Detecting, Localizing, and Tracking an Unknown Number of Moving Targets Using a Team of Mobile Robots

Philip Dames, Pratap Tokekar, and Vijay Kumar

Abstract Target tracking is a fundamental problem in robotics research and has been the subject of detailed studies over the years. In this paper, we introduce a new formulation, based on the mathematical concept of random finite sets, that allows for tracking an unknown and dynamic number of mobile targets with a team of robots. We show how to employ the Probability Hypothesis Density filter to simultaneously estimate the number of targets and their positions. Next, we present a greedy algorithm for assigning trajectories to the robots that maximize submodular objective functions and prove that this is a 2-approximation. We examine two such objective functions: the mutual information between the estimated target positions and future measurements from the robots, and the expected number of targets detected by the robot team. We provide extensive simulation evaluations using a real-world dataset.

1 INTRODUCTION

Target detection, localization, and tracking has many applications, ranging from search-and-rescue [9] to building smart cities [19]. Consequently, such problems have long been a subject of study in the robotics community. Active target tracking typically refers to two types of tasks: estimating the trajectories of the targets from the sensor data, and actively controlling the motion of the robotic sensors to gather the data. Both problems have been studied in the literature under many different settings. Solutions have been presented for radio-based sensors [12], range-only sensors [34], bearing sensors [21], and range and/or bearing sensors [35], under centralized and decentralized settings. Frew and Rock [8] design optimal trajectories for a single robot to track a single moving target using monocular vision. The problem of keeping targets in a robot's field-of-view can be formulated as a visual servoing problem. Gans et al. [10] design a controller which guarantees stability while keeping three or fewer targets in the field-of-view of a single mobile robot.

P. Dames, P. Tokekar, and V. Kumar
University of Pennsylvania, Department of Mechanical Engineering and Applied Mechanics, Philadelphia, PA 19104, USA, e-mail: {pdames, tokekar, kumar}@seas.upenn.edu

Spletzer and Taylor [28] present a general solution for the multi-robot, multi-target case using a particle filter formulation. Tracking multiple targets with multiple robots requires explicit or implicit assignment of targets to robots. Xu et al. [32] present a mixed nonlinear integer programming formulation for assigning robots to targets as well as for determining optimal robot positioning. Such a formulation is not directly applicable in our case since the number of targets itself is unknown, and thus explicit assignment is not possible. Recently, there has been some work on actively detecting and/or localizing an unknown number of stationary targets using radio sensors [14, 27], range-only sensors [4], and arbitrary sensor models [7].

Unlike most existing work, we study the case of tracking an *unknown* and *varying* number of *indistinguishable* targets. This introduces a number of challenges. First, we cannot maintain a separate estimator for each target, since the required number of estimators is unknown. Second, we must account for the fact that targets appear and disappear from the environment. Third, we cannot maintain a history of the target positions because we cannot uniquely identify individual targets, making prediction difficult. Finally, the system must be capable of handling false positive and false negative detections and unknown data association in addition to sensor noise. Despite these challenges, we present positive results towards solving the problem.

To solve this estimation problem we turn to Random Finite Sets (RFSs). RFSs are random variables whose realizations are finite sets. Distributions over RFSs have both a distribution over the cardinality of the set (*i.e.,* number of targets) and a distribution over the elements of the set (*i.e.,* position of the targets). The Probability Hypothesis Density (PHD) filter [22] is the most common estimation strategy based on RFSs. The PHD filter has recently received attention in robotics for use in robot localization [2], simultaneous localization and mapping [18], localizing static targets [7, 26], and more [1]. In this paper, we show how the PHD filter can be employed for tracking moving targets (Section 3.3).

