Distributed Multi-Target Search and Tracking Using a Coordinated Team of Ground and Aerial Robots

Jun Chen and Philip Dames

***Abstract*— The authors recently developed a distributed algorithm to enable a team of homogeneous robots to search for and track an unknown and time-varying number of dynamic targets. This algorithm combined a distributed version of the PHD filter (for multi-target tracking) with Lloyd's algorithm to drive the motion of the robots. In this paper we extend this previous work to allow a heterogeneous team of ground and aerial robots to perform the search and tracking tasks in a coordinated manner. Both types of robots are equipped with sensors that have a finite field of view and which may receive both false positive and false negative detections. The aerial robots may vary the size of their sensor field of view (FoV) by changing elevation. This increase in the FoV coincides with a decrease in the accuracy and reliability of the sensor. The ground robots maintain the target tracking information while the aerial robots provide additional sensor coverage. We develop two new distributed algorithms to provide filter updates and to make control decisions in this heterogeneous team. Both algorithms only require robots to communicate with nearby robots and use minimal bandwidth. We demonstrate the efficacy of our approach through a series of simulated experiments which show that the heterogeneous teams are able to achieve more accurate tracking in less time than our previous work.**

I. INTRODUCTION

Target search and tracking are fundamental problems in robotics, with applications to mapping, environmental monitoring, surveillance, search and rescue, and more [1]. Many situations call for performing these two tasks simultaneously, where the same team of robots must search a space in order to detect the presence of any targets of interest and then track the motion of any targets that are detected. Most approaches to solve these problems utilize a homogeneous team of robots, allowing all robots to utilize the same governing equations, while a few others seek to utilize a heterogeneous team for superior results. However, we will demonstrate that a heterogeneous team is able to achieve superior results, as long as the coordination mechanism allows the different types of robots to take advantage of their unique capabilities.

Coverage control in heterogeneous sensor networks has been studied by a number of researchers [2]–[6] with the objective of allowing teams mobile sensors with different sensor footprints, detection models, etc. to complete a uniform task together (*e.g.,* surveillance, coverage, or search) under a uniform control law. In contrast, each subteam of our proposed heterogeneous robot team belongs to a separate distributed control system in order to take advantage of its unique capabilities in all aspects including kinematic, sensing, and power.

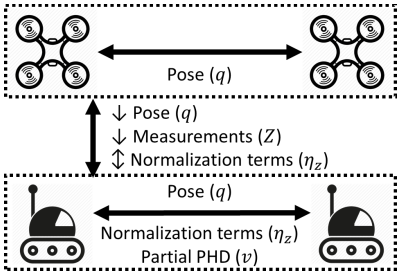


Fig. 1. Information sharing process between and within the two cooperating homogeneous teams.

The coordination of mixed air-ground robot teams has been studied in recent years [7]–[11]. However, all of these prior works use a ground station to monitor the air team, to facilitate communication between air and ground robots, or to make control decisions. Yu *et al.* [12] proposed a distributed cooperative planning algorithm for a combined air-ground team to search and track in urban environments. However, this approach is limited to a single target. The main contribution in our work is the development of a novel distributed algorithm, outlined in Fig. 1, that enables an air-ground robot team to search for and track multiple targets. By using distributed algorithms, our approach scales to arbitrarily large teams of robots and ensures that the multi-robot system is robust to the failure of any individual agent(s). The advantage of our proposed method lies in its ability to scale up both team scale and the number of subteams while taking advantage of each type of robot in order to achieve more accurate tracking in less time, which has not shown in previous literature.

In the combined search and tracking task, the targets may be either stationary or dynamic, the number of targets may change over time as they enter or leave the area of interest, and the target motion is uncertain. Additionally, the robots are equipped with noisy and unreliable sensors. To provide a reliable target estimate that accounts for all of these sources of uncertainty, we use the Probability Hypothesis Density filter [13], which provides an estimate of the spatial density of targets. Note that unlike other multi-target trackers, the PHD filter does *not* provide target tracks, *i.e.,* unique identities of each target [14]. However, the goal of this work is to locate and track all of the targets, not to uniquely identify each individual target. Furthermore, compared to other multi-target trackers, the PHD filter offers a straight-

*This work was supported by NSF Grant IIS-1830419.

