

Multi-Agent Coordinated Interception of Multiple Rogue Drones

Panayiota Valianti, Savvas Papaioannou, Panayiotis Kolios, and Georgios Ellinas

Abstract—Over the last few years there has been an unprecedented interest in unmanned aerial vehicles (UAVs). However, drones potentially pose great threats to security and public safety, especially when their malicious use involves critical infrastructures and public spaces. This work proposes a multi-agent counter-drone system where a team of pursuer drones cooperate in order to track and jam multiple rogue drones. Specifically, a cooperative multi-agent approach is proposed in which the best joint mobility and power control actions of each agent are chosen so that the rogue drones are optimally tracked and jammed over time. Two variants of the joint optimization problem are developed and extensive simulations are conducted so as to evaluate the performance of the proposed approach.

I. INTRODUCTION

The usage of drones has grown significantly in size and importance in recent years. Indeed, drones nowadays are extremely appealing with many applications including aerial surveillance, motion picture filmmaking, mineral exploration, and remote sensing just to name few.

Drones, however, have also shown to pose a significant challenge to public security and personal privacy [1]. Recently, the study of anti-drone systems has attracted a considerable amount of interest [2], [3]. This work investigates the scenario where a team of autonomous mobile agents (i.e., drones) cooperate in order to track and jam multiple rogue drones (targets). It is assumed that the rogue drones exhibit stochastic dynamics and thus their trajectory over time needs to be estimated from noisy sensor measurements. In addition, it is assumed that the mobile agents exhibit a limited sensing range and that they can detect the presence of the rogue drones inside their sensing range with probability less than one. Finally, due to sensing imperfections, it is further assumed in this work that in addition to the target measurements the mobile agents receive false-alarm measurements as well. Fig. 1 provides an illustrative representation of the proposed problem setup.

The main objectives of the agents are to cooperate in order to: (a) accurately track the rogue drones over a period of time, and (b) to effectively jam and disable the rogue drones in the air by transmitting power from their on-board antennas, while at the same time ensuring that the received power between the cooperating agents is kept below interfering levels. The contributions of this work are the following:

The authors are with the Department of Electrical and Computer Engineering and the KIOS Research and Innovation Center of Excellence (KIOS CoE), University of Cyprus, Nicosia, 1678, Cyprus. E-mail: {valianti.panayiota, papaioannou.savvas, pkolios, gellinas}@ucy.ac.cy

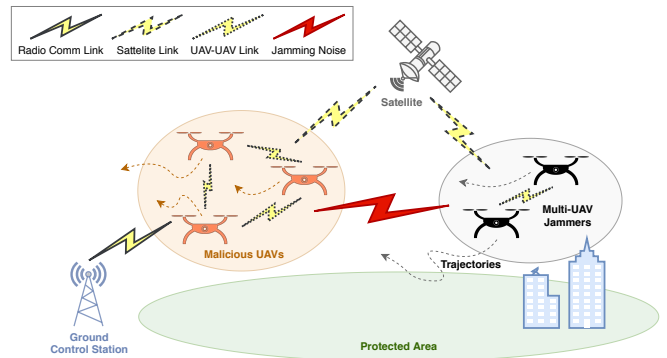


Fig. 1: Multiple agents tracking and jamming multiple targets.

- 1) A novel multi-agent framework is proposed for the joint tracking and jamming problem which realistically captures all the important characteristics of this problem including the agent limited sensing range, the detection uncertainty, the existence of false alarms, and the uncertain target trajectory.
- 2) A novel mathematical programming model is derived to formulate the joint tracking and jamming problem which is then solved using a Mixed-Integer Linear Program solver as part of an iterative decision and control algorithm.
- 3) Extensive simulation experiments are conducted that demonstrate the performance of the proposed framework.

The rest of the paper is organized as follows. Section II describes the related work, while Section III provides a brief overview on stochastic filtering. Thereafter, Section IV presents the models that describe the system dynamics, while Section V formulates the joint tracking and jamming problem. The evaluation of the proposed framework is presented in Section VI where an extensive set of simulation results are presented, followed by concluding remarks in Section VII.

