An Event-Triggered Multi-UAV Coordination Scheme for Simultaneous Tracking and Pursuit of Multiple Moving Targets

TUA A. TAMBA[1], (Member, IEEE), IMMANUEL R. SANTJOKO[1], YUL Y. NAZARUDDIN[2],(Member, IEEE), and VEBI NADHIRA[2], (Member, IEEE)

[1]Department of Electrical Engineering, Faculty of Engineering Technology, Parahyangan Catholic University, Bandung, Indonesia
[2]Department of Engineering Physics, Faculty of Industrial Technology, Institut Teknologi Bandung, Bandung, Indonesia

Corresponding author: Tua A. Tamba (ttamba@unpar.ac.id).

ABSTRACT This paper proposes event-triggered cooperative localization and pursuit control methods of multiple unmanned aerial vehicles (UAVs) for the purpose of tracking and monitoring multiple moving target vessels in a maritime patrol scenario. In the proposed method, each UAV is assumed to be equipped with an optical sensor, which is used to measure its distances while moving in the air to the target vessels moving on the water surface. In particular, each UAV combines its obtained measurements with those of other UAVs to be subsequently processed using an event-triggered distributed extended Kalman filter algorithm to obtain its estimates of target vessels' positions. The UAVs then use the position estimates of the targets to construct dynamic convex hulls that connect all of the detected moving target vessels. The multi-UAV tracker system then generates spatio-temporal reference path curves using cycloid-type trajectories with a formation geometry that is adapted in such a way to cover/contain the constructed convex hulls of the target vessels. Based on the generated reference path, a distributed cooperative control law based on the moving path following approach is finally proposed to maintain tracker-target's relative geometry that guarantee accurate tracking of the moving vessels. Extensive simulation results are presented to demonstrate the effective performance, computational efficiency, and scalability of the proposed scheme.

INDEX TERMS Range-based localization, target pursuit, multi-tracker, multi-target, moving path following

I. INTRODUCTION

Indonesia is known to be the country that contributes the highest to the total marine capture and fishery production in the Southeast Asia region. Geographically, 71% of its territory is covered by water, which provides shelter to 28.57% of the overall fish species of the world [1]. In 2020 alone, Indonesia's fishery production reached up to 6.49 mega-tonne, which accounted for approximately around 35.6% of its region's total production [2]. Consequently, Indonesia's fishery sector plays a crucial role in sustaining the livelihoods of millions of its population. Approximately more than 2.2 million Indonesian fishermen engaged in marine capture activities, many of whom are part of small-scale fisheries that are critical to support local economies [3]. Such vast marine resources not only contribute to the high biodiversity of, but also offer environmental and economical potentials to Indonesia.

Despite their promising potential, marine resources and the environment have also been observed to attract various illegal activities such as *illegal fishing*. Illegal fishing refers to any form of fishing activities that is carried out in the territorial waters of a country without proper authorization or in ways that violate laws, regulations, and environmental protections [4], [5]. In Indonesia, monitoring and enforcement of the law against illegal fishing activities are significantly challenging due to the large size of its maritime area. The Indonesia Ministry for Marine Affairs and Fisheries specifically acknowledged in a 2017 BBC interview that policing and monitoring activities such as illegal fishing at sea are almost impossible due to the remote nature and limited connectivity of Indonesian islands and communities [6]. In these regards, innovative approaches that are not only economically cost-effective but also practically efficient are needed to help support the country's efforts to tackle illegal fishing activities.

In recent years, unmanned aerial vehicles (UAVs) have been developed and used to track and pursue moving target objects [7]–[15]. Due to their agile movements and greater coverage capabilities, UAVs' deployment for surveillance and reconnaissance tasks has been proven to be more effi-

cient when compared to conventional human-led patrol approaches. One of such methods which motivates the work presented in this paper is the range-based simultaneous localization and pursuit (SLAP). In essence, SLAP is a technique for estimating the position and movement of a moving object (such as a car or vessel) using only range measurement data from sensors [11], [12], [16]. In this technique, each tracker is equipped with optical sensors and autonomous navigation systems which allow it to monitor and track a detected moving target. When extended to the case of tracking multiple targets using multiple trackers, this technique has the potential to be implemented on multi-UAV operation for continuous and adaptive surveillance of e.g. illegal vessel/fishing activities over vast maritime areas. The use of a multi-UAV system in this case is therefore expected to not only help reduce the need for extensive human resources but also provide real-time information and tracking of potential illegal fishing activities.

This paper proposes a multi-UAV coordination approach which equips a multi-tracker SLAP with a cooperation scheme for tracking moving geometric shapes formed from the estimated locations of multiple moving target vessels. Thus, rather than using single-point target assumption as in conventional SLAP method, this paper instead considers an extension which allows multiple trackers to follow moving convex hulls formed from the locations of multiple moving vessels. In the proposed method, each UAV is assumed to be equipped with optical sensors to measure its distances while moving in the air to the target vessels moving in waters. These measurements are processed to obtain estimates of the targets' state, which then subsequently used to construct convex hulls that cover all detected targets. The multi-UAV tracker system then generates spatio-temporal (S-T) curve paths as cycloid-type trajectories with a formation geometry that is adapted to the constructed convex hulls of the moving target vessels [17]. An event-triggered distributed extended Kalman filter (ET-DEKF) is used to reduce communication cost while maintaining the accuracy of the estimation of the whole state of the vessel [18]. A distributed cooperative control law based on the moving path following approach [19] is then designed to maintain the tracker-target's relative geometry to ensure accurate tracking and ensure collision avoidance.

II. SYSTEM DESCRIPTION AND PROBLEM STATEMENT

This section formulates the multi-agent systems (MAS) interaction between a group of UAVs that are assigned to cooperatively localize and pursue a group of moving USVs targets. In the following exposition, the UAVs will be referred to as *trackers* while the USVs will be referred to as *targets*. This section also discusses the cooperative localization and cooperative pursuit problems that the trackers should solve.

A. KINEMATICS MODEL OF TRACKER-TARGET MAS

Let $\{\mathcal{I}\} : \mathbf{O}_{\mathcal{I}} - \mathbf{X}_{\mathcal{I}} - \mathbf{Y}_{\mathcal{I}} - \mathbf{Z}_{\mathcal{I}}$ be an inertial frame of the considered MAS with an origin at $\mathbf{O}_{\mathcal{I}}$. Consider a number N of UAV trackers, each of which is identified by an index symbol $i = 1, 2, \ldots, N$. Let the motion characteristics of each

UAV tracker be modeled by the kinematics of its body frame $\{\mathcal{B}\}^i$ with respect to (wrt) $\{\mathcal{I}\}$. Let $\mathbf{p}_1^i = \begin{bmatrix} x^i, y^i, z^i \end{bmatrix}^{\mathrm{T}}$ be the position vector of the UAV tracker, while $\mathbf{p}_2^i = \begin{bmatrix} \phi^i, \theta^i, \psi^i \end{bmatrix}^{\mathrm{T}}$ be the attitude vector of the UAV tracker which consists of the roll, pitch, and yaw angles wrt $\{\mathcal{I}\}$, cf. FIGURE 1a. Define the state variables of each tracker as $\mathbf{p}^i = \begin{bmatrix} \mathbf{p}_1^i, \mathbf{p}_2^i \end{bmatrix}^{\mathrm{T}}$.

Let $\mathbf{u}^i = \begin{bmatrix} v_p^i, \dot{\phi}_{\mathcal{B}}^i, \dot{\theta}_{\mathcal{B}}^i, \dot{\psi}_{\mathcal{B}}^i \end{bmatrix}^{\mathrm{T}}$ be the input vector of each UAV tracker, where v_p^i is the surge velocity and $\begin{bmatrix} \dot{\phi}_{\mathcal{B}}^i, \dot{\theta}_{\mathcal{B}}^i, \dot{\psi}_{\mathcal{B}}^i \end{bmatrix}$ is the attitude velocity vector in the tracker's body frame $\{\mathcal{B}^i\}$. Using such state and input variables, the kinematics model of each UAV tracker may then be expressed as follows

$$
\begin{aligned}
\dot{x}^i &= v_p^i\, c_{\psi^i}\, c_{\theta^i}, \\
\dot{y}^i &= v_p^i\, s_{\psi^i}\, c_{\theta}^i, \\
\dot{z}^i &= v_p^i\, s_{\theta^i}, \\
\dot{\phi}^i &= \dot{\phi}_{\mathcal{B}}^i + \dot{\theta}_{\mathcal{B}}^i\, s_{\phi^i}\, t_{\theta^i} + \dot{\psi}_{\mathcal{B}}^i c_{\phi^i}\, t_{\theta^i}, \\
\dot{\theta}^i &= \dot{\theta}_{\mathcal{B}}^i\, c_{\phi^i} - \dot{\psi}_{\mathcal{B}}^i\, s_{\phi^i}, \\
\dot{\psi}^i &= \dot{\theta}_{\mathcal{B}}^i\, s_{\phi^i}\, c_{\theta^i} + \dot{\psi}_{\mathcal{B}}^i\, c_{\phi^i}\, c_{\theta^i},
\end{aligned}
\tag{1}
$$

where $s_{\phi^i} := \sin(\phi^i)$, $c_{\theta^i} := \cos(\theta^i)$, and $t_{\psi^i} := \tan(\psi^i)$.

Next, consider a group of M target vessels that should be tracked by UAV trackers, each of which is identified by an index $j = 1, 2, \ldots, M$. Let $\mathbf{q}_1^j = \begin{bmatrix} x^j, y^j, z^j \end{bmatrix}^{\mathrm{T}}$ be the position vector of the target j, while $\mathbf{q}_2^j = \begin{bmatrix} v_q^j, \psi^j, \dot{\psi}^j \end{bmatrix}^{\mathrm{T}}$ be the navigation vector of the target j consisting of the scalar surge speed, yaw, and yaw rate wrt the inertial frame $\{\mathcal{I}\}$. This paper assumes that each target movement is modeled by a planar kinematic motion ($\dot{z}^j = 0$) with constant surge speed ($\dot{v}_q = 0$) which satisfies the following differential equations.

$$
\begin{aligned}
\dot{x}^j &= v^j\, c_{\psi^j} + \mu \\
\dot{y}^j &= v^j\, s_{\psi^j} + \mu \\
\dot{\psi}^j &= \dot{\psi}^j + \mu \\
\ddot{\psi}^j &= 0^j + \mu
\end{aligned}
\tag{2}
$$

where $\mu \sim \mathcal{N}(0, \beta)$ is a Gaussian noise with mean zero and covariance matrix of β. Based on (2), the state variables of each target are then defined as $\mathbf{q}^j = \begin{bmatrix} \mathbf{q}_1^j, \mathbf{q}_2^j \end{bmatrix}^{\mathrm{T}}$ as illustrated in FIGURE 1b. It is assumed that each UAV tracker (1) is equipped with a distance sensor that provides measured distance data to each target vessel (2). Under such an assumption,

(a) Kinematic of UAV tracker   (b) Kinematic of target vessel
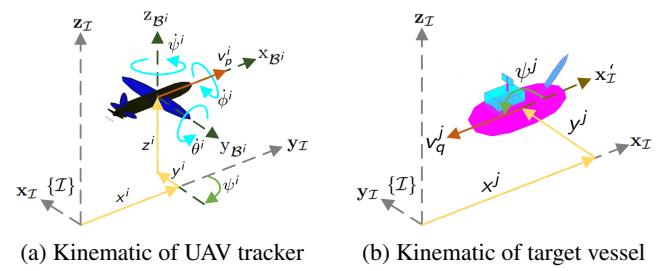
FIGURE 1: Schematic and kinematics target-tracker MAS.

this article considers the problem of controlling UAV trackers to track and pursue moving target vessels based on distance information measured between them obtained from sensors. Section II-B states this problem in a more formal manner.