J. Chen and P. Dames are with the Department of Mechanical Engineering, Temple University, Philadelphia, PA 19122, USA {jchen,pdames}@temple.edu

forward computational form and does not require solving the data association problem, *i.e.,* matching measurements to targets, a difficult and computationally intensive task when the density of targets is high.

Previously, we introduced a distributed version of the PHD filter and tested it with ground-only [15] and air-only [16] teams. We used the output of the PHD filter as the importance weighting function within Lloyd's algorithm [17]. This effectively drives the robots to follow previously detected targets and to explore unknown areas that may contain targets. However, we will demonstrate that a straightforward application of this previous approach to a heterogeneous team is not as effective as an approach that treats each type of robot differently. Specifically, we exploit the differences in robots' mobility and computation to develop distributed algorithms to enable this coordination. We demonstrate the efficacy of this approach, relative to a naïve extension of the algorithms for a homogeneous team, through a series of simulated experiments. It is beyond the scope of this paper to compare our work against a centralized solution [7]–[11].

II. PROBLEM FORMULATION

A team of R_g ground robots and R_a aerial robots track the motion of an unknown number of targets, which move in a convex polygonal environment $E \subset \mathbb{R}^2$. The environment must be convex in order to compute the Voronoi cells, though this requirement can be relaxed by using the results from Bhattacharya et al. [18]. At time t, the pose of ground robot r_g is $q^t_{r_g} \in SE(2)$ and of aerial robot r_a is $q^t_{r_a} \in SE(3)$. At each time step, a robot r (either ground or air) collects a set of m measurements, $Z = \{z_1, \ldots, z_m\}$. m varies over time due to false positive and false negative detections and due to the motion of both targets and robots, which cause targets to enter and leave the sensor field of view (FoV). The team seeks to determine the set of targets, $X^t = \{x^t_1, \ldots, x^t_n\}$, where each $x^t_i \in E$. Note that this set encodes both the number of targets (*i.e.,* the cardinality of the set $|X^t|$) and the state of each target (*i.e.,* the elements of the set x^t_i).

A. PHD Filter

The sets X and Z from above are realizations of random finite sets (RFSs). An RFS is a set containing a random number of random elements, *e.g.,* each of the n elements x_i in the set $X = \{x_1, \ldots, x_n\}$ is a vector indicating the state of a single target [19]. The first order moment of an RFS is known as the *Probability Hypothesis Density* (PHD) [20], which we denote $v(x)$. The PHD takes the form of a *target* density function over the state space of a target. The PHD filter recursively updates this target density function in order to track the motion of the targets. Note that the PHD filter is the continuous version of the bin-occupancy filter [21].

The PHD filter uses three models to describe the motion of the targets: 1) The motion model, $f(x \mid \xi)$, describes the likelihood of an individual target transitioning from an initial state ξ to a new state x. 2) The survival probability model, $p_s(x)$, describes the likelihood that a target with state x will continue to exist from one time step to the next. 3) The birth PHD, $b(x)$, encodes both the number and locations of the new targets that may appear in the environment.

The PHD filter also uses three models to describe the ability of robots to detect targets: 1) The detection model, $p_d(x \mid q)$, gives the probability of a robot with state q successfully detecting a target with state x. Note that the probability of detection is identically zero for all x outside the sensor FoV. 2) The measurement model, $g(z \mid x, q)$, gives the likelihood of a robot with state q receiving a measurement z from a target with state x. 3) The false positive (or clutter) PHD, $c(z \mid q)$, describes both the number and locations of the clutter measurements.