II. RELATED WORK

Existing literature on the problem of preventing rogue drones from entering a secure area includes the work in [4], where a low-cost terrestrial tracking and jamming solution is proposed. More recent works utilize a swarm of pursuer drones, where one common approach is to formulate the interception problem as a pursuit-evasion problem, in which multiple pursuers are chasing evader(s) in order to capture

them. In [5], [6], a number of pursuer drones in a 2-D environment strategically attempt to minimize the area of a single rogue drone (evader), which guarantees the capture of the evader, that is assumed, when at least one of the pursuers reaches the evader within a specific distance. More recently, the authors in [7] considered the same problem and extended it to multiple evaders. Further, the work in [8] investigates a more complicated scenario in which the pursuer drones are prohibited from entering no-fly zones. When the evader enters a no-fly zone, the pursuers get distributed on the boundary of the zone to immediately capture the evader once it exits the no-fly zone. Another recent solution distributes the swarm of pursuer drones in a formation around the rogue drone in order to decrease its movement and guide it outside the secured area [9]. In that work, the authors assume that the rogue drone attempts to avoid collisions and that a high-quality tracking system is accessible that is able to detect and track the rogue drone.

Complementary to the aforementioned studies, this work investigates the use of a swarm of pursuer drones that aim to intercept multiple rogue drones using signal jamming. The novelty of the proposed approach lies in the joint mobility and power control solution presented for accurately tracking the moving drones, while at the same time maximizing the power received at the targets, so as to jam their communication and sensing circuitry and prevent them from completing their mission.

III. BACKGROUND ON STOCHASTIC FILTERING

The aim of Bayesian filtering [10] is to recursively estimate the conditional posterior distribution, i.e. $p(x_t|y_{1:t})$ of the target state x_t at time t given all the measurements y_1, y_2, \dots, y_t up to time t . Given an initial density on the state $p(x_0)$, the posterior density at time t can be computed using Bayes recursion (also known as the Bayes filter) as:

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} \quad (1)$$

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{\int p(y_t|x_t)p(x_t|y_{1:t-1})dx_t} \quad (2)$$

where Eqs. (1) and (2) are referred to as the prediction and update steps, respectively, and functions $p(x_t|x_{t-1})$ and $p(y_t|x_t)$ are the known transitional density and likelihood functions, respectively. At each time step, the hidden state x_t is usually extracted from the posterior distribution using the expected a posteriori (EAP) or the maximum a posteriori (MAP) estimators.

In this work, the above recursion is used in order to estimate the states of the rogue drones (i.e., targets). It is assumed that the number of targets is fixed and known and that the assignment of measurements to targets is also known (i.e., the data association problem is solved). In essence, each agent j maintains a separate density $p_t^{ji}(\cdot)$ for each target i and at each time-step the agents exchange their estimates to obtain the final target state through covariance intersection [11].

IV. SYSTEM MODEL

This section elaborates on the system model for the proposed approach.

A. Target Dynamics

Let the state of a single target have the following form [12]:

$$x_t \sim p(x_t|x_{t-1}) \quad (3)$$

$$y_t \sim p(y_t|x_t, s_t) \quad (4)$$

where $x_t \in \mathbb{R}^{n_x}$ is the target state at time t (i.e., hidden random variable), $y_t \in \mathbb{R}^{n_y}$ is the target measurement, and $s_t \in \mathbb{R}^{n_s}$ is the state of the agent. The target states follow a first order Markov process defined in terms of the transitional density $p(x_t|x_{t-1})$ and the dependence of the measurement y_t on the target, and the agent state is modeled by the conditional density $p(y_t|x_t, s_t)$, i.e., the likelihood function. More specifically, the target state evolves in time according to the following discrete time dynamics:

$$x_t = \zeta(x_{t-1}) + \nu_t \quad (5)$$

where the non-linear function $\zeta : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_x}$ models the dynamical behavior of the target and the process noise $\nu_t \in \mathbb{R}^{n_x}$ is normally distributed and models the random disturbances in the state evolution. The measurement equation is given by:

$$y_t = h(x_t, s_t) + w_t \quad (6)$$

where the non-linear function $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \rightarrow \mathbb{R}^{n_y}$ defines the relationship between the states (x_t, s_t) and the measurement y_t . The measurement noise w_t is normally distributed and independent of ν_t . Without loss of generality, in this work it is assumed that the target state vector $x_t \in \mathbb{R}^5$ is defined as $x = [x, \dot{x}, y, \dot{y}, \omega]^\top$ where (x, y) give the 2D position of the target in Cartesian coordinates, (\dot{x}, \dot{y}) are the velocities of the target in the x and y direction respectively, and ω is the turn rate. On the other hand, the measurement vector $y_t \in \mathbb{R}^2$ is composed of noisy heading and range observations.