B. PROBLEM SETUP AND FORMULATION

Given the kinematics of the tracker and target systems in (1) and (2), respectively, this article examines the control problem of the tracker-target MAS interaction by formulating two sub-problems, namely *cooperative localization* and *cooperative pursuit* problems. The challenge in cooperative localization problem is to ensure that the trackers' estimation $\hat{\mathbf{q}}^{i,j}(t)$ of the target vessels' state variables based on the distance measurement data obtained from the sensor converge to a consensus:

$$\lim_{t \to \infty} \left\| \hat{\mathbf{q}}^{i,j}(t) - \mathbf{q}^j(t) \right\| \leq r_e. \tag{3}$$

The trackers' state variables estimate $\hat{\mathbf{q}}^{i,j}(t)$ in (3) will be developed using a distributed extended Kalman filter (DEKF) method [18]. Meanwhile, the challenge in cooperative pursuit problem is to ensure that over time, all trackers converge and remain within a specified distance from targets, i.e.:

$$\lim_{t \to \infty} \left\| \mathbf{p}_1^i(t) - \mathbf{q}_1^j(t) \right\| \leq r_c. \tag{4}$$

In (3)-(4), r_c and r_e are some specified positive constants. This paper develops an event-triggered DEKF (ET-DEKF) scheme to achieve the cooperative localization task (3), and a moving path following control scheme to achieve the cooperative pursuit task (4). These are detailed in the following sections.

III. MOVING TARGETS LOCALIZATION & TRACKING

Since UAV trackers are tasked with locating and pursuing target vessels using distance measurement between them obtained from sensors, a filtering technique based on the discrete-time version of the target vessel model is developed and utilized in this paper. Specifically, for a sampling period of Δt, the discretization of (2) at each discrete time instant $k \in \mathbb{N}$ of the following form is considered.

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \dot{\mathbf{q}}_k \Delta t = f(\mathbf{q}_k) \tag{5}$$

$$\begin{bmatrix} x_{k+1}^j \\ y_{k+1}^j \\ z_{k+1}^j \\ v_{q,k+1}^j \\ \psi_{k+1}^j \\ \dot{\psi}_{k+1}^j \end{bmatrix} = \begin{bmatrix} x_k^j + v_{q,k}^j \, c_{\psi_k^j} \, \Delta t \\ y_k^j + v_{q,k}^j \, s_{\psi_k^j} \, \Delta t \\ z_k^j \\ v_{q,k}^j \\ \psi_k^j + \dot{\psi}_k^j \, \Delta t \\ \dot{\psi}_k^j \end{bmatrix} \tag{6}$$

with a Jacobian matrix of (5) at time k is defined as follows.

$$F_k \equiv \left. \frac{\partial f}{\partial \mathbf{q}} \right|_{\mathbf{q}_k} = \begin{bmatrix} 1 & 0 & 0 & c_{\psi_k^j} \Delta t & -v_{q,k}^j \, s_{\psi_k^j} \, \Delta t & 0 \\ 0 & 1 & 0 & s_{\psi_k^j} \Delta t & v_{q,k}^j \, c_{\psi_k^j} \, \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Let $h_k^{i,j}$ be the observed distance model from UAV tracker i to target vessel j at discrete time k of the following form.

$$h_k^{i,j} = \left\| \mathbf{q}_{1,k}^j - \mathbf{p}_{1,k}^i \right\| \tag{8}$$

where $\| \cdot \|$ denotes the Euclidean norm. Assume the sensor data are corrupted by white Gaussian noise whose magnitude increases as the tracker-target distance gets larger. The measurement of each tracker is thus assumed to be of the form

$$m_k^{i,j} = h_k^{i,j} + h_k^{i,j} \, \mathbf{w}_k \tag{9}$$

where $\mathbf{w}_k \sim \mathcal{N}(0, \sigma)$ is a Gaussian noise with mean zero and covariance matrix of σ. The linearized model (5) and the corresponding measurement model (9) are used by each tracker to implement an ET-DEKF algorithm to estimate the state variables and localize the position of each target vessel. This algorithm uses the Jacobian matrix of (8) below.

$$H_k = \left. \frac{\partial h_k^{i,j}}{\partial \mathbf{q}} \right|_{\mathbf{q}_k} = \left[\frac{(\mathbf{q}_{1,k}^j - \mathbf{p}_{1,k}^i)^{\mathrm{T}}}{\left\| \mathbf{q}_{1,k}^j - \mathbf{p}_{1,k}^i \right\|}, \quad 0_{1x3} \right] \tag{10}$$

A. EVENT-TRIGGERED COOPERATIVE LOCALIZATION

The objective of the cooperative localization task in (3) is achieved through the implementation of an ET-DEKF scheme, which ensures that data exchange between trackers for the implementation of the local DEKF by each tracker is carried out only when necessary. In this regard, both computation effort and communication resource utilization can be reduced during the tracker-target MAS interaction. To this end, consider the target state and measurement equations in (5) and (9), respectively. Let $Y_k^j \equiv \{m_l^{i,j}\}_{l=0}^k$ be the measured range data obtained by the sensor until time k. Denote with $\ell(\mathbf{q}_k^{i,j}|Y_{k-1}^j) \sim \mathcal{N}(\hat{\mathbf{q}}_{k|k-1}^{i,j}, \mathcal{P}_{k|k-1})$ the prior density and with $\ell(\mathbf{q}_k^{i,j}|Y_k^j) \sim \mathcal{N}(\hat{\mathbf{q}}_{k|k}^{i,j}, \mathcal{P}_{k|k}^{i,j})$ the posterior density of the target's state estimate at time k, respectively, where $\mathcal{P}_{k|k-1}^{i,j}$ is the covariance matrix at time k given the measurement data until time $k-1$. Define the information vector $\xi_k^{i,j}$ and matrix $\Omega_k^{i,j}$ below, which capture the uncertainties in the state estimation.

$$\xi_{k|k-1}^{i,j} = \left(\mathcal{P}_{k|k-1}^{i,j} \right)^{-1} \hat{\mathbf{q}}_{k|k-1}^{i,j}, \text{ and } \xi_{k|k}^{i,j} = \left(\mathcal{P}_{k|k}^{i,j} \right)^{-1} \hat{\mathbf{q}}_{k|k}^{i,j}$$

$$\Omega_{k|k-1}^{i,j} = \left(\mathcal{P}_{k|k-1}^{i,j} \right)^{-1} \tag{11}$$

Algorithm 1 summarizes the ET-DEKF that is used by the tracker i to estimate the state variables $(\hat{\mathbf{q}}^{i,j})$ of the moving target vessel j. It consists of three main phases that are iterated recursively, namely (i) Measurement-Correction, (ii) ET Communication-Fusion, and (iii) State Estimation Prediction.

The first Measurement-Correction phase is intended to update the measurement and the corresponding information vector/matrix whenever new sensor data is obtained. This is done by first updating the measurement in accordance to (12)

$$\tilde{y}_k^{i,j} = m_k^{i,j} - h_k^{i,j} \left(\hat{\mathbf{q}}_{k|k-1}^{i,j} \right) + H_k^{i,j} \, \hat{\mathbf{q}}_{k|k-1}^{i,j} \tag{12}$$

Algorithm 1 ET-DEKF for tracker i

1: **procedure** Initialization
2:     At $k = 0$, initialize $\hat{\mathbf{q}}_{1|0}^{i,j}$, $\mathcal{P}_{1|0}^{i,j}$
3:     $\Omega_{1|0}^{i,j} = \left(\mathcal{P}_{1|0}^{i,j}\right)^{-1}, \xi_{1|0}^{i,j} = \Omega_{1|0}^{i,j}\,\hat{\mathbf{q}}_{1|0}^{i,j}$
4:     **return** $\hat{\mathbf{q}}_{1|0}^{i,j}$, $\Omega_{1|0}^{i,j}$, $\xi_{1|0}^{i,j}$

    At each k, repeat the following procedures:
5: **procedure** Measurement-Correction
6:     **if** new range sensor data is received **then**
7:         Compute measurement update $\tilde{y}_k^{i,j}$ using (12)
8:         Correct information data $\tilde{\xi}_k^{i,j}$ and $\tilde{\Omega}_k^{i,j}$ using (13)
9:     **else** $\tilde{\xi}_k^{i,j} = \xi_{k|k-1}^{i,j}$, $\tilde{\Omega}_k^{i,j} = \Omega_{k|k-1}^{i,j}$
10: **procedure** ET Communication-Fusion
11:     Define $\alpha(k)$ in (15) and evaluate ET logic in (16)
12:     **if** $\max\{\alpha\} > \Pi$ **then**
13:         **procedure** Communication
14:             Transmit message $\mathcal{M}_e(k)$ to neighbors
15:         **procedure** Fusion
17:             Perform data fusion using (14) to update local
18:             estimate of each tracker
19:             **return** $\xi_{k|k}$, $\Omega_{k|k}$, $\hat{\mathbf{q}}_{k|k} = \Omega_{k|k}^{-1}\,\xi_{k|k}$
20:     **else**
21:         **procedure** Propagation
22:             **return** $\Omega_k^{j,i} = \tilde{\Omega}_{k|k-1}^{j,i}$, $\hat{\mathbf{q}}_{k|k} = \Omega_{k|k}^{-1}\,\tilde{\xi}_{k|k}$
23: **procedure** Prediction
24:     Compute state and information prediction using (17)
25:     **return** $\hat{\mathbf{q}}_{k+1|k}$, $\Omega_{k+1|k}$, $\xi_{k+1|k}$

---

and then correcting the information vector/matrix based on the innovation induced by new sensor data as in (13).

$$\tilde{\xi}_k^{i,j} = \xi_{k|k-1}^{i,j} + \left(H_k^{i,j}\right)^\top V^{i,j}\tilde{y}_k^{i,j}$$
$$\tilde{\Omega}_k^{i,j} = \Omega_{k|k-1}^{i,j} + \left(H_k^{i,j}\right)^\top V^{i,j} H_k^{i,j} \tag{13}$$

In (13), $V^{i,j}$ is a positive definite matrix of suitable dimension which accounts for uncertainties in the sensor measurements.

The ET Communication-Fusion in the second phase is intended to manage the inter-agent communication and data exchange among trackers, as well as the fusion of such exchanged data within each tracker. Specifically, each tracker performs a communication to transmit a message $\mathcal{M}_e(k) \triangleq \left\{ \tilde{\xi}_k^{i,j}, \tilde{\Omega}_k^{i,j}, \gamma_k^i \right\}$ containing its latest local estimate of the probability density function (PDF) parameters and path angle γ_k^i of the target to neighboring trackers. Subsequently, each tracker then performs fusion (14) of the obtained data to update and correct its local estimate of the PDF parameters of the target.

$$\xi_{k|k}^{i,j} = \sum_{n \in \mathcal{N}_{in}^i \cup \{i\}} \pi^{i,n} \tilde{\xi}_k^{n,j}, \quad \Omega_{k|k}^{i,j} = \sum_{n \in \mathcal{N}_{in}^i \cup \{i\}} \pi^{i,n} \tilde{\Omega}_k^{n,j} \tag{14}$$

where $\pi^{i,n} := y_k^{n,j}/(m_k^{i,j} + y_k^{n,j})$ is a consensus weighting parameter whose value depends on the relative proximity of each tracker to the target, and satisfies the following conditions: (i) $\pi^{i,n} > 0$, and (ii) $\sum_{n \in \mathcal{N}_{in}^i \cup \{i\}} \pi^{i,n} = 1$.