Using these target and sensor models, the PHD filter prediction and update equations are:

$$\bar{v}^t(x) = b(x) + \int_E f(x \mid \xi) p_s(\xi) v^{t-1}(\xi)\, d\xi \tag{1}$$

$$v^t(x) = (1 - p_d(x \mid q))\bar{v}^t(x) + \sum_{z \in Z_t} \frac{\psi_{z,q}(x)\bar{v}^t(x)}{\eta_z(\bar{v}^t)} \tag{2}$$

$$\eta_z(v) = c(z \mid q) + \int_E \psi_{z,q}(x) v(x)\, dx \tag{3}$$

$$\psi_{z,q}(x) = g(z \mid x, q) p_d(x \mid q), \tag{4}$$

where $\psi_{z,q}(x)$ is the probability of a sensor at q receiving measurement z from a target with state x. In this work we represent the PHD using a set of weighted particles [22].

B. Lloyd's Algorithm

Lloyd's algorithm minimizes the value of the functional

$$\mathcal{H}(\{q_1, \ldots, q_R) = \int_E \min_{r \in \{1,\ldots,R\}} f\big(d(x, q_r)\big)\phi(x)\, dx, \tag{5}$$

where $d(x, q)$ measures the distances between elements in E, $f(\cdot)$ is a monotonically increasing function, and $\phi(x)$ is a non-negative weighting function. We use $f(x) = x^2$, a standard choice. The minimum inside of the integral induces a partition on the environment $V_r = \{x \mid d(x, q_r) \leq d(x, q_i), \forall i \neq r\}$. This is the Voronoi partition, and these V_r are the Voronoi cells. Cortes *et al.* [23] show that the gradient of (5) with respect to the state of each robot is independent of the states of the other robots, and that iteratively moving each robot r to its weighted centroid,

$$q^*_r = \frac{\int_{V_r} x\phi(x)\, dx}{\int_{V_r} \phi(x)\, dx}, \tag{6}$$

achieves a local minimum of $\mathcal{H}$. Our prior work introduced the use of the PHD as the weighting function, setting $\phi(x) = v(x)$ [16], effectively coupling the tracking and control.

C. Distributed PHD Filter

Our previous work [16] uses the Voronoi partition in order to distribute the PHD filter across a team of robots. Specifically, each robot is responsible for maintaining the portion of the PHD within its cell. As the robots move, so do their Voronoi cells. When this happens, the team transfers ownership of portions of the PHD from one robot to another [16, Algorithm 1]. As the targets move, they may cross

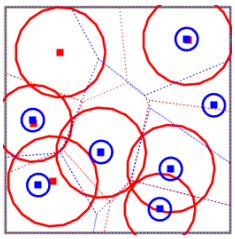


Fig. 2. A heterogeneous team of 7 ground (blue squares) and 7 aerial robots (red squares) shown with their sensor footprints (circles). Each type of robot has its own Voronoi partition (dashed lines) and the two types of robots cooperate to locate targets.

the boundary from one cell into another. The distributed version of the PHD filter prediction step, (1), requires each robot to share the flux of targets across the boundaries of its cell with its neighbors [16, Algorithm 2]. Finally, the normalization term in the PHD update equation, (2), requires an integral over the full sensor FoV. When a robot's FoV overlaps with the Voronoi cell of another robot, the two robots must exchange data in order to correctly compute the normalization term (3) [16, Algorithm 3]. When the sensor FoV of two or more robots overlap, each robot must also apply the measurement updates in the same order to ensure that the resulting PHD is consistent. This enables each robot to update the PHD using only information from its neighbors, offering a significant advantage for large teams, like the ones considered in this paper. We showed that this distributed PHD filter yields identical results to a centralized filter.

III. Coordination of Heterogeneous Robots

When using a team of heterogeneous robots, we argue that the most effective coordination strategies should take advantage of the differences between each type of robot. To accomplish this, each subteam has a separate Voronoi diagram (Figure 2). The only requirement is that each subteam is able to move independently of the others, *i.e.,* a robot in subteam 1 cannot pick an action that will cause it to collide with a robot in subteam 2. This allows each subteam to take advantage of their differences in mobility.