An important consideration for target tracking is the motion model for the targets. A number of parametric motion models have been proposed in the literature (see [20] for a detailed survey). We employ a data-driven technique to extract the motion model, instead of assuming any parametric form. Specifically, we use Gaussian Process (GP) regression to learn a map of velocity vectors for the targets, as Joseph et al. do in their work [13]. Additionally, we show how to model the appearance and disappearance of targets within the environment (Section 3.2). Next, we present a control policy to assign trajectories for all robots in order to maximize the objective function over a receding horizon. We study two objective functions using the PHD filter: mutual information and the expected number of detections by the robots. We show that both objective functions are submodular, and use a result based on [30] to prove that our greedy control policy is a 2-approximation (Section 3.4).

In addition to the theoretical analysis we offer, we evaluate our algorithm using simulated experiments. While our framework may be applied to a number of robot and sensor models, for the purposes of testing we restrict our attention to fixed winged aerial robots with downward facing cameras. We use a real-world taxi motion dataset from [23] for the targets and to verify our models. The simulation results reveal that robot teams using the information-based control objective track a

smaller number of targets with higher precision compared to teams that maximize the expected number of detections (Section 4).

2 PROBLEM FORMULATION

We address the problem of a team of R robots monitoring an area in order to detect, localize, and track an unknown number of moving targets using an inexpensive camera. The robots are able to localize themselves within the environment (*e.g.,* using GPS) and robot r has pose q_t^r at time t.

The number of targets, n_t, is unknown and varies over time, since individual targets may enter and leave the area of interest. We use Random Finite Sets (RFSs) to represent the number and state of targets at any time. In the target tracking scenario, RFSs may represent either targets states or measurements. Let $X_t = \{x_{1,t}, x_{2,t}, \ldots, x_{n_t,t}\}$ denote a realization of a RFS of target states at time t. A probability distribution of a RFS is characterized by a discrete distribution over the cardinality of the set and a family of densities for the elements of the set conditioned on the size, *i.e.,*

$$p(X = \{x_1, \ldots, x_n\}) = p(|X| = n)\, p(\{x_1, \ldots, x_n\} \mid |X| = n). \tag{1}$$

The first statistical moment of a distribution over a RFS is called the Probability Hypothesis Density (PHD). The PHD, $v(x)$, is a density function over the state space of an individual target with the property that the integral over any region S is the expected number of targets in that region, *i.e.,*

$$\int_S v(x)\, dx = E[|X \cap S|]. \tag{2}$$

The PHD filter makes the assumption that targets are independent and identically distributed and that the cardinality of the target set is characterized by a Poisson distribution. The likelihood of such an RFS is

$$p(X) = \exp\left(- \int v(x)\, dx \right) \prod_{x \in X} v(x), \tag{3}$$

which is fully characterized by the PHD.

Each robot receives a set of measurements $Z_t^r = \{z_{1,t}^r, z_{2,t}^r, \ldots, z_{m,t}^r\}$ to targets that it detects within the field of view (FoV) of its sensor. The number of measurements, m_t, varies over time due to false negative and false positive detections and the motion of the robots and the targets. Let $p_d(x \mid q)$ denote the probability of a robot at q detecting a target with state x. $p_d(x \mid q) = 0$ for targets outside of the FoV of the sensor and having $p_d(x \mid q) < 1$ indicates the possibility of a false negative, or missed, detection. When a target is successfully detected, the sensor returns a measurement $z \sim g(\cdot \mid x, q)$. The sensor can also return measurements to clutter objects, causing false positive detections. Let $c(z \mid q)$ denote the PHD of clutter measurements.

The PHD filter is somewhat analogous to the Kalman filter, recursively updating the statistical moments necessary to fully characterize a distribution over the target states. Like the Kalman filter, there are two equations: the prediction and the update,

$$\bar{v}_t(x) = b(x) + \int p_s(\xi) f(x \mid \xi) v(\xi)\, d\xi \tag{4}$$

$$v_t(x) = \big(1 - p_d(x \mid q)\big)\bar{v}_t(x) + \sum_{z \in Z_t} \frac{p_d(x \mid q) g(z \mid x, q) \bar{v}_t(x)}{c(z,q) + \int p_d(\xi \mid q) g(z \mid \xi, q) \bar{v}_t(\xi)\, d\xi}. \tag{5}$$

Here $\bar{v}_t(\cdot)$ is the predicted target PHD; $b(\cdot)$ is the PHD of target births, which accounts for new targets entering the area; $p_s(\cdot)$ is the target survival probability, which accounts for targets leaving the area; and $f(\cdot \mid \xi)$ is the target motion model. In the following section, we show how to learn these parameters from a real-world dataset.