Note that these communication and fusion procedures are essential in distributed estimation of MAS to improve the estimation accuracy of the target's state and ensure the attainment of estimation consensus among all agents. However, performing these procedures at each time new measurement data is received will require large communication and computational resources, especially when the number of MAS agents becomes increasingly large [20]. To address this issue, an ET scheme is used to schedule the execution of the Communication-Fusion procedure only when necessary (i.e. certain condition is satisfied). In particular, a condition on the error covariance matrix α at each time k of the form

$$\alpha(k) \equiv G(P_{k|k-1}^{i,j})G^T \tag{15}$$

with G being a weighting matrix of suitable dimension, is used as a criterion to decide the next execution time instants of the Communication-Fusion procedure [21]. In this regard, let $\{k_\iota\}_{\iota \subseteq \mathbb{N}}$ with $0 = k_0 < k_1 < k_2 < \cdots < k_\iota < k_{\iota+1} < \ldots$ denote the set of time instants when the Communication-Fusion procedure is performed. The ET scheme in this paper uses α as triggering parameter according to the logic below.

$$\alpha(k) := \alpha(k_\iota), \quad \text{for } k_\iota \le k < k_{\iota+1}$$
$$k_{\iota+1} = \inf\{k > k_\iota \mid \alpha(k) \le \Pi\} \tag{16}$$

where Π is a designed constant threshold parameter. It can be seen in (16) that the proposed logic triggers data exchange between trackers whenever the condition $(\max(\alpha) > \Pi)$ is *violated*. Thus, so long as such a condition is not satisfied, the Communication-Fusion procedure is skipped, and each tracker proceeds to perform only the Propagation phase to maintain its state evolution independently. The proposed ET-DEKF scheme thus balances efficient computational utilization with effective communication engagement for each tracker by exchanging data among them only when necessary.

The State Estimation Prediction in the third phase updates the predictions of the state variable estimate and information vector/matrix of the tracker according to the equations below.

$$\hat{\mathbf{q}}_{k+1|k}^{i,j} = f(\hat{\mathbf{q}}_{k|k}^{i,j})$$
$$\Omega_{k+1|k}^{i,j} = W - W F_k \left(\Omega_{k|k}^{i,j} + F_k^\top W F_k\right)^{-1} F_k^\top W \tag{17}$$
$$\xi_{k+1|k} = \Omega_{k+1|k}\,\hat{\mathbf{q}}_{k+1|k}$$

where W is the inverse of the covariance matrix of the process noise. Note that the resulting error of the state variable estimates produced by DEKF is guaranteed to be bounded as long as the estimation is initialized sufficiently close to the true one and the noises are sufficiently small. Finally, each time new sensor data are received, the algorithm returns to the first Measurement-Correction phase and is reiterated accordingly.

B. MOVING PATH FOLLOWING FOR TARGET PURSUIT

Based on the estimated state variables of the targets that are obtained using the ET-DEKF method, the objective of the trackers is to track and follow the targets as they move. For this purpose, this paper first defines time-varying geometric

(a) Two target vessels (b) Three target vessels
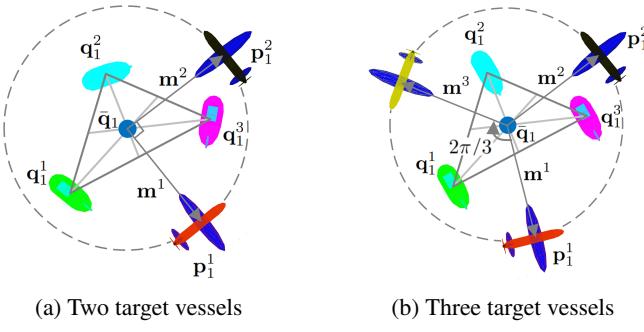
FIGURE 2: Illustrations of trackers-targets geometries.
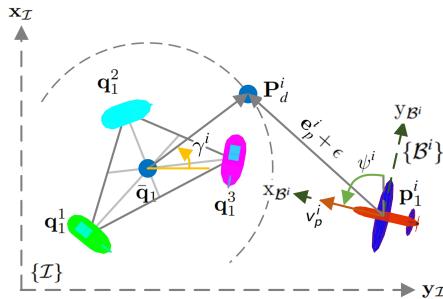
FIGURE 3: Illustration of target pursuit method (3 targets).

shapes that cover the area enclosed by the position coordinates of the targets. As illustrated in FIGURE 2, such shapes are defined as convex hulls that are constructed from the position coordinates of the targets as they move over time. For the case of multi-target pursuit, this paper uses the centroid of the targets' convex hull as the focal coordinate of the trackers' trajectories, while also optimizing the available information to enhance the estimation of the state variables of the targets. Each of such convex hulls is denoted as $\bar{\mathbf{q}} = [\bar{\mathbf{q}}_1, \bar{\mathbf{q}}_2]^\top$, and is defined as the mean value of the targets' state variables below.

$$\bar{\mathbf{q}} = \frac{1}{M} \sum_{j \in \mathbb{N}_j^M} \mathbf{q}^j. \tag{18}$$

Based on the constructed convex hulls of the targets, the trackers then define cycloid-type reference shapes that should be tracked and followed to ensure the containment of the targets' geometric shape. For the case of single-target pursuit, the tracker uses a fixed-radius circular trajectory $r = \zeta$. However, for the case of multi-target pursuit, the radius of the circular shape is set to be adjustable in accordance with the varying distance between the targets and the focal point. This adjustment is made while maintaining a fixed distance of ζ outside the farthest target from the centroid, to ensure that all targets remain inside the tracker's trajectory (cf. Fig. 2). The radius of the trackers' trajectory is thus defined as

$$r = \max\left(\left\|\bar{\mathbf{q}}_1 - \mathbf{q}_1^j\right\|\right) + \zeta. \tag{19}$$

This paper designs a planar, circular-shaped reference trajectory for the trackers to ensure the encirclement of the centroid of the targets' geometric shape movement. Such a reference trajectory essentially provides a clear view of the targets'

movement (cf. FIGURE 3). Thus, let $\mathbf{s}(\gamma)$ be a path in the inertial frame $\{\mathcal{I}\}$ with cycloid-type trajectories of the form

$$\mathbf{s}(\gamma^i) = \left[r\cos\left(\gamma^i + \gamma_0^i\right), r\sin\left(\gamma^i + \gamma_0^i\right), r_{\mathbf{z}}\right]^\top, \tag{20}$$

where $r_{\mathbf{z}}$ is the height of the trajectory as measured from the target's position, and γ^i is the path angle parameter for $\mathbf{s}$ with initial γ_0^i. Borrowing the idea from the moving path following (MPF) control method proposed in [22], a reference spatio-temporal (S-T) curve $\mathbf{p}_{\mathrm{d}}$ along the path around the centroid of estimated target position $\bar{\hat{\mathbf{q}}}_1^i$ can be constructed as follows.

$$\mathbf{p}_{\mathrm{d}}^i\left(\gamma^i, t\right) = \mathbf{s}\left(\gamma^i\right) + \bar{\hat{\mathbf{q}}}_1^i(t). \tag{21}$$

For multi-tracker scenarios, the paths $\mathbf{s}$ can be parameterized so as to ensure that the S-T curves produce the desired geometric formations relative to the target. Specifically, the parameterization is done in this paper so that $\gamma_0^1 - \gamma_0^2 = \pi/2$ for $N = 2$ and $\gamma_0^{[i_1]} - \gamma_0^{[i_2]} = 2\pi/N$ for $N \geq 3$, where i_1 and i_2 denote the indices of any two adjacent trackers [23].

To ensure cooperative motions of the trackers in achieving the objective (4), the following distributed control law for the correction speed $\omega_{\mathrm{c}}^i \equiv \dot{\psi}_{\mathrm{c}}^i$ is used as a consensus protocol.

$$\omega_{\mathrm{c}}^i = \begin{cases} -k_c \sum_{n \in \mathcal{N}_{\mathrm{in}}^i} \left(\gamma^i - \gamma^{[n]}\right), & \text{if } \max(\alpha) > \Pi, \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

where $k_c > 0$ is a designed coupling gain. Using such a corrected speed, each tracker's desired speed ω_{d} for tracking the path angle parameter γ^i can be computed as follows.

$$\omega_{\mathrm{d}}^i = \bar{\omega} + \omega_{\mathrm{c}}^i, \tag{23}$$

where $\bar{\omega}$ denotes the desired speed for all γ^i. Building upon the work in [13], this paper implements two controllers on each tracker, namely (i) a target tracking control signal $\mathbf{u}^i$ in the tracker kinematic (1), and (ii) a control signal (22) which ensures the tracker encircle the target's motion shape and satisfies $\dot{\gamma} = \omega_{\mathrm{d}}$. As illustrated in FIGURE 3, the pursuit position error $\mathbf{e}_{\mathrm{p}}$ of each tracker can be defined with respect to its body frame $\{\mathcal{B}\}^i$ as the difference between the tracker's position and the desired trajectory, i.e.

$$\mathbf{e}_{\mathrm{p}}^i = R^\top\left(\mathbf{p}_2^i\right)\left(\mathbf{p}_1^i - \mathbf{P}_d^i\right) - \boldsymbol{\epsilon}, \tag{24}$$

where $R\left(\mathbf{p}_2^i\right)$ is the rotation matrix of $\{\mathcal{B}\}^i$ wrt $\{\mathcal{I}\}$, and $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \epsilon_3]$ is a vector with arbitrarily small-valued elements which triggers the tracker to move towards the reference circular trajectory to achieves the objective in (4). In this regard, a stabilizing control input $\mathbf{u}^i$ below for the tracker kinematic model can be used drive the error $\mathbf{e}_{\mathrm{p}}$ towards zero.

$$\mathbf{u}^i = \bar{\Delta}\left(R^\top\left(\mathbf{p}_2^i\right)\left(\mathbf{s}'\left(\gamma^i\right)\omega_{\mathrm{d}}^i + \bar{\hat{\mathbf{q}}}_1^i\right) - K_{\mathrm{p}}\hat{\mathbf{e}}_{\mathrm{p}}^i\right), \tag{25}$$

where K_{p} is positive definite matrix and $\bar{\Delta} = \Delta^\top\left(\Delta\Delta^\top\right)^{-1}$, in which Δ is a matrix that is defined as follows

$$\Delta = \begin{bmatrix} 1 & 0 & -\epsilon_3 & \epsilon_2 \\ 0 & \epsilon_3 & 0 & -\epsilon_1 \\ 0 & -\epsilon_2 & \epsilon_1 & 0 \end{bmatrix}. \tag{26}$$
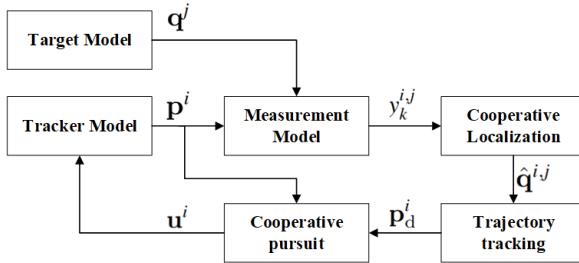
FIGURE 4: The schematic of the overall system.

C. OVERALL SYSTEM ARCHITECTURE

FIGURE 4 shows the block diagram schematic of the proposed cooperative localization and pursuit control of targets vessels that move along some target trajectories. Using an onboard range sensor, each UAV tracker first measures the distance, $m_k^{i,j}$, to each target. The measured data are then used in the proposed ET-DEKF algorithm to estimate the state variables of each moving target $\hat{q}^{i,j}$. Based on these estimates, the proposed scheme constructs a convex hull which encapsulates the detected locations of the targets, and designs an S-T curve as a reference trajectory for cooperative tracking purposes, to ensure the containment of the geometric shape of the constructed convex hull trajectory p_d^i. The proposed cooperative pursuit control algorithm then compares the desired S-T curve trajectory with the current state variables of the trackers, and then uses it as a basis to compute suitable control input signals u^i that render the error between the two minimum/zero.