However, the subteams must still exchange information with one another in order to effectively coordinate. In our scenario, the most relevant piece of information is the current target estimate, which is represented by the PHD. One subteam maintains the PHD and shares this with the other subteams. This allows the subteam with the greatest computational and communication resources to perform the tracking task. We assume that each robot has sufficient communication range and bandwidth to exchange information with all of its Voronoi neighbors and with all other robots who have Voronoi cells that overlap with its sensor FoV.

In this paper, we assume that we have two types of robots: one that is constrained to move on the ground and one that moves in the air. Aerial robots have a higher maximum speed while the ground robots have more computational resources, which is the typical case for real-world systems.

Algorithm 1 Distributed PHD Update

1: **for** each aerial robot r_a **do**
2: Send pose $q^t_{r_a}$ and measurement set $Z^t_{r_a}$ to all ground neighbors
3: **end for**
4: **for** each ground robot r_g **do**
5: Compute (3) within Voronoi cell V_{r_g}
6: Send partial η_z back to the respective aerial robot(s)
7: **end for**
8: **for** each aerial robot r_a **do**
9: Compute full normalization term (3)
10: Send normalization term to all ground neighbors
11: **end for**
12: **for** each ground robot r_g **do**
13: Compute PHD update (2)
14: **end for**

A. Distributed PHD Update

Due to their superior computational resources, we utilize the ground subteam, along with its associated Voronoi partition, to maintain the distributed PHD filter [16]. In order to incorporate the measurement data from the aerial robots, we must add an additional step. Outlined in Algorithm 1 and Fig. 1, this requires three rounds of communication between the subteams. First, each aerial robot r_a must send its pose and measurement set to each ground robot whose Voronoi cell overlaps with r_a's sensor FoV. The ground robots, which have the probabilistic sensor models for the aerial robots stored on board, then compute the portion of the normalization integral within their Voronoi cells and send this back to the aerial robot. The aerial robot then computes the full normalization term and sends this back to the ground robots, which perform the PHD updates. Note that all of the ground robots must apply the measurements from the aerial robots in the same order as one another. This can be done effectively by using the ID of each aerial robot to provide a total ordering, just like the ordering of updates in the original algorithm [16, Algorithm 3].

Using this algorithm, the mixed air-ground team can achieve a consistent PHD estimate. This algorithm utilizes the additional computational power of the ground robots to maintain the filter and the additional sensing provided by the aerial robots to detect and track targets. The algorithm is also scalable to large teams. Each exchange of data between any pair of robots is very low bandwidth: each step involves sending $|Z|$ quantities (measurements or η terms). Furthermore, the number of robots that must communicate with each other is of constant size with respect to the team, since each robot is only required to communicate with its direct neighbors.

B. Distributed Goal Assignment

Ground robots use the same distributed goal assignment algorithm from our previous work [16]: each robot computes the weighted centroid of its Voronoi cell using the PHD as the weighting function and drives towards this point.

Algorithm 2 Aerial Robot Goal Assignment

for Each aerial robot r_a **do**
  Send Voronoi cell $V_{r_a}^t$ to all ground neighbors $\mathcal{N}_{r_a}$
end for
for Each ground robot r_g **do**
  Compute overlapping region(s) $V_{g,a} = V_{r_a} \cap V_{r_g}$
  Compute partial integrals $\int_{V_{g,a}} v(x)\,dx$, $\int_{V_{g,a}} xv(x)\,dx$, and $\int_{V_{g,a}} xx^T v(x)\,dx$
  Send results back to the respective aerial robot(s)
end for
for Each aerial robot r_a **do**
  Add partial integrals to get full integrals
  Compute centroid $q_{2D}^* = \frac{\int xv(x)\,dx}{\int v(x)\,dx}$
  Compute covariance $\Sigma = \frac{\int xx^T v(x)\,dx}{\int v(x)\,dx} - q_{2D}^*(q_{2D}^*)^T$
end for

Aerial robots cannot do this on their own since they have no information about the PHD stored on board. Instead, they must communicate with nearby ground robots to compute their goal locations. To pick actions for aerial robots, we take the approach from [16] and separate the motion of the aerial robots into two components: in-plane and out-of-plane. The in-plane motion is determined in the same way as the ground robots, using a (separate) 2D Voronoi diagram and applying Lloyd's algorithm using the PHD as the importance weighting function. Let the in-plane component of the aerial robot's goal position be q_{2D}^*.

