence of external disturbances[2]. Furthermore, it was shown that the system is responsive to the disturbances on any of the vehicles and coordinate with one another to slow down (or speed up) while achieving significant reduction of inter-vehicle communication when compared with periodic transmission. Future work would involve the formal investigation of practical issues such as communication losses and delays. This would enable us to test the vehicles underwater, using the acoustic communication channels.

References

[1] R. Ghabcheloo, A. P. Aguiar, A. Pascoal, C. Silvestre, I. Kaminer, and J. Hespanha, "Coordinated path-following in the presence of communication losses and time delays," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 234–265, 2009.

[2] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1362–1379, Aug 2007.

[3] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan 2007.

[4] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.

[5] A. Anta and P. Tabuada, "To sample or not to sample: Self-triggered control for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2030–2042, 2010.

[6] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2012.

[7] G. S. Seyboth, D. V. Dimarogonas, and K. H. Johansson, "Event-based broadcasting for multi-agent average consensus," *Automatica*, vol. 49, no. 1, pp. 245–252, 2013.

[8] Y. Fan, L. Liu, G. Feng, and Y. Wang, "Self-triggered consensus for multi-agent systems with zeno-free triggers," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, pp. 2779–2784, 2015.

[9] C. Nowzari and J. Cortés, "Distributed event-triggered coordination for average consensus on weight-balanced digraphs," *Automatica*, vol. 68, pp. 237–244, 2016.

[10] A. P. Aguiar and A. M. Pascoal, "Coordinated path-following control for nonlinear systems with logic-based communication," in *2007 46th IEEE Conference on Decision and Control*, 2007, pp. 1473–1479.

[11] R. P. Jain, A. P. Aguiar, and J. Sousa, "Self-triggered cooperative path following control of fixed wing unmanned aerial vehicles," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, June 2017, pp. 1231–1240.

[12] E. D. Sontag and Y. Wang, "On characterizations of the input-to-state stability property," *Systems and Control Letters*, vol. 24, no. 5, pp. 351 – 359, 1995.

[13] R. Ghabcheloo, A. Pascoal, C. Silvestre, and I. Kaminer, "Nonlinear co-ordinated path following control of multiple wheeled robots with bidirectional communication constraints," *International Journal of Adaptive control and signal Processing*, vol. 21, no. 2-3, pp. 133–157, 2007.

[14] A. Alessandretti, A. P. Aguiar, and J. Colin N., "Trajectory-tracking and path-following controllers for constrained underactuated vehicles using model predictive control," in *2013 European Control Conference (ECC) July 17-19, 2013, Zurich, Switzerland*, 2013, pp. 1371–1376.

[15] H. K. Khalil, *Nonlinear systems*. Upper Saddle River, (N.J.): Prentice Hall, 1996. [Online]. Available: http://opac.inria.fr/record=b1091137

[2]The authors would like to acknowledge the support of José Pinto and all the members of the Underwater Systems and Technologies Laboratory (LSTS) of University of Porto, in conducting the experiments. The video to the experiments can be found at https://youtu.be/x0HBSeFwRvs