IV. SIMULATION EXPERIMENTS

This section presents two simulation cases, namely: (a) small-scale target-tracker MAS involving at most three UAV trackers and three target vessels, and (b) medium-scale target-tracker MAS involving up to ten trackers and targets. In the small-scale case (a), the targets are simulated to follow two types of movement scenarios, namely: (i) synchronous motion, where all targets move in coordinated patterns, and (ii) spreading motion, where targets gradually diverge from each other toward different directions. In addition, the simulations in case (a) were examined using single, two, and three trackers to evaluate whether an increase in the number of trackers will improve the accuracy of the resulting state estimation and pursuit control schemes. Meanwhile, the simulations in case (b) are intended to numerically illustrate the computational complexity involved in the implementations of the proposed method as the number of MAS agents increases. In both cases, the motions of the UAV trackers and the target vessels were simulated on the basis of kinematic models in Section II.

A. SMALL-SCALE TRACKER-TARGET MAS SIMULATION

This simulation case considers MAS interaction between at most three UAV trackers and three target vessels. For each tracker, the initial condition is set to: $p_0^1 = [55, -60, 25, \pi, 0, 0]^\top$, $p_0^2 = [80, 35, -50, -\frac{\pi}{4}, 0, \frac{\pi}{2}]^\top$, $p_0^3 = [-70, 75, -4, 0, 0, 0]^\top$, with a maximum and minimum control input of the form $u_{\mathbf{min}}^i = [0, -1, -1, -1]^\top$, and $u_{\mathbf{max}}^i =$

$[10, 1, 1, 1]^\top$, respectively. The covariance matrix of the process noise is set to $W = 10^{-4} \times \texttt{diag}\{10, 10, 10, 10, 1, 0.1\}$, and the path angle parameter of the tracker is bounded by $\ddot{\gamma}_{\mathbf{min}}^i = -0.01$ and $\ddot{\gamma}_{\mathbf{max}}^i = 0.01$, respectively. With regard to trackers, the initial condition of each agent is set to $\hat{q}_0^{1i} = [5, 0, 0, 0.2, 0.2, -0.1]^\top$, $\hat{q}_0^{2i} = [15, 10, 0, 0.2, 0.2, -0.1]^\top$, $\hat{q}_0^{3i} = [-10, 15, 0, 0.2, 0.2, -0.1]^\top$, with the noise parameters of: $\hat{P}_0^{i,j} = \texttt{diag}\{100, 100, 100, 1, 1, 0.1\}$, $\sigma = 0.005$, $V^{i,j} = 1$. The ET-DEKF parameters in Algorithm 1 are: $\Pi = 5$, $G = 0.1 \times \texttt{diag}\{5, 5, 5, 5, 1, 1\}$, $\Omega = 20$, while the control signal parameters used for the pursuit control are as follows: $\psi^j \in [0, 2\pi]$, $v \geq 0$, $r_{\mathbf{z}} = 20$, $k_c = 0.1$, $k_\gamma = 200$, $k_p = \texttt{diag}\{0.5, 0.5, 0.5\}$, $\bar{\omega} = 0.1$, and $\epsilon = [-1, -1, 0]^\top$.

Simulations are conducted for two motion scenarios of the targets, namely synchronous (sync.) and spreading (spread.) motions. In the synchronous motion scenario, each target maintains collective formation according to trajectories (27).

$$\begin{aligned} q_1^1(t) &= [30\sin(0.01t), 0.1t, 0]^\top \\ q_1^2(t) &= [30\sin(0.01t) + 10, 0.1t + 10, 0]^\top \quad (27) \\ q_1^3(t) &= [30\sin(0.01t) - 10, 0.1t + 10, 0]^\top \end{aligned}$$

In the spreading motion scenario, each target follows a distinct trajectory and diverges from each other over time according to the following trajectories.

$$\begin{aligned} q_1^1(t) &= [30\sin(0.01t), -0.1t, 0]^\top, \\ q_1^2(t) &= [20\sin(0.01t) + 10, 0.15t + 10, 0]^\top, \quad (28) \\ q_1^3(t) &= [-30\sin(0.01t) - 10, 0.1t + 10, 0]^\top. \end{aligned}$$

For both scenarios, simulations were conducted for a total duration of 400 seconds, with a sampling interval of 0.1 second, and sensor measurement interval of 2 seconds.

1) Simulation Results of Synchronous Motion Scenario

Simulations were first conducted to examine the performance of the proposed DEKF method for the case when one, two, and three tracker UAVs are used to localize three target vessels that are moving according to (27). FIGURE 5 plots the position estimation errors of the three targets when only one tracker is deployed. It can be seen in these plots that the estimation errors tend to decay toward zero. When two trackers are deployed, the resulting estimation errors as depicted in FIGURE 6 are reduced and decay faster. Such improvements in the resulting estimation errors are achieved due to the additional information obtained by each tracker through data exchange and fusion with another tracker. Finally, when three
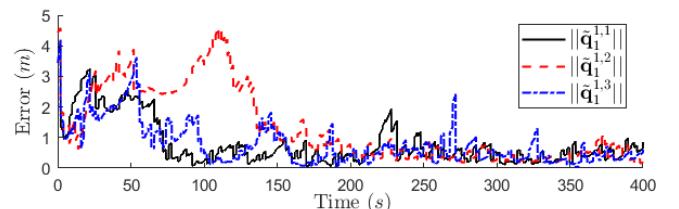

FIGURE 5: Sync. targets' estimation errors (1 tracker).
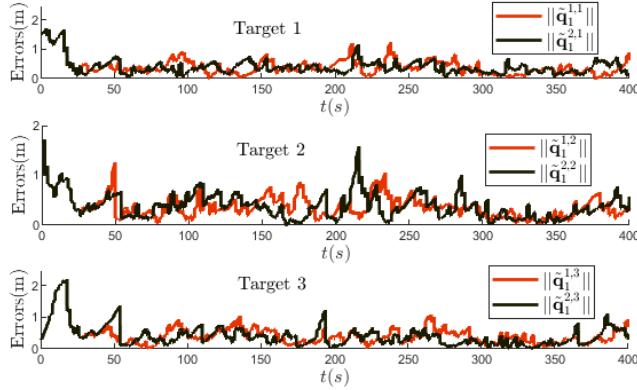
FIGURE 6: Sync. targets' estimation errors (2 trackers).
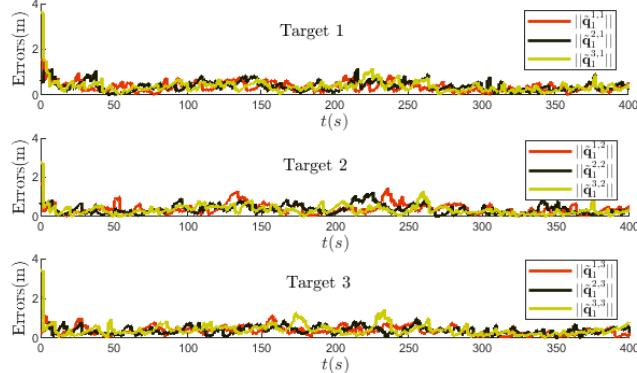
FIGURE 7: Sync. targets' estimation errors (3 trackers).

FIGURE 8: Activation events between two sync. trackers.

FIGURE 9: Activation events between three sync. trackers.

trackers are deployed, the estimation errors of each tracker as shown in FIGURE 7 are further decreased. The root mean square (RMS) values of the estimation error when using one, two, and three trackers are 0.491, 0.476, and 0.441, respectively. These results demonstrate the increased performance of the proposed DEKF scheme as the number of deployed trackers increases. Note also that the use of an ET scheme to trigger the activation of data exchange between trackers also improves the computational efficiency of the proposed ET-DEKF cooperative localization method. In particular, FIGURE 8 and FIGURE 9 illustrate the events of communication activations for data exchange when two and three trackers are deployed, respectively. In these figures, the percentages of the total simulation time during which such events were triggered between the trackers are 8.46% and 6.97% when two and three trackers are used, respectively. It can be seen that the number of inter-agent communication events reduces as the number of employed trackers increases. This suggests that the proposed ET-DEKF method requires trackers to exchange data only when needed, and the number of such data exchange instances reduces as the number of trackers increases.

Based on the obtained estimates of the targets' positions, the trackers then proceed with performing the pursuit and tracking control tasks. In this research, the pursuit and tracking control are performed with respect to the centroid of the convex hull that is formed by the targets' position estimates at each sampling time. This strategy is particularly useful when the number of trackers is less than that of the targets.

The construction of such a convex hull for the case of single tracker is illustrated in FIGURE 10. Note that the corresponding convex hull is a triangle that is constructed every certain period of time from the position estimates of the three targets. Using these convex hulls, a cycloid-type reference trajectory (dashed blue line) to be followed by the trackers is constructed as line segments that connect the centroids of the propagating triangles. A similar procedure can be used for different number of trackers, as illustrated in FIGURE 11 for the case of using two trackers.

Given the constructed convex hulls and the corresponding reference trajectory, each tracker then generates a S-T curve of the form (20) which encapsulates the convex hull that has the targets' position estimates as its vertices. The generated
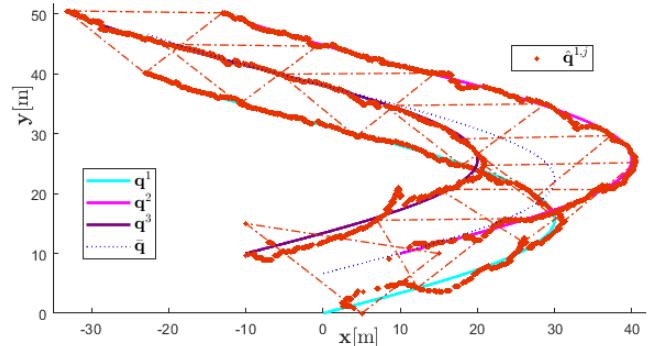

FIGURE 10: Sync. targets' convex hulls and the reference trajectory construction when using one tracker.

cycloid-type S-T curves for the case when using one, two, and three trackers are illustrated in FIGURE 12a, FIGURE 12b, and FIGURE 12c, respectively. Note in these figures that the centers of the resulting S-T curves are shifting in accordance with the movements of the centroid of the encapsulated moving targets' convex hulls.
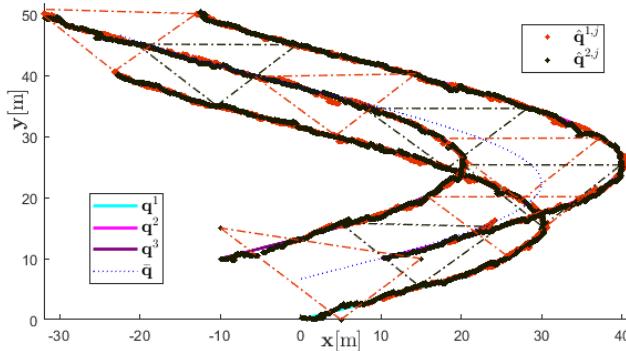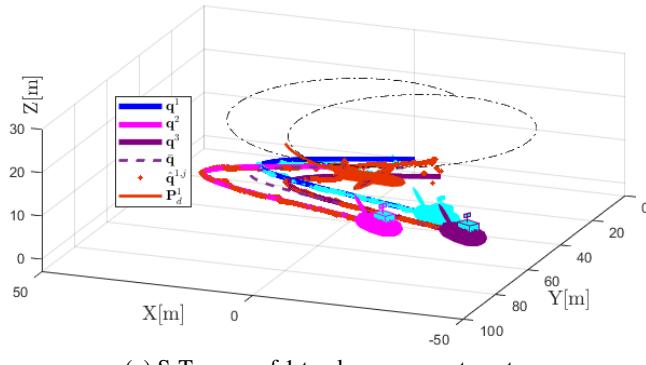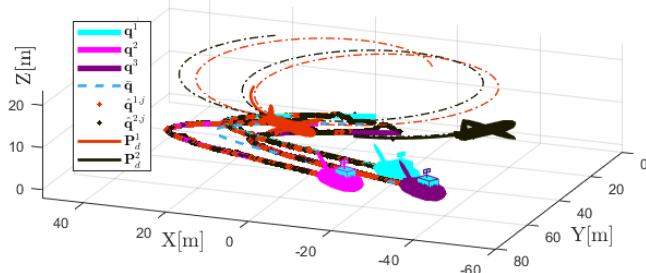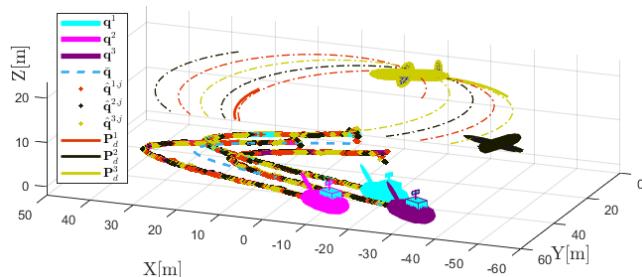

FIGURE 11: Sync. targets' convex hulls and the reference trajectory construction when using two trackers.