Note that the representation in the PHD filter is inherently different from more traditional target trackers. With the PHD, there is no notion of target labels or of individual target tracks. Instead, the PHD filter tracks the density of targets over time, yielding information about the bulk motion rather than about the motion of individual targets. Future work will examine the recent Labeled MeMBer filter [25], which is also based on RFSs but uses a different representation such that it is able to output labeled target tracks.

3 TARGET TRACKING FRAMEWORK

The representative problem that we consider is of a team of fixed-wing aerial robots equipped with downward-facing cameras tracking vehicles driving on the ground. However the same methodology could be extended to work with robots with other mobility constraints (*e.g.,* ground vehicles or quadrotor platforms) and other sensor modalities (*e.g.,* lidars or 3D depth cameras).

3.1 Sensor Parameterization

The problem of detecting vehicles using aerial imagery has been well studied [11, 33]. We use such studies to inform our selection of the sensor detection, measurement, and clutter models. The approaches presented in [11,33] are similar, searching for image features over a range of scales in order to detect cars of different sizes or to detect cars from different elevations or with different image resolutions. In general, the system is able to have a higher detection rate if we accept a larger number of false positive detections [33, Fig. 12], [11, Fig. 8]. The detection rate may also vary with the number of pixels per target, which may be computed, using the robot pose, the approximate length scale of a target, and the image resolution, to be

$$\#\text{ pixels per car} = \text{pixels per radian} \times \arctan \frac{\text{length of target}}{\text{distance from camera to target}}. \tag{6}$$

We assume a logistic relationship between the number of pixels per target, $n_{\text{px}}(x,q)$, and the detection rate,

$$p_d(x \mid q) = p_0 + \frac{p_{d,\max} - p_{d,0}}{1 + \exp(-k(n_{\text{px}}(x,q) - n_{p,0}))}, \tag{7}$$

where $p_{d,0}$, $p_{d,\max}$, k, and $n_{px,0}$ are design parameters.

The camera returns pixel (*i.e.,* bearing) measurements to the cars detected within the image. Using the pose of the robot, we can project measurements onto the ground plane to localize the targets. The measurement model is

$$g(z \mid x,q) = \mathcal{N}(z; [r_x, c_x]^T, \sigma^2 I), \tag{8}$$

where r_x, c_x are the pixel row and column values in an image taken at q, of a target at x, σ is the standard deviation in pixels, and I is a 2×2 identity matrix.

Like the targets, the clutter is modeled as a Poisson RFS, which is completely characterized by the PHD. Without *a priori* knowledge of locations that are likely to have clutter, the best choice is to use a uniform distribution over the measurement space. For most computer vision-based detection algorithms, the expected number of clutter detections depends upon the detection model, with a high detection likelihood resulting in a higher detection rate [11, 33].

3.2 Target Parameterization

In order to predict how the target set evolves, we need models for the motion of individual targets as well as the birth/death processes of the targets. A number of motion models have been proposed in the literature, ranging from adversarial [5] to stochastic [20]. We take a data-driven approach to modeling the targets' motion, utilizing real-world datasets that are available [15]. In particular, we use Gaussian Process (GP) regression [24] to learn the function that maps the position coordinates of the targets to velocity vectors, as shown by Joseph et al. [13]. Unlike [13], we use a single GP rather than a mixture of GPs.