B. Agent Dynamics

Let $S = \{1, 2, \dots, |S|\}$ be the set of all available controllable mobile agents (pursuers). Each agent $j \in S$ is subject to the following discrete-time dynamics:

$$s_t^j = s_{t-1}^j + \begin{bmatrix} l_1 \Delta_R \cos(l_2 \Delta_\theta) \\ l_1 \Delta_R \sin(l_2 \Delta_\theta) \end{bmatrix}, \quad \begin{matrix} l_1 = 0, \dots, N_R \\ l_2 = 0, \dots, N_\theta \end{matrix} \quad (7)$$

where $s_{t-1}^j = [s_x^j, s_y^j]^\top \in \mathbb{R}^2$ denotes the position of agent j (i.e., xy-coordinates) at time $t-1$, Δ_R is the radial step size, $\Delta_\theta = 2\pi/N_\theta$, and the parameters (N_θ, N_R) control the number of possible control actions. $\mathbb{U}_t^j = \{s_t^{j,1}, s_t^{j,2}, \dots, s_t^{j,|\mathbb{U}_t^j|}\}$ as computed by Eq. (7) is denoted as the set of all admissible control actions of agent j at time t .

C. Agent Sensing Model

The agent exhibits a limited sensing range for detecting nearby targets which is modeled by the function $p_D(x_t, s_t)$. This function gives the probability that a target with state x_t at time t is detected by an agent with state $s_t = [s_x, s_y]^\top \in R^2$. More specifically, the target with state x_t and 2D Cartesian coordinates $p_t = Hx_t$ (where H is a matrix that extracts the (x, y) coordinates of a target from its state vector) is detected by an agent with state s_t with probability which is given by:

$$p_D(x_t, s_t) = \begin{cases} p_D^{\max} & \text{if } d_t < R_0 \\ \max\{0, p_D^{\max} - \eta(d_t - R_0)\} & \text{if } d_t \geq R_0 \end{cases} \quad (8)$$

where $d_t = \|Hx_t - s_t\|_2$ denotes the Euclidean distance between the agent and the target, p_D^{\max} is the detection probability for targets that reside within distance R_0 from the agent's position, and parameter η captures the reduced effectiveness of the agent to detect distant targets.

Suppose now that there are exactly T targets inside the area of interest at each time step and that the targets are independent of each other. Each agent is equipped with a range finding sensor which consist of T sensing slots for detecting each one of the T targets. However, due to sensor imperfections at each sensing slot the agent also receives false-alarm measurements c_t^1, \dots, c_t^n which are being received with an average rate of λ and which are distributed according to the density function $p_c(c)$. In essence, at each time step and at each sensing slot each agent $j \in S$ receives the following measurement set:

$$\Upsilon_t^{ji} = \bigcup \left\{ a \subset \{\emptyset, y_t^{ji}\}, b \subseteq \{c_t^1, \dots, c_t^n\} \right\}, \forall i \in [1, \dots, T] \quad (9)$$

where y_t^{ji} is the the measurement from the i^{th} target which has been received by agent j at sensing slot i .

D. Path-Loss Model

Since drone-to-drone wireless channels are generally line-of-sight (LOS) dominated, the free space path-loss model (with path-loss exponent 2) is adopted in this work, as follows:

$$\Lambda(\text{dB}) = 20\log_{10}(d_m) + 20\log_{10}(f_{\text{GHz}}) + 32.45 - G_t - G_r \quad (10)$$

where d_m is the distance between transmit and receive antennas in meters, f_{GHz} is the carrier frequency in GHz and G_t and G_r are the gains of the transmit and receive antennas, respectively.

V. COORDINATED TRACKING AND WIRELESS JAMMING

A. Rogue Drone State Estimation

To estimate the state of the rogue drone, stochastic filtering is used as described in Section III by applying Eqs. (1-2) recursively over time. In order to do that the rogue drone transitional density is needed, which can be obtained directly from Eq. (5) and the measurement likelihood function i.e., $p(\Upsilon_t^{ji}|x_t^i, s_t^j)$ which needs to be computed according to the measurement model of Eq. (9). As already discussed in

Section IV-C, the agents receive in each sensing slot, not only target measurements but false-alarms as well and so the likelihood function for each sensing slot i.e., $p(\Upsilon_t^{ji}|x_t^i, s_t^j)$ can be computed as:

$$p(\Upsilon_t^{ji}|x_t^i, s_t^j) = \left[1 - p_D(x_t^i, s_t^j)\right] e^{-\lambda} \prod_{\ell \in \Upsilon_t^{ji}} \lambda p_c(\ell) + e^{-\lambda} p_D(x_t^i, s_t^j) \sum_{\ell \in \Upsilon_t^{ji}} p(\ell|x_t^i, s_t^j) \prod_{\substack{\varepsilon \in \Upsilon_t^{ji} \\ \varepsilon \neq \ell}} \lambda p_c(\varepsilon) \quad (11)$$