(a) S-T curve of 1 tracker on sync. targets.

(b) S-T curves of 2 trackers on sync. targets.

(c) S-T curves of 3 trackers on sync. targets.

FIGURE 12: S-T curves illustration of sync. targets.

Using the generated reference trajectory, each tracker then implements the tracking control input (25). The resulting pursuit and tracking errors when one, two, and three trackers are deployed to pursue the three moving targets are shown in FIGURE 13a, FIGURE 13b, and FIGURE 13c, respectively, and the corresponding control inputs generated for these cases are shown in FIGURE 14, FIGURE 15, and FIGURE 16, respectively. It can be seen in these figures that the pursuit error for each tracker decays in an exponential manner towards zero within less than 25 time steps. More specifically, the resulting RMS values of the tracking error when using one, two, and three trackers are 1.119, 0.500, and 0.475,

(a) One tracker

(b) Two trackers

(c) Three trackers
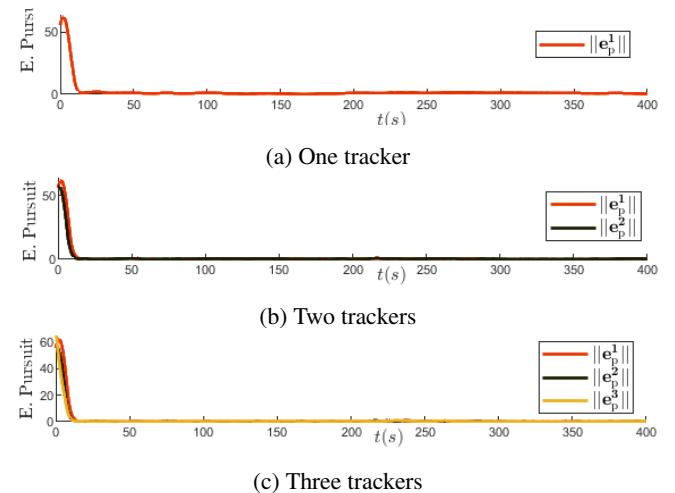
FIGURE 13: Illustration of the trackers' pursuit errors.
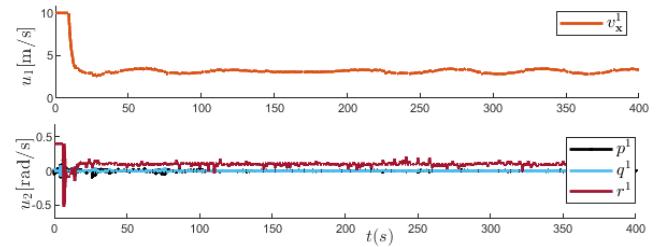
FIGURE 14: Control inputs in sync. targets (1 tracker).
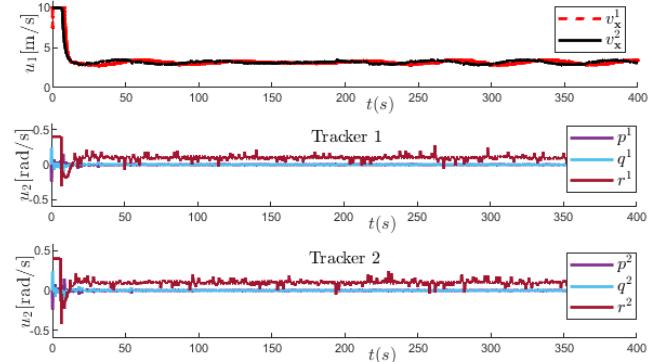
FIGURE 15: Control inputs in sync. targets (2 trackers).
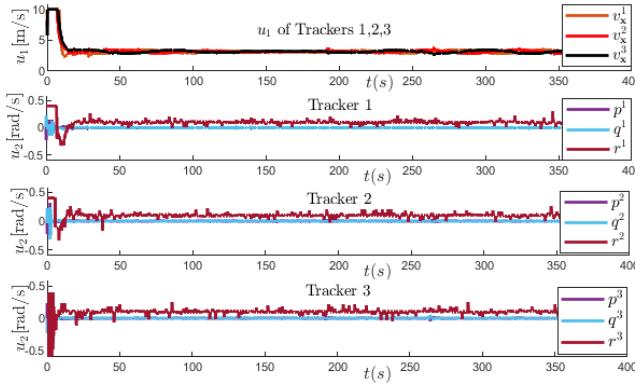
FIGURE 16: Control inputs in sync. targets (3 trackers).

respectively. These demonstrate the increased performance of the proposed control methods as the accuracy of the targets' position estimations are increased due to the implementation of data fusion scheme in the developed ET-DEKF method.

2) Simulation results of Spreading Motion Scenario

As in the synchronous motion, simulations of the spreading target motion were first performed to evaluate the performance of the proposed ET-DEKF method in localizing targets that move according to the trajectories in (28). FIGURE 17, FIGURE 18, and FIGURE 19, respectively, show the position estimation errors of the three targets when one, two, and three trackers are deployed. The RMS values of these estimation errors are 0.737, 0.4947, and 0.468, respectively. Based on these results, a similar conclusion can be drawn in the case of spreading motion to that of the synchronous case, where the performance of the proposed DEKF method increases as the number of trackers deployed increases. In addition, FIGURE 20 and FIGURE 21 show the events of communi-
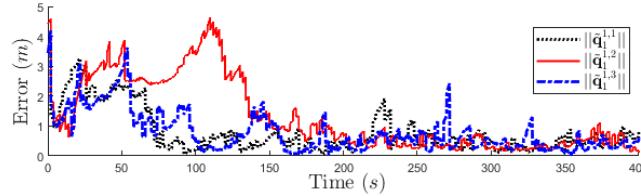
FIGURE 17: Spread. targets' estimation errors (1 tracker).
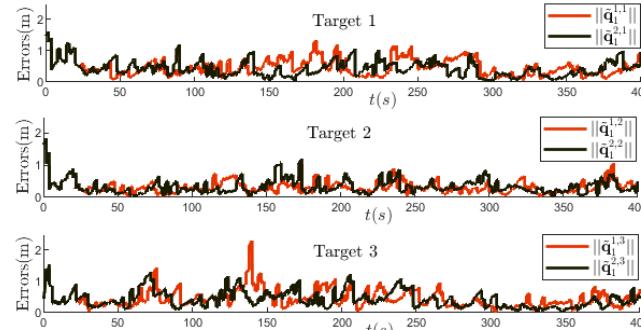
FIGURE 18: Spread targets' estimation errors (2 trackers).
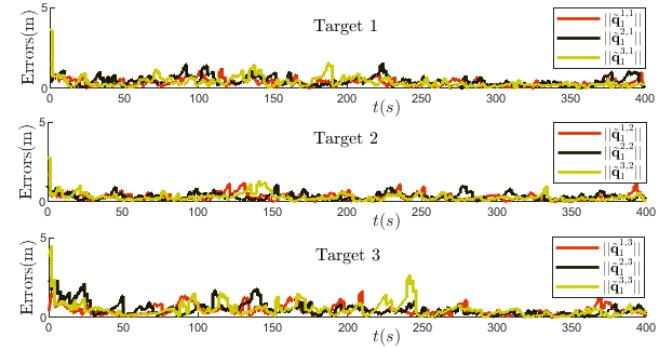
FIGURE 19: Spread. targets' estimation errors (3 trackers).

cation activation when two and three trackers, respectively, are deployed, with the percentages of the total simulation time during which such events were triggered are 7.96% and 6.97%, respectively. These results also agree with those of the synchronous motion in which the proposed ET-DEKF method requires trackers to exchange data only when needed, and the number of such exchange events reduces as the number of trackers deployed increases.

Using the estimates of the targets' positions obtained from the ET-DEKF method, the trackers then proceed with performing the pursuit and tracking control tasks. FIGURE 22 shows the resulting pursuit errors when up to three trackers are deployed to track three targets moving in spreading motions. The corresponding control inputs when using one, two,
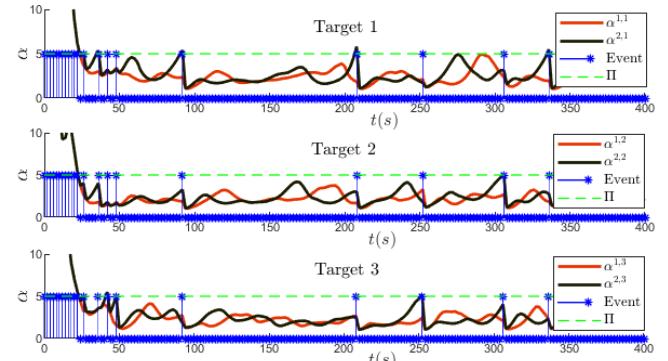
FIGURE 20: Activation events between 2 spread. trackers.
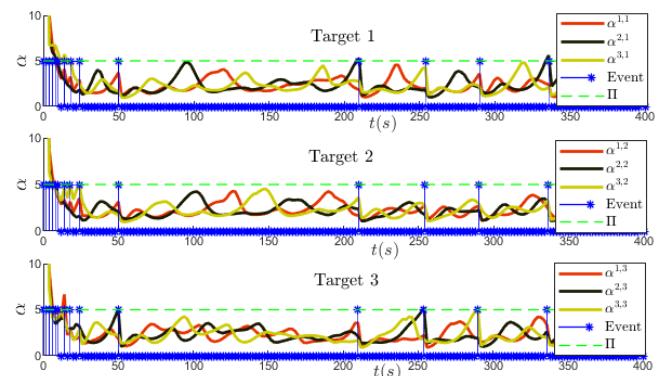
FIGURE 21: Activation events between 3 spread. trackers.