GP regression is a Bayesian approximation technique to learn some function $f(X)$ given measurements $y = f(x) + \varepsilon$ corrupted by Gaussian noise, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Here, $x = [x_1, x_2]^T$ refers to the position coordinates of the targets. We learn two separate functions, f_1 and f_2, one for each axis of the ground plane, assuming that the velocities along the two axes are independent. Instead of assuming a parametric model for f_i, GP regression assumes that the joint distribution of $f_i(X)$ defined over any collection of positions, $X = \{x^1, \ldots, x^k\}$, is always Gaussian. Thus, $f_i(X)$ is completely specified by its mean function, $m_i(X) = E[f_i(X)]$ and covariance function, $k_i(X, X') = E[(f_i(X) - m_i(X))(f_i(X') - m_i(X'))]$.

Given observed velocity vectors Y_1 and Y_2 taken at some subset of positions, X, GP regression predicts the velocity vectors at another set of positions, X^*, as a Gaussian distribution with conditional mean and variance values [24]:

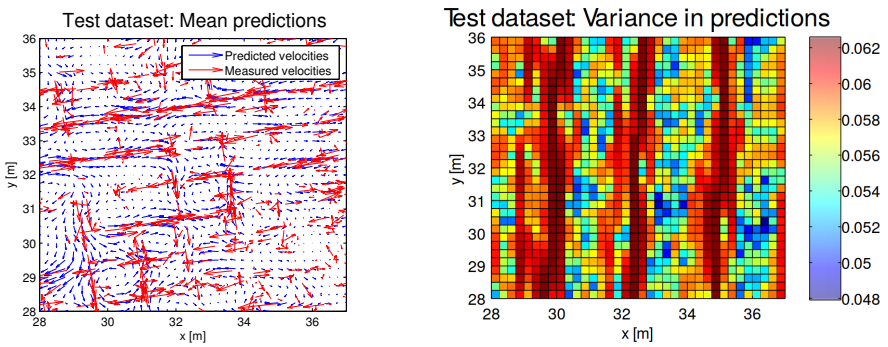


Fig. 1 The mean and covariance of the Gaussian Process regression motion model over a patch of the environment. The measured velocity vectors are shown in red, and the velocity vectors predicted over a grid are given in blue.

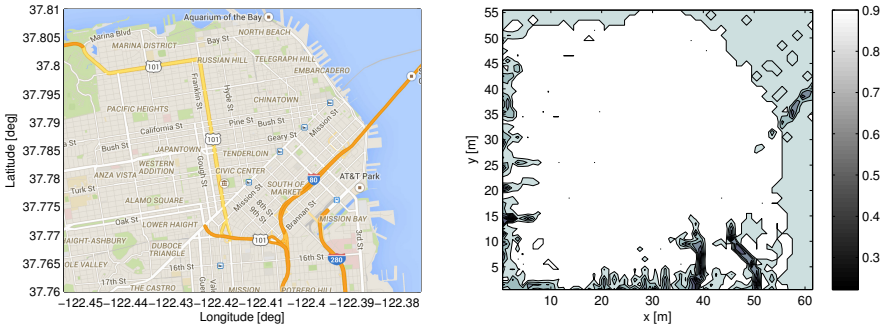
$$m_i(X^*|X) = m_i(X^*) + K_i(X^*,X)[K_i(X,X) + \sigma^2 I]^{-1}(Y_i - m_i(X))$$
$$\sigma^2_{i,X^*|X} = K_i(X^*,X^*) - K_i(X^*,X)[K_i(X,X) + \sigma^2 I]^{-1}K_i(X,X^*),$$

where $K_i(X,X')$ is a matrix whose $(m,n)^{th}$ entry is given by the covariance between $x^m \in X$ and $x^n \in X'$. We take the prior function, $m_i(X)$, to be a zero-mean distribution. Thus, if the covariance function is known, the above equations can fully predict the velocity values at arbitrary positions.