where $p(\ell|x_t^i, s_t^j)$ is the single target likelihood function according to Eq. (6). To obtain the above expression the fact that $\Upsilon_t^{ji} = \{c_t^1, \dots, c_t^{|\Upsilon_t^{ji}|}\}$ is first used if the UAV receives only false alarm measurements (i.e. no measurement from the roque drone) and then $p(\Upsilon_t^{ji}|x_t^i, s_t^j)$ is given by:

$$|\Upsilon_t^{ji}|! \left[1 - p_D(x_t^i, s_t^j)\right] \text{Pois}(|\Upsilon_t^{ji}|) \prod_{\ell \in \Upsilon_t^{ji}} p_c(\ell) \quad (12)$$

where the function $\text{Pois}(|\Upsilon_t^{ji}|) = \frac{\lambda^{|\Upsilon_t^{ji}|} e^{-\lambda}}{|\Upsilon_t^{ji}|!}$ is the Poisson distribution with rate parameter λ and gives the probability of obtaining exactly $|\Upsilon_t^{ji}|$ false alarm measurements at time t and $\prod_{\ell \in \Upsilon_t^{ji}} p_c(\ell)$ is the joint density of the false alarms assuming they are independent. The above expression is then adjusted to account for all possible permutations of the measurement sequence and for the fact that the target has not been detected, which explains the first line in Eq. (11).

The second line of Eq. (11) is due to the more general case in which the mobile agent j has detected target i in slot i and thus the measurement y_t^i has been received in addition to the false alarm measurements.

B. Cascaded Decision and Control Algorithm

This section describes how to actively control the movement of the agents (i.e., selecting their mobility control actions) in order to maintain tracking of the detected target. More specifically, the aim is to find the optimal control actions $u_t^j \in \mathbb{U}_t^j, \forall j \in S$ that must be taken at time step t by each agent j so that the state of the target is estimated as accurately as possible.

Observe that the tracking accuracy increases when the targets are being detected and thus target measurements are received. For this reason the probability of detection is used as the control strategy in order to guide the agents in locations where the probability of detecting targets is increased. Thus, denoting the control objective function for agent j as $\xi(x_t^{ji}, u_t^j)$, the optimization problem to solve now becomes:

$$u_t^{j*} = \arg \max_{u_t^j \in \mathbb{U}_t^j} \xi(x_t^{ji}, u_t^j) = \arg \max_{u_t^j \in \mathbb{U}_t^j} \prod_{i=1}^T p_D(x_t^{ji}, u_t^j) \quad (13)$$

where x_t^{ji} denotes the i^{th} target state at time t as observed from agent j . The goal is to find the control action u_t^{j*} which results in maximizing the probability of target detection. Since the

true target state x_t^{ji} is not available until action u_t^j is applied, it is approximated as:

$$x_t^{ji} \approx \tilde{x}_t^{ji} = \arg \max_x p_t^{ji}(x|\Upsilon_{1:t-1}) \quad (14)$$

In essence, the predictive density $p_t^{ji}(x|\Upsilon_{1:t-1})$ (i.e., Eq. (1)) is used and the most likely predicted target state for time t is extracted. Then, Eq. (13) is optimized with the predicted target states. Assuming that the agents are independent of each other, the joint control objective is defined as:

$$u_t^{1*}, \dots, u_t^{|S|*} = \arg \max_{u_t^j, \forall j \in S} \prod_{i \in T, j \in S} \xi(\tilde{x}_t^{ji}, u_t^j) \quad (15)$$

where $u_t^j \in \mathbb{U}_t^j$ and \tilde{x}_t^{ji} is agent j 's prediction of target i 's state and is obtained from Eq. (14) for each agent via their predictive density as previously explained.

The solution of the problem presented in Eq. (15) provides the optimal mobility controls for all agents with respect to joint target tracking. However, these controls are not necessarily the best for the objective of downing multiple rogue drones. More specifically, in order to capture and disable rogue drones, the agents need to transmit a specific amount of power towards the targets, while at the same time minimize the power interference between them. In order to achieve the joint objective of tracking and jamming the targets, a cascaded control architecture is utilized, in which the optimal mobility control actions which result in a satisfactory tracking accuracy are first found, followed by a second optimization step where these mobility actions are further refined in order to achieve jamming control. To achieve that, instead of finding the mobility control actions which maximize Eq. (15), the set $\Omega = \{\omega_1, \dots, \omega_{|\Omega|}\}$ is first computed for all combinations of actions $u_t^j, \forall j \in S$, which satisfy:

$$\prod_{i \in T, j \in S} \xi(\tilde{x}_t^{ji}, u_t^j(\omega_k)) > \vartheta, k \in \Omega \quad (16)$$

where $\vartheta \in [0, 1]$ is the desired threshold of the joint probability of detection. In the second step, the optimal mobility control action and the level of transmit power for each of the cooperating agents are selected to maximize the received jamming power at the targets. This optimal control and decision problem is formulated according to (P1) below as an optimization problem.