(a) One tracker

(b) Two trackers
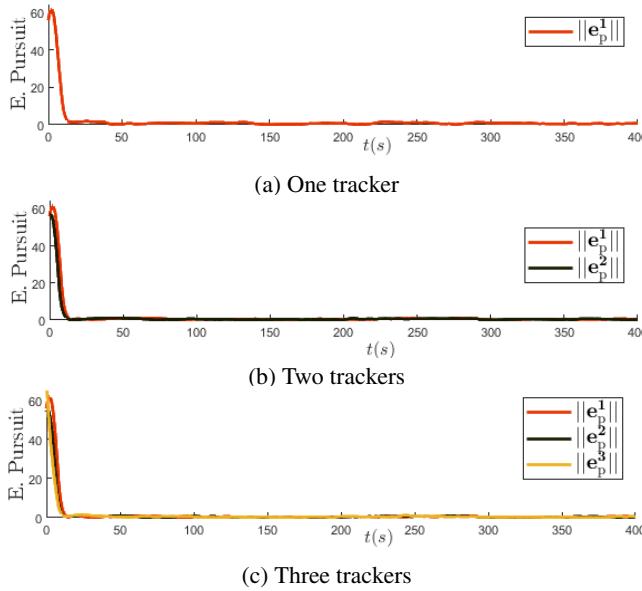
(c) Three trackers

FIGURE 22: Pursuit errors for spreading motion targets.
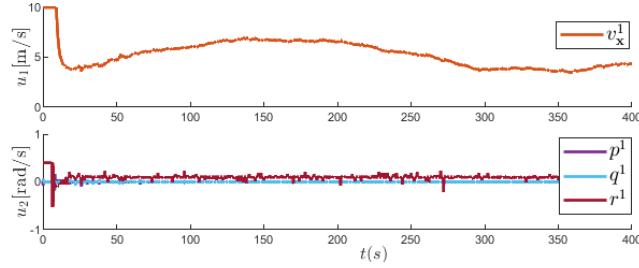
FIGURE 23: Control inputs in spread. targets (1 tracker).
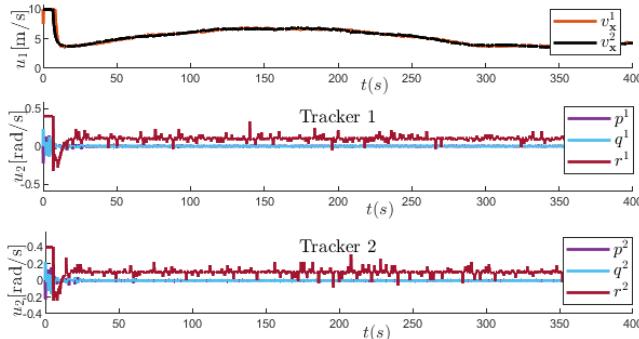
FIGURE 24: Control inputs in spread. targets (2 trackers).

FIGURE 25: Control inputs in spread. targets (3 trackers).

(a) S-T curve of 1 tracker on spread. targets.

(b) S-T curves of 2 trackers on spread. targets

(c) S-T curves of 3 trackers on spread. targets

FIGURE 26: S-T curves illustration of spread. targets.

and three trackers are shown in FIGURE 23, FIGURE 24, and FIGURE 25, respectively, which produce RMS values of the tracking errors of 1.268, 0.542, and 0.524, respectively. It can be observed in these plots that the control inputs remain similar across all cases, which is caused by consistent behavior among targets trajectories. These results also agree with those of the synchronous motion case, where the performance of the proposed pursuit and tracking control methods improves as the number of trackers increases. FIGURE 26 illustrates the S-T curves of the resulting tracking trajectories. When using a tracker, the trajectory in FIGURE 26a remains bounded to the target centroid of the specified cycloid curve radius.

When using two trackers, the trajectory of the second tracker in FIGURE 26b shows a phase shift of approximately $\pi/2$ as designed. When using three trackers, the trajectories in FIGURE 26c show that each tracker exhibits a phase shift of approximately $2\pi/3$ to achieve the desired coordination.

TABLE 1: Summary of simulation results.

	One tracker		Two trackers		Three trackers	
	Sync	Spread	Sync	Spread	Sync	Spread
Comm(%)	-	-	8.46	7.96	6.97	6.97
RMS($\mathbf{e}_{\mathrm{p}}^i$)	0.491	0.737	0.476	0.4947	0.441	0.468
RMS($\tilde{\mathbf{q}}^{i,j}$)	1.119	1.268	0.500	0.542	0.475	0.524

3) Synchronous & Spreading Motions Comparison

Comparison of simulation results for both synchronous and spreading targetS motions is summarized in TABLE 1. In this table, RMS$\left(\tilde{\mathbf{q}}^{i,j}\right)$ and RMS$\left(\mathbf{e}_{\mathrm{p}}^i\right)$ denote the RMS values of the DEKF estimation error and the tracking control error, respectively, at simulation times $50 < t < 400$. The Event (%) denotes the percentage of the overall simulation time during which communication activation events between tracker UAVs were triggered. It can be seen for all scenarios in this table that the increase in the number of trackers improves the performance of both the estimation and pursuit control of the trackers, while at the same time it reduces the number/frequency of communication activation between trackers. In particular, the estimation errors of the spreading targets case are higher than those of synchronous targets case. This is indeed as expected due to the used simulation scenario, whereby the strength of measurement noise is set to be proportional to the distance between trackers and targets. In addition, such a setup is also reflected in the achieved pursuit errors, whereby the values for the spreading targets case are larger than those of the synchronous targets case as a result of wider targets' trajectories. Despite having higher estimation errors, the spreading targets case requires less communication activation compared to the synchronous targets case.

B. MEDIUM-SCALE TRACKER-TARGET MAS SIMULATION

A medium scale simulation scenario is also conducted considering four trackers and five targets which move with different motion patterns. Two of the trackers follow a similar trajectory, while the others move in distinct patterns: one along a diagonal path, one on a straight line path along the X-axis, and one on a circular path. The performance of both cooperative localization and pursuit are then examined.

The proposed ET-DEKF method demonstrates good target localization performance and achieves accurate position estimation for different tracking scenarios. FIGURE 27 shows the position estimation errors of the five target vessels with an RMS$\left(\tilde{\mathbf{q}}^{i,j}\right)$ value of 0.576 m. Such a higher RMS value when compared to small-scale simulation cases is primarily caused by the fifth target which follows a circular motion path. Specifically, while the target motion accounts for the angular speed dynamics, the estimation accuracy for circular trajectories remains slightly less precise than those of linear and curve motions. In addition, the event-triggering plot in FIGURE 28 which shows that communication between agents was activated only up to 19.90% of the total simulation time clearly demonstrates the improvement in computational efficiency of
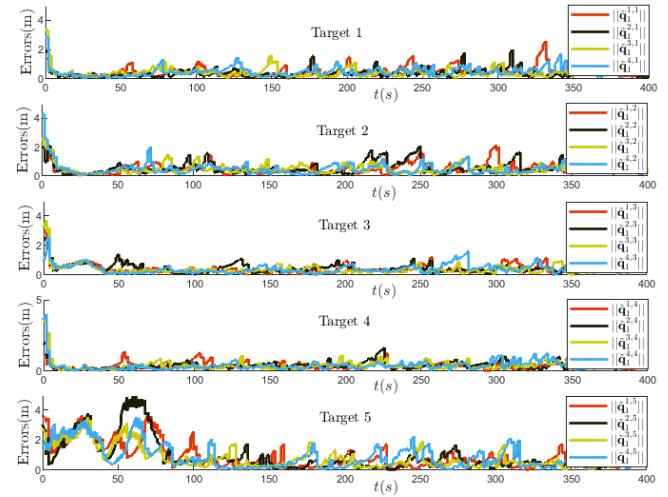
FIGURE 27: Estimation errors (medium-scale scenario).
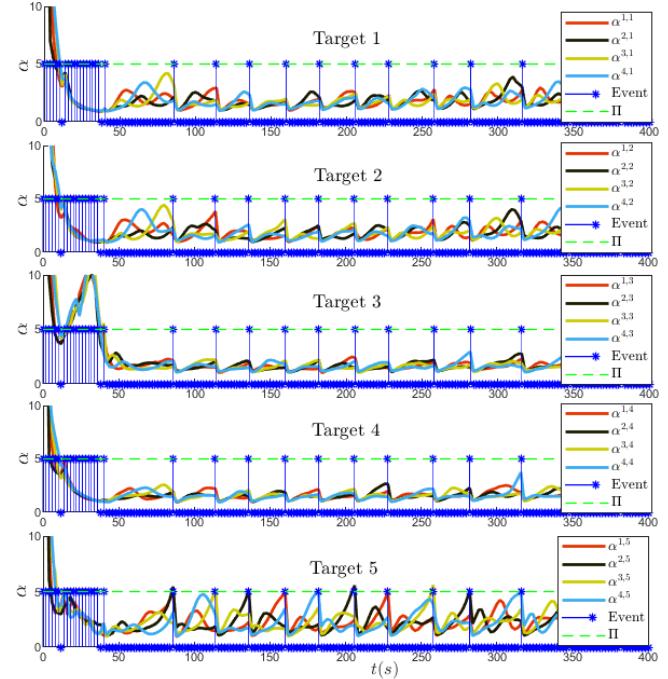
FIGURE 28: ET activation events (medium-scale scenario).

the proposed ET-DEKF method. The high frequency of events at the beginning of the simulation was due to a significant initial estimation error of the third target. As the estimation errors decay, the events of communication activation decrease and are mainly activated to track the motion of the fifth target. These results highlight the effectiveness of the proposed ET-DEKF method in performing the adaptive localization task while maintaining efficient resource utilization.

Using ET-DEKF estimation of the targets' positions, the trackers implement the pursuit and tracking control methods. The pursuit errors shown in FIGURE 29 demonstrate a gradual convergence toward zero with RMS value RMS$\left(\tilde{\mathbf{q}}^{i,j}\right)$ of 0.552 m, which is consistent with the small-scale results. The S-T curves of the trackers in FIGURE 30 show that the trajectories remain bounded within the specified radius
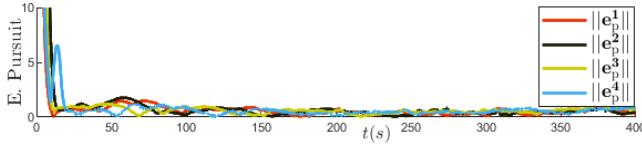
FIGURE 29: Pursuit errors (medium-scale scenario).
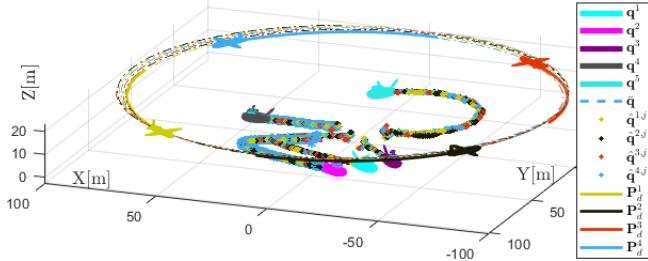
FIGURE 30: S-T curves illustration (medium-scale scenario).
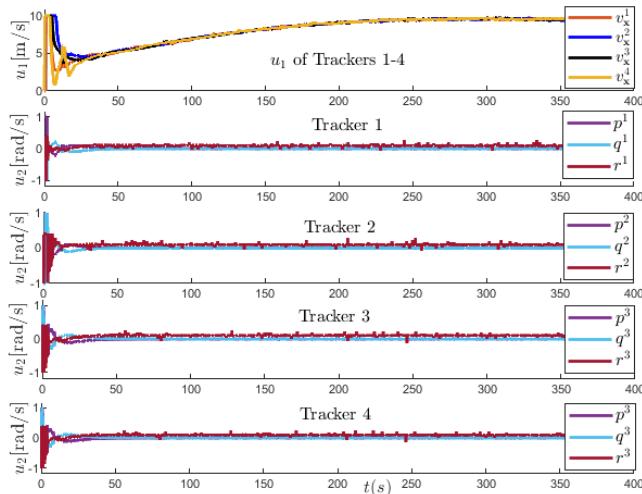
FIGURE 31: Pursuit control inputs (medium-scale scenario).

of the cycloid curve and closely follow the centroid of the target cluster. The corresponding control inputs shown in FIGURE 31. suggests that trackers input remains consistent in all cases while nearly reaches the maximum limit value, indicating further expansion of the inter-target radius is not feasible. If the radius increases, the trackers may not be able to maintain the desired trajectory and potentially compromise the tracking performance. In general, these simulation results further emphasize the effectiveness of the proposed tracker-target cooperative localization and pursuit methods.