We assume that the covariance function belongs to the Matérn class with parameter $\nu = 3/2$ [24] since this choice of covariance function yields a better fit as compared to the standard squared-exponential function used by Joseph et al. [13]. The length hyperparameter of the Matérn covariance is learned using training data from the Cabspotting taxi dataset from [23]. Figure 1 shows the predicted mean and variance values given by the GP regression using the learned hyperparameter values.

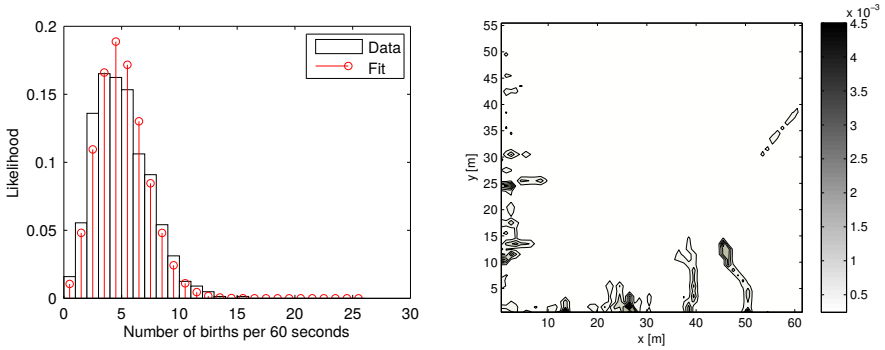
We use an empirical approach to learn the target survival and birth processes. For both processes, we overlay a uniform grid (1 m resolution) over the environment. Whenever a target appears in a cell, we add one to the survival count if the target was previously in another cell, we add one to the birth count if the target was previously outside the environment, and add one to the death count if at the next time step the target leaves the environment. The birth count for each cell is initialized to 10, so that the distribution of birth locations is uniform if there is no data. Similarly, the survival and death count for each cell are initialized to 9 and 1, respectively. The survival probability in a cell is given by the ratio of the survival count to the total survival and death counts in that cell. In the absence of data, this yields a uniform probability of survival of 0.9.

Figure 2a shows the environments used in the simulations, with the target survival probability in Fig. 2b and birth PHD in Fig. 3. As Fig. 2b shows, the targets survive with high probability in the majority of the environment. The probability decreases near the western and southern edges of the environment, where there are roads along



(a) Environment **(b)** Survival probability

Fig. 2 (a) The area of interest, a roughly 6.15×5.56 km region surrounding downtown San Francisco. (b) The probability of target survival as a function of position.



(a) Birth rate **(b)** Birth PDF

Fig. 3 Empirical target birth PHD.

the edge of the environment. These same areas also have the highest rates of target births, as Fig. 3b shows. One may also clearly see the highways in the southeast and the bridge in the northeast, which have the highest rates of traffic, and thus of target births and deaths. The target birth rate per minute, when considering all 536 taxis in the dataset, is 4.548 targets per minute of real time. The actual and fit birth rates are shown in Fig. 3a, with the Poisson approximation fitting the data well.

3.3 PHD Filter

We utilize the Sequential Monte Carlo (SMC) PHD filter from Vo et al. [31]. This approximates the PHD using a set of weighted particles, $v(x) \approx \sum_{i=1}^{P_t} w_i \delta(\mathbf{x} - \mathbf{x}_i)$. The SMC PHD filter allows for arbitrary, non-linear sensor and motion models, including a finite field of view for the sensor. New particles are added to the PHD

using the birth PHD described above as well as using the most recent measurement set. A fixed number of particles, P_b, are drawn from the birth PHD and an additional P_m particles are drawn from the inverse measurement model for each measurements in the most recent set, Z_t. The weight of each of these particles is $w = \frac{\int c(z)\,dz}{P_b + |Z_t| P_m}$, where $|Z_t|$ is the cardinality of the measurement set.

3.4 Control Policy

In this section, we present our control policy for assigning trajectories for the robots. We study two objective functions for the control policy.