$$(P1) \max_{P_k^j, b_k} \sum_{i=1}^{|T|} \sum_{k=1}^{|\Omega|} \sum_{j=1}^{|S|} P_k^j \Lambda(\tilde{x}_t^{ji}, u_t^j(\omega_k)) \quad (17)$$

$$\text{s.t.} \quad \sum_{i \in S, i \neq j} P_k^i \Lambda(u_t^i(\omega_k), u_t^j(\omega_k)) < \Delta^j, \forall j \in S, k \in \Omega \quad (18)$$

$$b_k |S| P_{max} \geq \sum_{j=1}^{|S|} P_k^j, \quad \forall k \in \Omega \quad (19)$$

$$\sum_{k=1}^{|\Omega|} b_k = 1 \quad (20)$$

$$0 \leq P_k^j \leq P_{max}, b_k \in \{0, 1\}, \quad \forall j \in S, k \in \Omega \quad (21)$$

(P1) is a mixed-integer linear problem where the best alternative mobility control action and the transmit power assigned for each agent are decided in order to maximize the total target received power, while ensuring that the interference level between cooperating agents is maintained below a threshold Δ , as described in constraint (18). Constraint (19) ensures that when a combination of control actions is selected, the total transmit power is curbed by the maximum transmit power P_{max} of all agents. The selection of a single combination of control actions is ensured through Eq. (20). The decision variables of optimization problem (P1) are defined in Eq. (21). However, (P1) does not guarantee that every target will receive satisfactory power; rather it maximizes the accumulated power. An alternatively approach is to maximize the minimum received power to the targets as expressed by optimization problem (P2), while ensuring that the interference and feasibility constraints expressed in (18)-(21) are again satisfied.

$$(P2) \max_{P_k^j, b_k} \min_{i \in T} \sum_{k=1}^{|\Omega|} \sum_{j=1}^{|S|} P_k^j \Lambda(\tilde{x}_t^{ji}, u_t^j(\omega_k)) \quad (22)$$

s.t. (18) – (21)

In either case, the resulting cascade control algorithm as presented in Alg. 1 computes the mobility and transmit power states to cooperatively intercept a set of rogue drones using the proposed joint tracking and jamming approach.

Algorithm 1 Cascaded Decision and Control

- Input:** $p^{ji}(x_{t-1}|\Upsilon_{1:t-1}), \forall j, \forall i, P_{max}$
- 1: Compute the predictive density $p^{ji}(x_t|\Upsilon_{1:t-1}), \forall j \in S$ using Eq. (1), for each target i .
 - 2: Compute an estimate of the target state for time t as $\tilde{x}_t^{ji} = \arg \max_x p^{ji}(x|\Upsilon_{1:t-1}), \forall j \in S$ (i.e., Eq. (14)), for each target i .
 - 3: Define set Ω of all combinations of control actions, which satisfy the tracking accuracy constraint ϑ using Eq. (16).
 - 4: Solve problem (P1) or (P2) for combinations Ω to obtain the optimal mobility and power controls for each agent.
 - 5: Execute the optimal mobility control actions and transmit power levels.
 - 6: Receive the measurement set $\Upsilon_t^{ji}, \forall j \in S$, for each target slot $i \in [1, \dots, T]$.
 - 7: Compute posterior density $p^{ji}(x_t|\Upsilon_{1:t}), \forall j \in S$ using Eq. (2), for each target i .
 - 8: Estimate target state \hat{x}_t^{ji} as $\arg \max_x p^{ji}(x|\Upsilon_{1:t})$.
 - 9: Exchange estimated target states and covariances and fuse using covariance intersection.
-