C. LARGER-SCALE TRACKER-TARGET MAS SIMULATION

This section evaluates a larger-scale tracker-target MAS simulation to examine the computational complexity and scalability of the proposed cooperative localization and pursuit methods. In particular, the simulation scenario consists of the combination of up to $N = 9$ trackers and $M = 10$ targets that move in synchronous motion. The computational complexity and scalability are evaluated for different combinations of tracker-target pairs, such as $(N = 1, M = 1)$, $(N = 1, M = 2)$, $(N = 2, M = 3)$, etc. Performance comparisons are also

presented between the method proposed in this paper with those of [24] (Method 2) and [25] (Method 3). A summary of Method 2 and Method 3 is briefly discussed in the Appendix.

1) Evaluation of Scalability

The performance of the cooperative localization method is evaluated based on the RMS values achieved on the estimation error of the target states defined as $\mathbf{q}_1^j - \hat{\mathbf{q}}_1^{i,j}$. This evaluation metric is denoted with RMS$(\tilde{\mathbf{q}}^{i,j})$-$M$ with respect to the number M of the targets as illustrated in FIGURE 32a. As can be seen in this figure, all the methods demonstrate accurate and good performance of cooperative localization results for large numbers of trackers and targets combinations and thereby highlight their scalability and effectiveness for larger-scale simulation scenarios. Notice for $M \leq 2$ that all methods, in particular Method 2, exhibit significantly higher estimation errors, which can essentially be attributed to the lower accuracy of a single tracker due to the absence of data fusion process. In addition, the performance of the pursuit control method is evaluated using the RMS value $\mathbf{e}_p$ of the pursuit errors. This evaluation metric is denoted with RMS$(\mathbf{e}_p)$-M with respect to the number M of the targets as shown in FIGURE 32b. It can be seen in this figure that the method proposed in this paper and Method 2 achieve more accurate tracking result with an average RMS$(\mathbf{e}_p)$ values of 0.423 and 0.527, respectively, while Method 3 results in a higher average RMS$(\mathbf{e}_p)$ of 1.749. These results clearly demonstrate the better performance of the proposed cooperative localization and pursuit control methods compared to the methods proposed in [24] and [25].

2) Evaluation of Computational Complexity

The computational complexity of the cooperative localization method is evaluated based on the average time $\bar{t}_l$ required by each tracker to estimate the state variables of all targets as the number of such targets increases. As shown in FIGURE 33a, the computation time required to perform the proposed ET-DEKF method scales linearly with the increasing number of targets. Note in this figure that all methods exhibit an approximately similar trend of computational time, with each tracker requiring 0.928 ms on average to successfully localize

(a) RMS$(\tilde{\mathbf{q}}^{i,j})$

(b) RMS$(\mathbf{e}_p^i)$
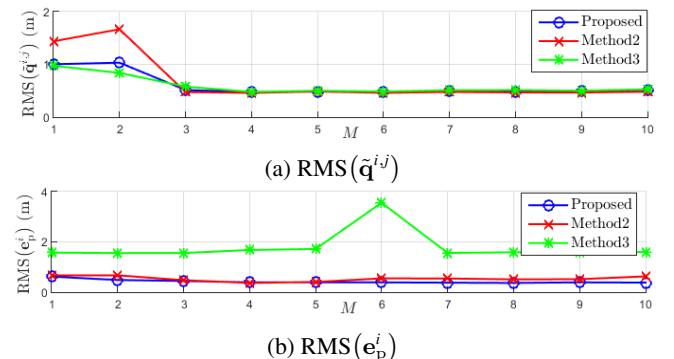
FIGURE 32: Comparison of scalability performance.

(a) $\bar{t}_l$

(b) $\bar{t}_p$
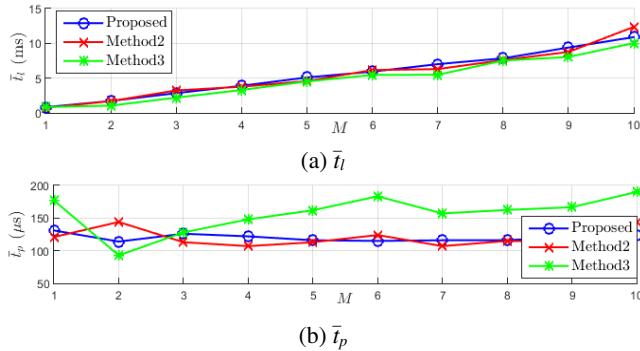
FIGURE 33: Comparison of average computational time.
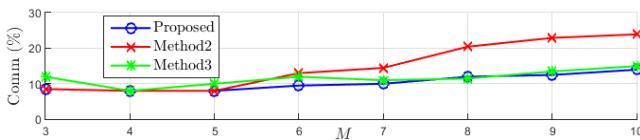
FIGURE 34: Comparison of ET activation.

each target. In addition, the computational complexity of the cooperative pursuit control method is evaluated based on the average time $\bar{t}_p$-M required by each tracker to generate the control input to follow the reference S-T curve. As shown in FIGURE 33b, the computation time t_p to generate the pursuit control inputs remain approximately constant wrt the increase in the number of targets, with an average value of $120\mu s$ for the method proposed in this paper and Method 2, and a slightly higher value of $292\mu s$ for Method 3. In particular, the communication event plots shown in FIGURE 34 indicate the computational advantage offered by the ET strategy proposed in this paper which requires fewer communication update events than other methods. This essentially results in the reduction of communication overhead and preserves computational resources. However, one may observe that the frequency of ET updates also increases with increasing number of targets. Future works may be directed toward exploring more localized and adaptive ET mechanisms that can balance communication demand/allocation in large-scale scenarios of target-tracker MAS interaction.

V. CONCLUDING REMARK

This paper has presented a multi-UAV coordination scheme for position estimation and tracking control purposes of multiple moving target vessels to improve maritime surveillance and tracking of illegal fishing activities. The proposed scheme uses range-based event-triggered distributed extended Kalman filter (DEKF) to estimate the targets' state variables. The integration of spatial-temporal (S-T) curves and adaptive geometric formations ensures the target remains within the sensor's measurement range at all times. The proposed approach thus has the potential to reduce the need for extensive human resources while providing real-time tracking capabilities to address illegal fishing activities.

REFERENCES

[1] N. Cahyani et al., "Inventorizing marine biodiversity using eDNA from Indonesian coral reefs: comparative throughput analysis using different bioinformatic pipelines," Mar. Biodivers., vol. 54, no. 3, p. 39, 2024.

[2] M. Smithrithee, Fishery Statistical Bulletin of Southeast Asia 2020. Southeast Asian Fisheries Development Center, Jan. 2023.

[3] A. Kusdiantoro, Fahrudin, S. Wisudo, and B. Juanda, "Sustainable development index of marine capture fisheries in indonesia," Int. J. Sustain. Dev. Plan., vol. 15, no. 8, pp. 1267–1275, 2020.

[4] Yulia, S. Zubaidah, and H. Ainun, "Illegal fishing by foreign vessels against fish resources in sulawesi sea waters, indonesia," IOP Conf. Ser. Earth Environ. Sci., vol. 860, no. 1, p. 012095, Oct 2021.

[5] I. Chapsos and S. Hamilton, "Illegal fishing and fisheries crime as a transnational organized crime in Indonesia," Trends. Organ. Crime, vol. 22, no. 3, pp. 255–273, 2019.

[6] I. Chapsos, J. Koning, and M. Noortmann, "Involving local fishing communities in policy making: Addressing illegal fishing in Indonesia," Mar. Policy, vol. 109, p. 103708, 2019.

[7] C. M. Clark et al., "Tracking and following a tagged leopard shark with an autonomous underwater vehicle," J. Field Robot., vol. 30, no. 3, pp. 309–322, 2013.

[8] A. Zolich, T. A. Johansen, J. A. Alfredsen, J. Kuttenkeuler, and E. Erstorp, "A formation of unmanned vehicles for tracking of an acoustic fish-tag," in Proc. OCEANS 2017-Anchorage, 2017, pp. 1–6.

[9] M. S. Arifin et al., "Experimental modeling of a quadrotor uav using an indoor local positioning system," in Proc. 5th Int. Conf. Electric Vehicular Technology, 2018, pp. 25–30.

[10] N. Crasta, D. Moreno-Salinas, B. Bayat, A. M. Pascoal, and J. Aranda, "Range-based underwater target localization using an autonomous surface vehicle: Observability analysis," in Proc. IEEE/ION Position, Location and Navigation Symposium (PLANS), 2018, pp. 487–496.

[11] I. Masmitja et al., "Optimal path shape for range-only underwater target localization using a wave glider," Int. J. Robot. Res., vol. 37, no. 12, pp. 1447–1462, 2018.

[12] T.-M. Nguyen, Z. Qiu, M. Cao, T. H. Nguyen, and L. Xie, "An integrated localization-navigation scheme for distance-based docking of UAVs," in Proc. IEEE/RSJ IROS, 2018, pp. 5245–5250.

[13] N. T. Hung, F. F. C. Rego, and A. M. Pascoal, "Cooperative distributed estimation and control of multiple autonomous vehicles for range-based underwater target localization and pursuit," IEEE Trans. Control Syst. Technol., vol. 30, no. 4, pp. 1433–1447, 2022.

[14] G. Bonaventura and T. A. Tamba, "Multi quadrotors coverage optimization using reinforcement learning with negotiation," IAES Int. J. Artif. Intell., vol. 13, no. 3, pp. 2978–2986, 2024.

[15] Y. Y. Nazaruddin et al., "Communication-efficient optimal-based control of a quadrotor uav by event-triggered mechanism," in Proc. 5th Asian Conf. Defense Technol., 2018, pp. 96–101.

[16] N. Crasta, D. Moreno-Salinas, A. M. Pascoal, and J. Aranda, "Multiple autonomous surface vehicle motion planning for cooperative range-based underwater target localization," Annu. Rev. Control, vol. 46, pp. 326–342, 2018.

[17] T. Oliveira, A. P. Aguiar, and P. Encarnação, "Moving path following for unmanned aerial vehicles with applications to single and multiple target tracking problems," IEEE Trans. Robot., vol. 32, no. 5, pp. 1062–1078, 2016.

[18] G. Battistelli and L. Chisci, "Stability of consensus extended Kalman filter for distributed state estimation," Automatica, vol. 68, pp. 169–178, 2016.

[19] W. Li, S. Zhou, M. Shi, J. Yue, B. Lin, and K. Qin, "Collision avoidance time-varying group formation tracking control for multi-agent systems," Appl. Intell., vol. 55, no. 3, p. 175, Dec. 2024.

[20] T. A. Tamba, B. C. G. Cinun, Y. Y. Nazaruddin, M. Z. Romdlony, and B. Hu, "Event-triggered robust formation control of multi quadrotors for transmission line inspection," J. Mechatron. Electr. Power Veh. Technol., vol. 15, no. 2, pp. 197–207, 2024.

[21] A. Alanwar, H. Said, A. Mehta, and M. Althoff, "Event-triggered diffusion Kalman filters," in Proc. ACM/IEEE 11th ICCPS, 2020, pp. 206–215.

[22] T. Oliveira, A. P. Aguiar, and P. M. M. Encarnação, "Moving path following for unmanned aerial vehicles with applications to single and multiple target tracking problems," *IEEE Trans. Robot.*, vol. 32, pp. 1062–1078, 2016.