3.4.1 Mutual Information (MI) Objective

Mutual information is a way of quantifying the dependence between two random variables [6], and can be defined in multiple ways

$$I[\mathscr{X}, \mathscr{Z}] = \int p(X,Z) \log \frac{p(X,Z)}{p(X)p(Z)}\, dX dZ = \int \mathrm{KL}\big[p(X \mid Z) || p(X)\big]\, p(Z)\, dZ. \quad (9)$$

The last term above states that mutual information can be interpreted as the expected Kullback-Leibler divergence between the prior and the posterior, given the unknown future measurements. Thus, maximizing mutual information between the target set and the future measurements of the robots will cause the robots to take measurements that will change their belief quickly.

The robots utilize a receding horizon control policy. Each robot generates a set of candidate trajectories, with T measurements along each trajectory at evenly spaced intervals. The optimal strategy is then to choose robot trajectories that maximize the mutual information between the target set and its future measurements,

$$Q_\tau^* = \mathrm{argmax}_{Q_\tau \in Q_t^{1:R}} I[\mathscr{X}_{t+T}; \mathscr{Y}_\tau^{1:R} \mid Q_\tau], \quad (10)$$

where $\tau = \{t+1, \ldots, t+T\}$ is the time horizon, $\mathscr{X}_{t+T}$ is the predicted location of the targets at time $t+T$, $\mathscr{Y}_\tau^{1:R}$ is the collection of binary measurements for robots 1 to R from time steps $t+1$ to $t+T$, and Q_τ are the future poses of the robots. These measurements depend on the future locations of the robots $Q_\tau = \{q_{t+1}^1, \ldots, q_{t+T}^1, \ldots, q_{t+T}^R\}$. Computing Q_τ^* is computationally challenging, nevertheless, we show that a greedy strategy approximates Q_τ^* by a factor of 2.

We utilize binary measurements, rather than the full measurements sets, in order to decrease the computational complexity of the control policy. This allows us to derive a closed-form expression for (10), and we have previously shown that this approach effectively drives a team of robots to detect and localize static targets [7]. The binary measurements are defined to be $y = \mathbf{1}(Z \neq \varnothing)$, where $\mathbf{1}(\cdot)$ is the indicator function. Here $y = 0$ is the event that the robot receives no measurements to any (true or clutter) objects while $y = 1$ is the complement of this, *i.e.,* the robot receives at least one measurement. Kreucher et al. [17] take a similar approach, using a bi-

nary sensor model and an information-based objective function to schedule sensors to track an unknown number of targets.

Theorem 1. *The mutual information between the target set and the binary measurement model is a lower bound on the mutual information between the target set and the full measurement set, i.e.,* $I[\mathscr{X};\mathscr{Y} \mid Q] \leq I[\mathscr{X};\mathscr{Z} \mid Q]$.

Proof. Note that y is deterministically related to Z, $y = \mathbf{1}(Z \neq \varnothing)$. This allows us to apply the Data Processing Inequality [6, Theorem 2.8.1], which states that functions of the data cannot increase the amount of information. □

We utilize a greedy approximation strategy to evaluate (10), similar to that used by Tokekar et al. [30]. Using this approach, each robot computes the utility of each action according to (10). The robot and action with the highest utility are selected. The remainder of the team then plans again, conditioned on the action of the first robot, and the robot and action with the highest utility are again selected. This process repeats until all robots have been assigned an action. Using the fact that mutual information is a submodular set function of robot poses, we can show that this greedy assignment policy is a 2-approximation.

Lemma 1. $I[\mathscr{X};\mathscr{Y} \mid Q]$ *is a submodular set function of* Q.

Proof. See [16, Proposition 2]. □

Theorem 2. *Let* Q^G *be the robot poses selected by the greedy assignment policy and* Q^* *be the robot actions selected by the full, joint evaluation of* (10)*. Then greedy is a 2-approximation, i.e.,* $I[\mathscr{X};\mathscr{Y} \mid Q^G] \geq \frac{1}{2} I[\mathscr{X};\mathscr{Y} \mid Q^*]$.