VI. PERFORMANCE EVALUATION

A. Simulation Setup

In the simulation setup it is assumed that the targets maneuver in an area of 100 m \times 100 m and that the single target state at time t is described by $x_t = [x, \dot{x}, y, \dot{y}, \omega]_t^\top$ i.e., position, velocity, and angular turn rate components. For the

target dynamics, i.e., Eq. (5), it is assumed that the target motion follows a coordinated turn (CT) model [13], [14] with variable angular turn rate and thus the target dynamics are given by $x_t = \zeta(x_{t-1}) + G\nu_t$ and the functions $\zeta(x)$ and $G\nu_t$ are given by:

$$\zeta(x) = \begin{bmatrix} x + \frac{\dot{x}}{\omega} \sin(\omega T) - \frac{\dot{y}}{\omega} (1 - \cos(\omega T)) \\ \dot{x} \cos(\omega T) - \dot{y} \sin(\omega T) \\ y + \frac{\dot{y}}{\omega} (1 - \cos(\omega T)) + \frac{\dot{x}}{\omega} \sin(\omega T) \\ \dot{x} \sin(\omega T) + \dot{y} \cos(\omega T) \\ \omega \end{bmatrix}$$

$$G\nu_t = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ T & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ 0 & T & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \nu_x \\ \nu_y \\ \nu_\omega \end{bmatrix}$$

where $T = 1s$ is the sampling interval, $\nu_x \sim \mathcal{N}(0, \sigma_x^2)$, $\nu_y \sim \mathcal{N}(0, \sigma_y^2)$, $\nu_\omega \sim \mathcal{N}(0, \sigma_\omega^2)$ with $\sigma_x = \sigma_y = 4 \text{ m/s}^2$ and $\sigma_\omega = \pi/180 \text{ rad/s}$. Once an agent detects a target it receives heading and range measurements, thus the measurement model of Eq. (6) is given by $h(x_t, s_t) = [\arctan(\frac{s_y - y}{s_x - x}), \|s_t - Hx_t\|_2]$ where H is a matrix that extracts the target position from its state vector. The target measurement likelihood function is then given by $p(y_t|x_t, s_t) = \mathcal{N}(y_t; h(x_t, s_t), \Sigma^\top \Sigma)$ and Σ is defined as $\Sigma = \text{diag}(\sigma_\phi, \sigma_\zeta)$. The standard deviations ($\sigma_\phi, \sigma_\zeta$) are range dependent and given by $\sigma_\phi = \phi_0 + \beta_\phi \|s_t - Hx_t\|_2$ and $\sigma_\zeta = \zeta_0 + \beta_\zeta \|s_t - Hx_t\|_2^2$ with $\phi_0 = \pi/180 \text{ rad}$, $\beta_\phi = 10^{-5} \text{ rad/m}$, $\zeta_0 = 1.5 \text{ m}$, and $\beta_\zeta = 3 \times 10^{-4} \text{ m}^{-1}$. Moreover, the agent receives spurious measurements (i.e., clutter) with fixed Poisson rate $\lambda = 10$, uniformly distributed over the measurement space. The agent's sensing model parameters take the following values: $p_D^{\max} = 0.99$, $\eta = 0.003$, and $R_0 = 10 \text{ m}$. The agent's dynamical model has radial displacement $\Delta_R = 3 \text{ m}$, $N_R = 2$, and $N_\theta = 8$, which gives a total of 17 control actions, including the initial position of the agent. The agents carry isotropic antennas ($G_t = G_r = 1$) which at each time-step transmit jamming signals of continuous power levels with maximum value 0 dBm, that have carrier frequency $f_{GHz} = 2 \text{ GHz}$, or do not transmit any power. The detection threshold $\vartheta = 0.50$ is applied over normalized values and the interference threshold is $\Delta = -70 \text{ dBm}$. The solutions to the proposed MILPs were generated using Gurobi solver in Matlab. Finally, in order to handle the non-linear dynamics and measurement model, Bayes recursion in Eqs. (1) and (2) was implemented using particle filtering techniques [15].

B. Simulation Results

The initial track-and-jam scenario for the evaluation of the proposed approach assumes 3 agents and 2 targets and takes place during 100 time-steps. The scenario is depicted in Fig. 2 for the two proposed objective functions of problems (P1) and (P2) (described in Section V-B). In this scenario, 3 agents and 2 targets enter at $k = 1$ the $100\text{m} \times 100\text{m}$ area as shown