[23] N. T. Hung, N. Crasta, D. Moreno-Salinas, A. M. Pascoal, and T. A. Johansen, "Range-based target localization and pursuit with autonomous vehicles: An approach using posterior CRLB and model predictive control," *Robot. Auton. Syst.*, vol. 132, p. 103608, 2020.

[24] M. F. Reis, R. P. Jain, A. P. Aguiar, and J. B. de Sousa, "Robust moving path following control for robotic vehicles: Theory and experiments," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3192–3199, 2019.

[25] L. Lapierre, D. Soetanto, and A. Pascoal, "Nonlinear path following with applications to the control of autonomous underwater vehicles," in *Proc. 42nd IEEE CDC*, vol. 2. IEEE, 2003, pp. 1256–1261.

APPENDIX. SUMMARY OF METHODS FOR COMPARISON

A. SUMMARY AND RESULTS OF METHOD 2

This section describes Method 2 proposed in [24] that is used as the first comparison for the method proposed in this paper. In essence, the cooperative pursuit scheme developed in [24] chooses to consider the second-order dynamics of variable $\ddot{\gamma}^i$ (rather than $\dot{\gamma}^i$ as considered in this paper) defined as follows.

$$\ddot{\gamma}^i = -k_\gamma e^i_\gamma + \left(\hat{\mathbf{e}}^i_{\mathrm{p}}\right)^\top R^\top \left(\mathbf{p}^i_2\right) \mathbf{s}'\left(\gamma^i\right) + \dot{\omega}^i_c, \quad (29)$$

where k_γ is a constant to design, and $\dot{\gamma}$ is defined as below.

$$e^i_\gamma = \dot{\gamma}^i - \omega^i_{\mathrm{d}}. \quad (30)$$

To evaluate the performance of Method 2, simulations similar with those described in Section IV-A were performed using a synchronous target motion consisting of three trackers. The simulation results using Method 2 generally similar to those obtained using the methods proposed in this paper, but with some differences. FIGURE 35 shows the communication activation events when three trackers are deployed, while FIGURE 36 shows the corresponding tracker inputs. It can be seen that each update triggered by an event introduces fluctuations in the system response because the event modifies the correction speed ω^i_c in (22). This abrupt change causes rapid variations in the system response and triggers spikes in $\dot{\omega}^i_c$, which subsequently affect the dynamics of $\ddot{\gamma}^i$ in (29). The occurrences of such spikes in the control input are undesirable as they may introduce instability or excessive control effort that potentially can degrade the overall system performance. In contrast, FIGURE 16 shows that the method proposed in this paper exhibits a significantly smoother control response.
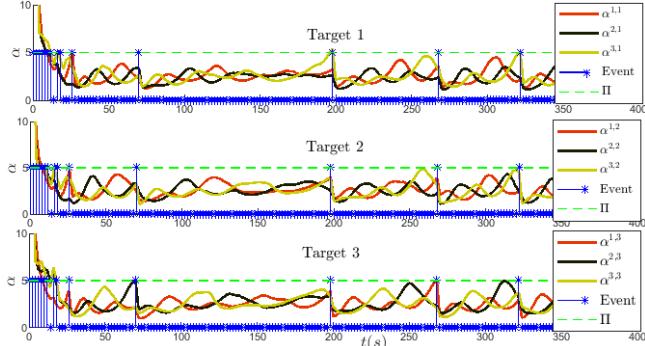
FIGURE 35: ET activation events for Method 2 [24].
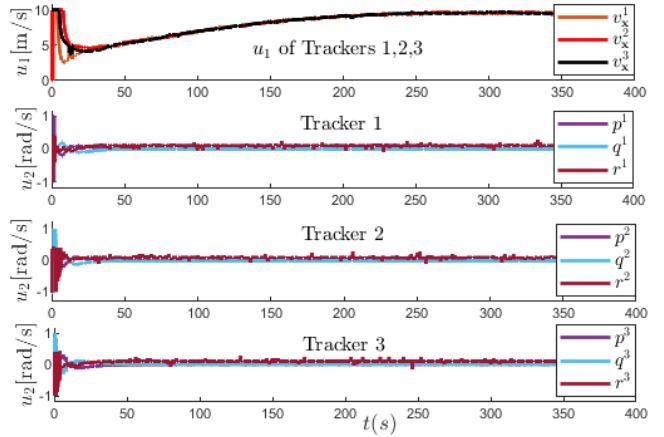
FIGURE 36: Example control inputs in Method 2.

The absence of sudden spikes essentially indicates that the proposed control action is more stable and reduces unnecessary actuation, leading to improved overall system efficiency.

B. SUMMARY AND RESULTS OF METHOD 3

The second approach used for the comparison is Method 3 developed in [25]. In this approach, the tracker input $\mathbf{u}^i = [v^i_p, \dot{\phi}^i_{\mathcal{B}}, \dot{\theta}^i_{\mathcal{B}}, \omega^i_{\mathcal{B}}]^\top$ described in (25) is modified into a two-dimensional control input that consists of $\dot{\phi}^i_{\mathcal{B}} = 0$ and $\dot{\theta}^i_{\mathcal{B}} = 0$ as illustrated in FIGURE 37. In this regard, the pursuit error $\mathbf{e}^i_\varphi$ is defined in the S-T curve wrt frame $\mathcal{F}^i$ as the form below.

$$\mathbf{e}^i_\varphi = R^\top\left(\psi^i_\varphi\right)\left(\mathbf{p}^i_1 - \mathbf{P}^i_d\right). \quad (31)$$

where ψ^i_φ is the orientation of the S-T curve on frame $\{\mathcal{I}\}$. The updated control law is then defined as follows.

$$v^i_p = k_1 v^i_\varphi + k_2 \mathbf{e}^i_{\varphi,\mathrm{t}}, \quad (32)$$

where k_1, k_2 are controller parameters and $v^i_\varphi = ||\dot{\mathbf{P}}^i_d||$ is the S-T curve speed. The yaw rate control input $\dot{\psi}^i_{\mathcal{B}}$ is defined as

$$\omega^i_{\mathcal{B}} = \kappa(\gamma^i)v^i_\varphi + \dot{\delta}^i - k_3\tilde{\psi}^i - k_4\mathbf{e}^i_{\varphi,\mathrm{n}}v^i_p\frac{\sin(\psi^i_e) - \sin(\delta^i)}{\tilde{\psi}^i}. \quad (33)$$

where $\kappa(\gamma^i)$ is a curvature function defined as follows.

$$\kappa(\gamma^i) = \frac{\mathbf{P}^{i\prime}_{d,\mathrm{x}}(\gamma^i)\,\mathbf{P}^{i\prime\prime}_{d,\mathrm{y}}(\gamma^i) + \mathbf{P}^{i\prime\prime}_{d,\mathrm{x}}(\gamma^i)\,\mathbf{P}^{i\prime}_{d,\mathrm{y}}(\gamma^i)}{||\mathbf{P}^{i\prime}_d(\gamma^i)||^3}. \quad (34)$$
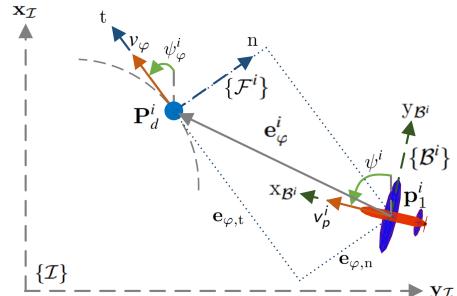
FIGURE 37: Illustration of S-T curve pursuit in Method 3.

where subscripts x and y denote values on the x and y axes, respectively, of the frame $\{\mathcal{I}\}$. The velocity v_φ^i is defined as

$$v_\varphi^i = \frac{v_p^i \cos(\psi_e^i)}{1 - \kappa(\gamma^i)\mathbf{e}_{\varphi,\mathrm{n}}^i}, \qquad (35)$$

with $\psi_e^i = \psi^i - \psi_\varphi^i$ denotes the error of the heading angle. The orientation compensation parameter δ^i is defined as

$$\delta^i = -\theta \tanh(k_5 \mathbf{e}_{\varphi,\mathrm{x}}^i). \qquad (36)$$

while the orientation error values $\tilde{\psi}^i$ is formulated as

$$\tilde{\psi}^i = \psi_e^i - \delta^i. \qquad (37)$$

Similarly with Method 2, the performance of Method 3 is evaluated using simulations based on synchronous target motions with three trackers, as described in Section IV-A. FIGURE 38 shows the pursuit errors achieved using Method 3 with control inputs as depicted in FIGURE 39. It can be seen that the pursuit errors remain bounded and the corresponding control inputs remain reasonably smooth. However, compared to the method proposed in this paper, Method 3 exhibits slower response times and produces noticeable state oscillations after reaching the steady state value, and eventually causes higher cooperative pursuit error values.
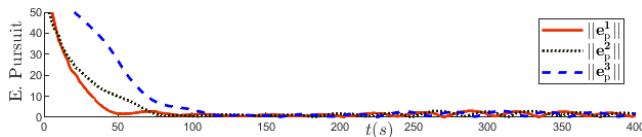

FIGURE 38: Illustration of pursuit errors (Method 3).
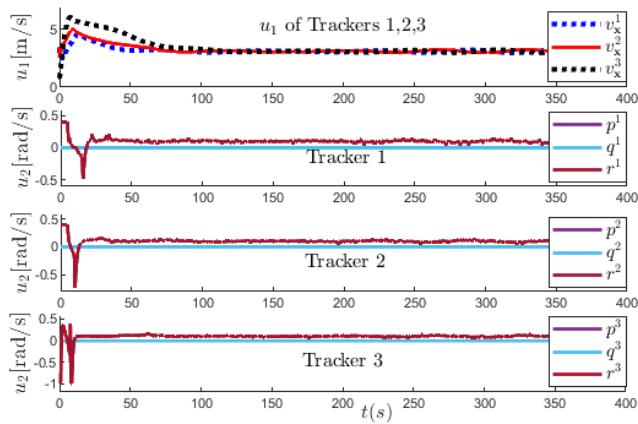
FIGURE 39: Illustration of control inputs (Method 3).

TUA AGUSTINUS TAMBA (Member, IEEE) received both MSEE and PhD degrees in electrical engineering from the University of Notre Dame, USA, in 2016, the M.Sc. degree in mechanical engineering from Pusan National University, Republic of Korea, in 2009, and the B.Eng. degree in engineering physics from Institut Teknologi Bandung, Indonesia, in 2006. He is currently an associate professor in the Department of Electrical Engineering at Parahyangan Catholic University, Indonesia. His main research interests include dynamical systems, control theory, and optimization with applications in mechatronics, robotics, automation, and networked dynamical systems.

IMMANUEL R. SANTJOKO is currently pursuing his B.Eng. degree in electrical engineering at Parahyangan Catholic University, Indonesia. His research interests include robotics, systems identification, and industrial process control systems.

YUL Y. NAZARUDDIN (Member, IEEE) received the BEng degree in engineering physics from Institut Teknologi Bandung, Indonesia, in 1982, the M.Sc. and D.I.C degrees from Imperial College of Science and Technology, University of London, UK, in 1985, and Dr. Ing. degree in electrical engineering from Ruhr University Bochum, Germany, in 1994. He is currently a Professor in the Faculty of Industrial Engineering at Institut Teknologi Bandung, Indonesia. His research interest include theory and applications of advanced instrumenation and control techniques for industrial control and automation, robotics, mechatronics, and intelligent systems.

VEBI NADHIRA (Member, IEEE) received the B.Eng. degree in engineering physics, M.Eng. degree in instrumentation and control, and Doctoral degree in engineering physics from Institut Teknologi Bandung, Indonesia. She is currently an assistant professor in Faculty of Industrial Engineering at Institut Teknologi Bandung, Indonesia.

• • •