The out-of-plane motion (*i.e.,* elevation) depends upon the distribution of both the aerial robots and the targets [16]. Since for most sensors, *e.g.,* downward-facing cameras, the diameter of the FoV is proportional to the elevation of the robot, we must first select a desired sensor radius. We compute this using two factors. The first factor, r_{cell}, depends on the spatial distribution of robots, and is the average of the radii of the circles centered at q_{2D}^* that are inscribed in and circumscribe the Voronoi cell. The more densely packed the robots are, the smaller their cells will be and so the lower each robot needs to be to ensure coverage. The second factor, r_{targets}, depends on the target distribution within the cell.

$$r_{\text{targets}} = 3\sqrt{\max\left(\text{eig}(\Sigma)\right)} \tag{7}$$

$$\Sigma = \frac{\int_V (x - q_{2D}^*)(x - q_{2D}^*)^T w(x)\,dx}{\int_V w(x)\,dx}, \tag{8}$$

where $\text{eig}(\Sigma)$ are the eigenvalues of the matrix. The more certain each robot is about the locations of targets, the lower it is able to fly while ensuring coverage.

The desired radius of the sensor footprint is given by the weighted sum of these two factors

$$r_{\text{des}} = \frac{w_{\text{cell}} r_{\text{cell}} + w_{\text{targets}} r_{\text{targets}}}{w_{\text{cell}} + w_{\text{targets}}}, \tag{9}$$

where w_{cell} is a constant value and $w_{\text{targets}} = \int_V w(x_{2D})\,dx_{2D}$ is the total weight of the PHD in the cell. This formula makes a trade-off between maintaining coverage in order to view possible new targets and focusing in on existing targets to maintain better tracking quality.

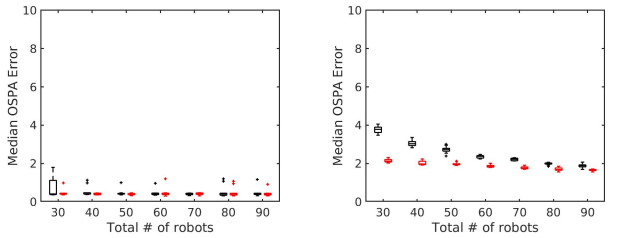
The aerial robots use Algorithm 2 to compute goal locations. We make a common assumption that each robot is capable of finding its Voronoi neighbor set with local information [24]. This distributed algorithm uses two rounds of low-bandwidth communication in order to compute the 2D centroid, q_{2D}^*, and target covariance, Σ. In order to reduce the amount of communication necessary, we use the equation $\Sigma = E[XX^T] - E[X]E[X]^T$ since this allows the robots to compute all of the terms in parallel.

IV. SIMULATIONS

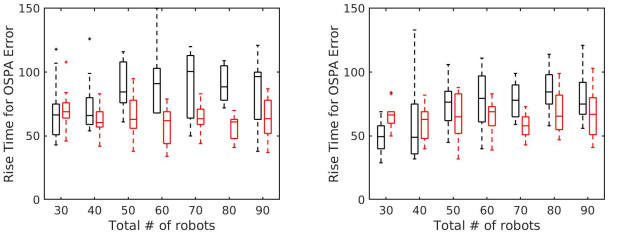
We conduct a set of simulated experiments using MATLAB in order to demonstrate the efficacy of our proposed distributed estimation and control algorithms. All robots know their pose at all times and are modeled as point robots that are both holonomic and kinematic. Ground robots have a maximum speed of 2 m/s while aerial robots have a maximum speed of 10 m/s. The elevation of each aerial robot is constrained to a certain range, since flying too low could lead to collisions with ground robots and UAVs are legally restricted to a maximum elevation. Each robot is equipped with an isotropic sensor with a finite sensing range to detect targets (5 m for ground robots). All targets within the sensor's FoV will be detected with equal possibility. As an aerial robot moves up in elevation, its sensor FoV increases in size while the detection probability decreases and the noise and clutter increase. All of the sensor and target models match those from our previous work [16].