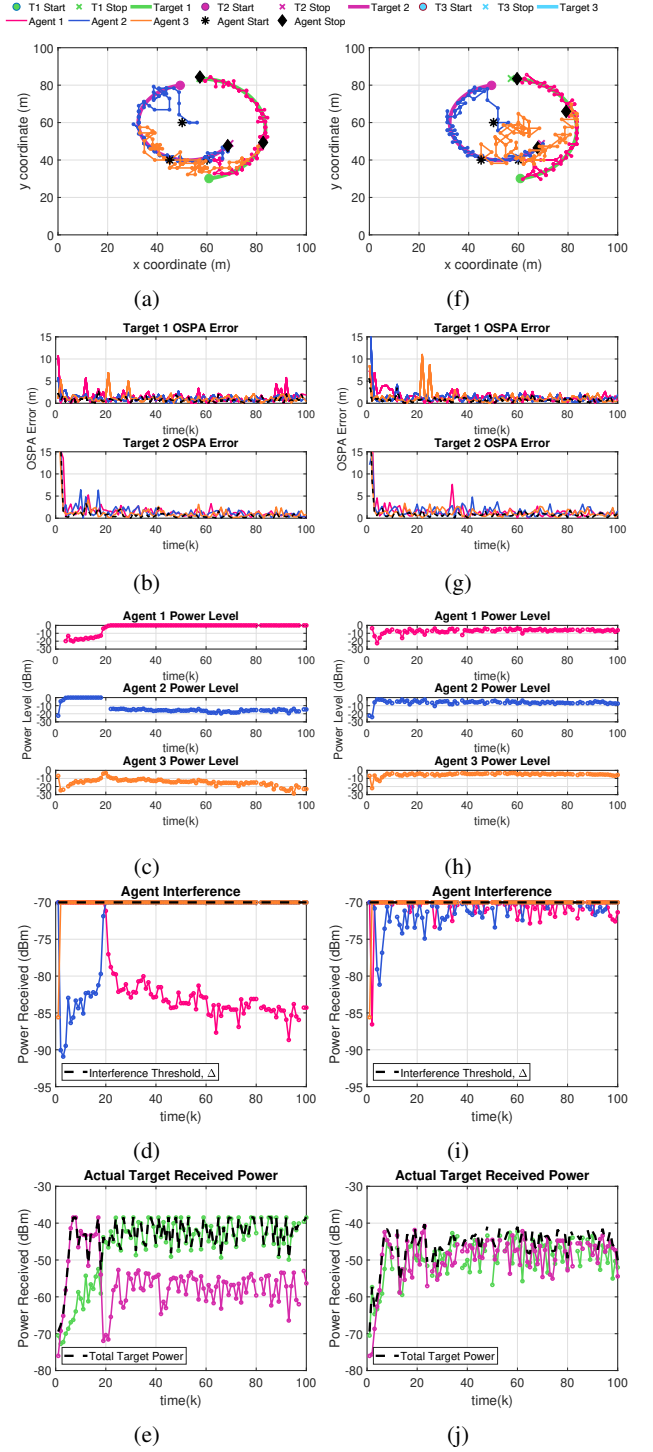


Fig. 2: Performance results for a simulated scenario that utilizes 3 agents and 2 targets for: (i) (P1) (2a - 2e) and (ii) (P2) (2f - 2j).

in Figs. 2a and 2f. The agent initial locations (marked with $*$) are (60, 40), (50, 60) and (45, 40) for agents 1, 2 and 3 respectively, and the target initial locations (marked with \bullet) are (60, 30) and (50, 80) for targets 1 and 2 respectively.

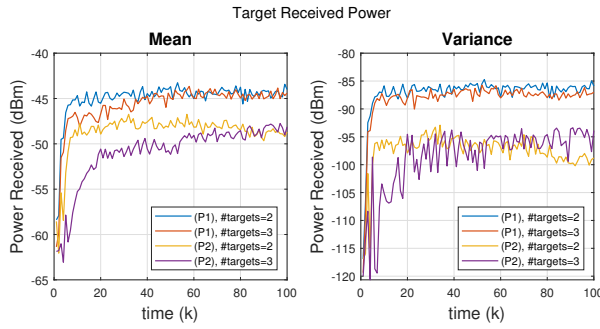


Fig. 3: Performance comparison based on the average received power to target involving 3 agents and varying number of targets.

The target and agent trajectories are shown in Figs. 2a and 2f. The agents use state fusion to improve their target state estimation and as observed from the OSPA error [16] in figures (Figs. 2b and 2g), where the fused state error is drawn with a black dashed line, each agent has satisfactory tracking performance.

The agent-selected power levels for each time-step are shown in Figs. 2e and 2h. The interference power received by each agent remains below the predetermined threshold for each time-step (Figs. 2d and 2i). As expected, the first objective function achieves higher total jamming power received by the targets (Fig. 2e) than the second (Fig. 2j) for each time-step. It is also noted all targets receive almost equal and sufficient jamming power (above -60 dBm) at each time-step utilizing the second objective function. On the contrary, when the first objective function is used, a significant difference between the received power of each target is observed (sometimes more than 20 dBm), and as a result only some of the targets receive sufficient jamming power at a time (a target's received power might even fall below the interference threshold for some time intervals). Nevertheless, through (P1) the subset of targets that are tracked best also receive substantial interference power (well above the sensitivity threshold of the communication and sensing receivers), and as such those targets are successfully intercepted.