Proof. It is known that the greedy algorithm yields a 2-approximation for maximizing a monotone, submodular function subject to a partition matroid constraint [3]. We can create a set system using the candidate robot actions, as shown in [30]. This set system defines a partition matroid, which along with the previous lemma proves the desired result. □

3.4.2 Expected Number of Detections (END) Objective

The Expected Number of Detections (END) objective function is given by

$$N[X \mid Q] = \int \left(1 - \prod_{q \in Q} \left(1 - p_d(x \mid q)\right)\right) v(x)\, dx. \tag{11}$$

This objective gives the expected number of targets detected by at least one robot, and is a submodular set function of Q so the greedy assignment algorithm will be a 2-approximation, similar to the previous theorem.

Lemma 2. *The END objective function,* $N[X \mid Q]$*, is a submodular function of* Q.

Proof. The difference in the objective when adding a single robot is

$$N[X;Q\cup\{q'\}]-N[X;Q]=\int p_d(x\mid q')\prod_{q\in Q}(1-p_d(x\mid q))v(x)\,dx.$$

For any $R\subseteq Q$, the product $\prod_{r\in R}(1-p_d(x\mid r))\geq\prod_{q\in Q}(1-p_d(x\mid q))$ since $p_d(x\mid q)\in[0,1],\forall x,q$. Thus $N[X;R\cup\{q'\}]-N[X;R]\geq N[X;Q\cup\{q'\}]-N[X;Q]$, so by definition $N[X,Q]$ is submodular. □

3.4.3 Trajectory Generation

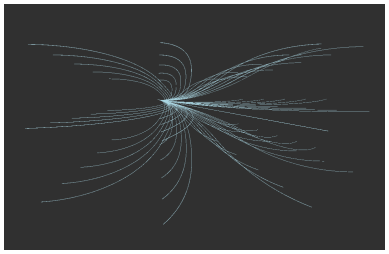


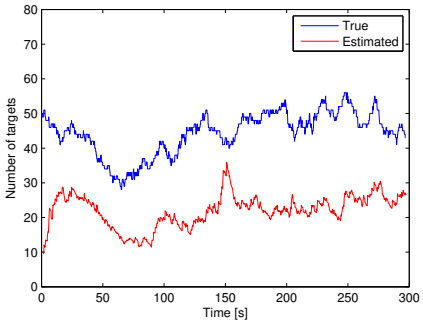
Fig. 4 Sample UAV trajectories.

We use a simple model for a fixed-wing aircraft with three basic control inputs: forward velocity, yaw rate, and pitch rate. For each control input we select a range of possible values. For each possible set of control inputs we integrate the position, yaw, and pitch forward in time using a 1-step Euler integration scheme. Any trajectories that bring the robots above or below the elevation limits are discarded as invalid, as are any that result in collision. The remaining trajectories are interpolated to yield the T poses at which each robot will take a measurement. Figure 4 shows sample trajectories for a single robot.

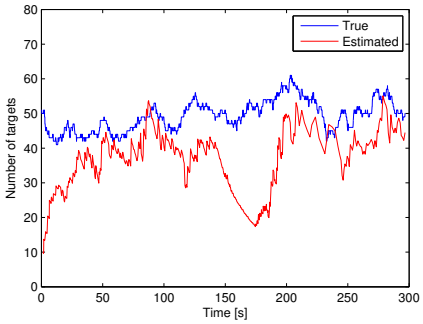
4 RESULTS

To test the performance of the different objective functions, we ran a series of experiments using simulated robots, varying the number of robots, the length of the planning horizon, the objective function, and the target motion model. We used teams of 2, 4, and 6 robots, keeping the number of planning steps constant ($T=2$) and keeping the total number of actions for the team constant ($RT=12$). We perform 5 trials with each configuration, randomly selecting a subset of 80 targets from the taxi database for the ground truth target motion in each trial. Note that the true number of targets in the area of interest varies over time as targets enter and leave. The robots monitor the area from Fig. 2a, which is scaled down by a factor of 100 from the real world. We also sped up the data by a factor of 60, so 1 s in simulation represents 1 min of real time, in order to speed up the simulations. The data is taken from 4–9 pm on May 18, 2008, a time of day where there will be plenty of taxi traffic.