The environment is an open 100×100 m area with no obstacles. All robots begin each trial at randomized locations within a $20\,\text{m} \times 10\,\text{m}$ box at the bottom center of the environment. Aerial robots begin at an elevation of 5 m. The PHD is represented by a uniform grid of particles. The grid resolution is 1 m, and initially the weight of each particle is set to $w_j = 10^{-4}$, so that the total expected number of targets is initially 1. We use the first order Optimal SubPattern Assignment (OSPA) metric [25], a commonly-accepted approach, with cutoff distance 10 m to measure the average error between true target set X and measurement set Z. A lower OSPA value indicates a more accurate tracking of the target set.

We consider two scenarios, one in which the targets are stationary and one in which the targets are dynamic. When the targets are stationary, the team explores for a maximum of 300 s. We plot the median value of the OSPA error over the final 50 s of the run to get the steady-state value. In the case of dynamic targets, the number of targets varies over time as new targets enter the search area and others leave it. To measure the steady-state performance, the robots explore for 1000 s and we plot the median OSPA error over the final 300 s, giving time for the robots to spread out across the environment. In both cases, we also measure the 95% rise time of the OSPA error metric, meaning the time it takes for the OSPA error to reach a value within 5% of the final



(a) OSPA – 20 Static Targets (b) OSPA – 20 Dynamic Targets



(c) Rise Time – 20 Static Targets (d) Rise Time – 20 Dynamic Targets

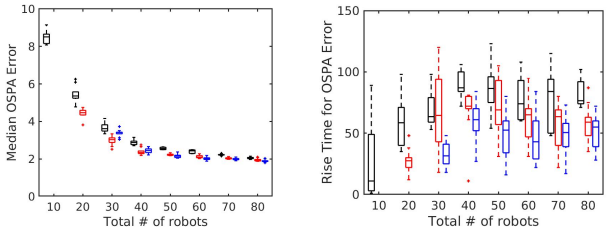
Fig. 3. Boxplots showing the results of teams using our `old` algorithm (black) and our `new` algorithm (red). The teams consist of 30–90 total robots, including 20 aerial robots, and track 20 static or dynamic targets. (a) and (b) show the steady-state OSPA error while (c) and (d) show the 95% rise time of the OSPA error.

steady-state value. We perform 10 trials of each configuration and show boxplots of all results.

1) Comparison with Previous Method: We first want to demonstrate that a team using our proposed coordination algorithm performs better than a team using our previous approach [16]. To do this we consider robot teams consisting of 20 aerial robots plus a varying number of ground robots, from 10–70, for a total team size of 30–90 robots. In the first scenario (`old`), we will use a naïve extension of our previous work: the team uses a single shared Voronoi diagram and both types of robots maintain portions of the PHD. In the second scenario (`new`), the heterogeneous team will use the algorithm proposed in this paper. Note that the target behavior is identical for both scenarios, so the only difference is in the motion of the robots (which also influences the measurements received).

Figure 3 shows the results of these trials, with the `new` team outperforming the `old` team in terms of accuracy, precision, speed, and repeatability. Figures 3a and 3b show that teams using `new` have a lower steady-state OSPA error and the spread in values is smaller compared to teams using `old`. Furthermore, Fig. 3c and 3d show that teams using `new` have a lower median rise time and a smaller spread. The only instances when the `old` team has a lower rise time is when the steady-state OSPA error is significantly higher, which naturally takes less time to reach.