This is also demonstrated by the results in Fig. 3, where 50 Monte-Carlo (MC) trials were conducted using the same setup as previously (with 3 agents) and a varying number of targets (2 and 3 targets). (P1) achieves higher target received power (on average) compared to (P2), but the variance of the received power at each target is higher than that of (P2).

VII. CONCLUSIONS

This work proposes utilizing a swarm of pursuer drones to track-and-jam multiple rogue drones. Specifically, a cooperative framework is proposed, where the pursuer drones jointly make optimized decisions on the mobility and power control actions of each individual agent, while ensuring that the received power at the targets is maximized. Performance evaluation results demonstrate the applicability and effectiveness

of the proposed technique to intercept and capture the rogue drones by successfully and efficiently tracking and jamming the rogue drones' sensing and communication components.

ACKNOWLEDGMENTS

This work is partially supported by the European Union Civil Protection under grant agreement No 783299 (SWIFTERS). It is also partially supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No 739551 (KIOS CoE) and from the Republic of Cyprus through the Directorate General for European Programmes, Coordination and Development.

REFERENCES

- [1] B. Nassi, A. Shabtai, R. Masuoka, and Y. Elovici, "Sok-security and privacy in the age of drones: Threats, challenges, solution mechanisms, and scientific gaps," *arXiv preprint arXiv:1903.05155*, 2019.
- [2] Y. Rizk, M. Awad, and E. W. Tunstel, "Decision making in multiagent systems: A survey," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 3, pp. 514–529, 2018.
- [3] T. H. Chung, G. A. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics," *Autonomous Robots*, vol. 31, 2011.
- [4] T. Multerer, A. Ganis, U. Prechtel, E. Miralles, A. Meusling, J. Mietzner, M. Vossiek, M. Loghi, and V. Ziegler, "Low-cost jamming system against small drones using a 3D MIMO radar based tracking," in *Proc. European Radar Conference (EURAD)*, Oct 2017, pp. 299–302.
- [5] Z. Zhou, W. Zhang, J. Ding, H. Huang, D. M. Stipanovic, and C. J. Tomlin, "Cooperative pursuit with Voronoi partitions," *Automatica*, vol. 72, pp. 64 – 72, 2016.
- [6] H. Huang, W. Zhang, J. Ding, D. M. Stipanovic, and C. J. Tomlin, "Guaranteed decentralized pursuit-evasion in the plane with multiple pursuers," in *Proc. IEEE Conference on Decision and Control (CDC) and European Control Conference (ECC)*, Dec 2011, pp. 4835–4840.
- [7] A. Pierson, Z. Wang, and M. Schwager, "Intercepting rogue robots: An algorithm for capturing multiple evaders with multiple pursuers," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 530–537, April 2017.
- [8] A. Pierson, A. Ataei, I. C. Paschalidis, and M. Schwager, "Cooperative multi-quadrotor pursuit of an evader in an environment with no-fly zones," in *Proc. IEEE International Conference on Robotics and Automation*, May 2016, pp. 320–326.
- [9] M. R. Brust, G. Danoy, P. Bouvry, D. Gashi, H. Pathak and M. P. Goncalves, "Defending against intrusion of malicious UAVs with networked UAV defense swarms," in *Proc. IEEE 42nd Conference on Local Computer Networks Workshops*, Oct 2017, pp. 103–111.
- [10] Z. Chen, "Bayesian filtering: From Kalman filters to particle filters, and beyond," Adaptive Syst. Lab., McMaster University, Hamilton, ON, Canada, Technical Report, 2003.
- [11] P.O. Arambel, C. Rago, and R.K. Mehra, "Covariance intersection algorithm for distributed spacecraft state estimation," in *Proc. IEEE American Control Conference (ACC)*, vol. 6, 2001, pp. 4398–4403.
- [12] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.
- [13] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, 2004.
- [14] M. Roth, G. Hendebay, and F. Gustafsson, "EKF/UKF maneuvering target tracking using coordinated turn models with polar/cartesian velocity," in *Proc. 17th International Conference on Information Fusion*, 2014, pp. 1–8.
- [15] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [16] D. Schuhmacher, B. T. Vo, and B. N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, Aug 2008.