The emergent behavior of the `new` team is such that the aerial robots quickly spread out over the environment at a high elevation, providing relatively low-quality measurements of a large area. These measurements from the aerial robots allow the ground robots to use information from outside of their own sensor FoV to move directly towards



(a) OSPA – 10 Dynamic Targets (b) Rise Time – 10 Dynamic Targets

Fig. 4. Boxplots showing the results of teams of 10–80 total robots, including 0 (black), 10 (red), and 20 (blue) aerial robots, tracking 10 dynamic targets. (a) shows the OSPA error while (b) shows the 95% rise time of the OSPA error.

areas that *likely* contain targets. This symbiotic relationship allows the full team to take advantage of the different strengths of each type of robot, thereby achieving superior results. In the `old` team, the aerial robots are significantly slowed because all robots share a single Voronoi diagram and so the aerial robots must first wait for the slower ground robots to spread out.

2) Moving Targets: In order to further explore the effects of coordination, we conduct trials using three different team compositions: ground robots only, ground robots with 10 aerial robots, and ground robots with 20 aerial robots. In each case there are initially 10 moving targets. As expected, all three team compositions show decreasing OSPA errors and rise times as the total number of robots increases. This is due to the ability of the team to better cover the boundaries of the environment to ensure that fewer new targets are missed while simultaneously tracking previously detected targets. As the team size increases, we see that the gap between the three team compositions narrows, the results become more consistent, and there are diminishing returns for adding more robots. All of this occurs because the environment becomes saturated with robots.

In every case, the ground-only team performed worse than either of the teams with aerial robots, demonstrating the utility of coordination. However, a team with 20 aerial robots does not uniformly perform better than a team with 10 aerial robots. Looking at teams with 30 total robots, we see that the team with 20 ground and 10 aerial robots has lower OSPA error than the team with 10 ground and 20 aerial robots. This is because in the steady state there were approximately 15–20 targets, so only the team with 20 ground robots has sufficient resources to track all of the targets using the higher precision sensors. Looking at the rise time for this same case, we see that the team with 20 aerial robots has a significantly lower and more consistent rise time than the team with 10 aerial robots. This again supports the intuition that the aerial robots provide more consistent information, but at a lower quality, than the ground robots.

3) Team Composition: To further explore the effects of team composition that we observed before, we consider the scenario of 51 robots tracking 20 dynamic targets. Note that the team always has at least one ground robot, to maintain

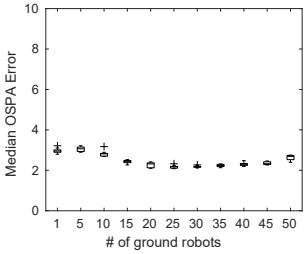


Fig. 5. Boxplots showing the OSPA error of teams of 51 total robots, including 1 to 50 ground robots, tracking 20 dynamic targets.

the PHD filter, and one aerial robot. Figure 5 shows the OSPA error as a function of the number of ground robots. Note that the minimum error occurs when the number of ground and aerial robots are nearly equal. In this case, the two types of robots can essentially pair off, with the aerial robot providing lower-quality coverage of a larger area. This reduces the number of targets that ground robots lose track of due to repeated false negative detections over a short time span. This is especially helpful since the targets have the same maximum speed as the ground robots.

V. CONCLUSIONS

In this paper we propose a distributed method that enables a heterogeneous team of ground and aerial robots to effectively search for and track an unknown number of targets in a known environment. The coordination mechanism takes advantage of the relative strength of each type of platform: ground robots offer increased computational ability and more precise sensors while aerial robots offer increased mobility and a variable sensor field of view. The ground team is responsible for maintaining the distributed multi-target tracker, using sensor measurements from both the ground and aerial robots. Since the control actions depend on the target locations, this requires the aerial robots to communicate with the ground robots in order to make decisions about where to next move. All of this communication is done over low-bandwidth channels and only requires local communication links, making the data-sharing algorithms distributed and scalable to large teams.

We demonstrate the effectiveness of the proposed heterogeneous coordination mechanism through a series of simulated experiments. Overall, the results indicate that teams using the proposed method perform better than homogeneous teams or heterogeneous teams that treat all robots identically. We found that changing the ratio of ground to aerial robots leads to different behavior in terms of the final tracking accuracy and the time necessary to achieve it. The results also show that the proposed algorithm is superior to our previous method when applied to different types of robots.

REFERENCES

[1] C. Robin and S. Lacroix, "Multi-robot target detection and tracking: taxonomy and survey," *Autonomous Robots*, vol. 40, no. 4, pp. 729–760, 2016.

[2] L. Lazos and R. Poovendran, "Stochastic coverage in heterogeneous sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 2, no. 3, pp. 325–358, 2006.

[3] M. Santos, Y. Diaz-Mercado, and M. Egerstedt, "Coverage control for multirobot teams with heterogeneous sensing capabilities," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 919–925, 2018.

[4] L. C. Pimenta, V. Kumar, R. C. Mesquita, and G. A. Pereira, "Sensing and coverage for a network of heterogeneous robots," in *2008 47th IEEE conference on decision and control*. IEEE, 2008, pp. 3947–3952.

[5] Y. Kantaros, M. Thanou, and A. Tzes, "Distributed coverage control for concave areas by a heterogeneous robot–swarm with visibility sensing constraints," *Automatica*, vol. 53, pp. 195–207, 2015.

[6] I. I. Hussein, D. M. Stipanovic, and Y. Wang, "Reliable coverage control using heterogeneous vehicles," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 6142–6147.

[7] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robotics and Automation Magazine*, pp. 16–25, 2006.

[8] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, "Multi-UAV-aided networks: aerial-ground cooperative vehicular networking architecture," *IEEE Vehicular Technology Magazine*, vol. 10, no. 4, pp. 36–44, 2015.

[9] E. Z. MacArthur, D. MacArthur, and C. Crane, "Use of cooperative unmanned air and ground vehicles for detection and disposal of mines," in *Intelligent Systems in Design and Manufacturing VI*, vol. 5999. International Society for Optics and Photonics, 2005, p. 599909.

[10] A. Elfes, M. Bergerman, J. R. H. Carvalho, E. C. de Paiva, J. Ramaos, and S. S. Bueno, "Air-ground robotic ensembles for cooperative applications: concepts and preliminary results," in *Proceedings of the International Conference on Field and Service Robotics*, 1999.

[11] A. Viguria, I. Maza, and A. Ollero, "Distributed service-based cooperation in aerial/ground robot teams applied to fire detection and extinguishing missions," *Advanced Robotics*, vol. 24, no. 1-2, pp. 1–23, 2010.

[12] H. Yu, K. Meier, M. Argyle, and R. W. Beard, "Cooperative path planning for target tracking in urban environments using unmanned air and ground vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 2, pp. 541–552, 2015.

[13] R. P. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[14] L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell, *Bayesian multiple target tracking*. Artech House, 2013.

[15] P. Dames, "Distributed multi-target search and tracking using the PHD filter," in *International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*. IEEE, 2017.

[16] ——, "Distributed multi-target search and tracking using the PHD filter," *Autonomous Robots*, 2019.

[17] S. Lloyd, "Least squares quantization in PCM," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[18] S. Bhattacharya, N. Michael, and V. Kumar, "Distributed coverage and exploration in unknown non-convex environments," in *Distributed Autonomous Rrobotic Systems*. Springer, 2013, pp. 61–75.

[19] R. P. Mahler, *Statistical multisource-multitarget information fusion*. Artech House, 2007.

[20] ——, "A theoretical foundation for the Stein-Winter probability hypothesis density (PHD) multitarget tracking approach," Army Research Office Alexandria VA, Tech. Rep., 2000.

[21] O. Erdinc, P. Willett, and Y. Bar-Shalom, "The bin-occupancy filter and its connection to the PHD filters," *IEEE Transactions on Signal Processing*, vol. 57, no. 11, pp. 4232–4246, 2009.

[22] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, 2005.

[23] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[24] M. Schwager, D. Rus, and J.-J. Slotine, "Decentralized, adaptive coverage control for networked robots," *The International Journal of Robotics Research*, vol. 28, no. 3, pp. 357–375, 2009.

[25] 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, 2008.