It is worth noting that a number of competing multi-target tracking methods exist [29], most notably the Multiple Hypothesis Tracker (MHT) and Joint Probabilistic Data Association (JPDA). However, to the best of our knowledge, there do not exist active multi-robot control policies based on these estimation algorithms. This makes comparisons to these methods beyond the scope of this paper.

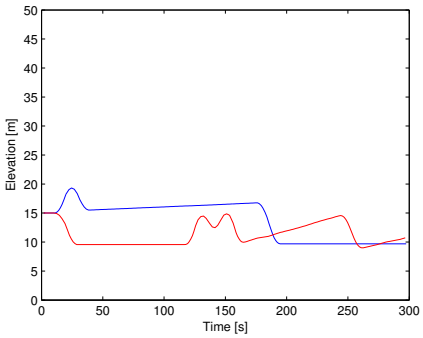


(a) GP motion model with MI objective

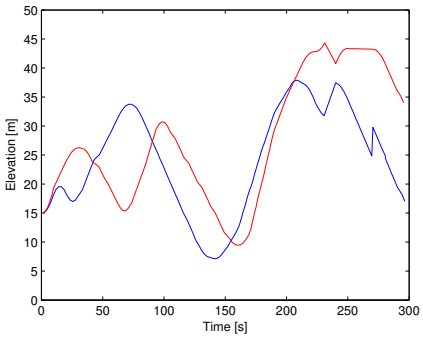


(b) GP motion model with END objective

Fig. 5 Ratio of the expected number to the true number of targets over a single run for $R = 2$ and $T = 6$.



(a) GP motion model with MI objective



(b) GP motion model with END objective

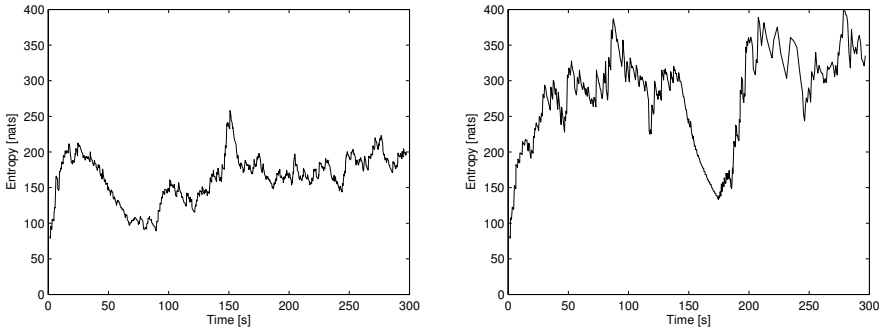
Fig. 6 The elevation of the robots over a single run for $R = 2$ and $T = 6$.

4.1 Moving Targets

The two target motion models that we consider are the Gaussian Process (GP) described in Sec. 3.2, and a Gaussian random walk (GRW) model. In GRW we model the target as performing a random walk, with a velocity drawn at random from a Gaussian distribution. Note that these models are used only to update the PHD; the actual targets trajectories are given by the taxi dataset. In both cases we use the survival and birth processes described in Sec. 3.2, with the birth rate set to 0.6788 to account for the reduced number of data files used.

Figures 5–7 show how the ratio of the expected number of targets to true targets, the robot elevation, and the target set entropy change during a single run. These are representative trials of a team of 2 robots with a planning horizon of 6 time steps.

Figures 8–10 show the ratio of the expected number of targets to true number of targets, the fraction of true targets within the sensor FoV, and the ratio of expected



(a) GP motion model with MI objective **(b)** GP motion model with END objective

Fig. 7 The entropy of the target set over a single run for $R = 2$ and $T = 6$.

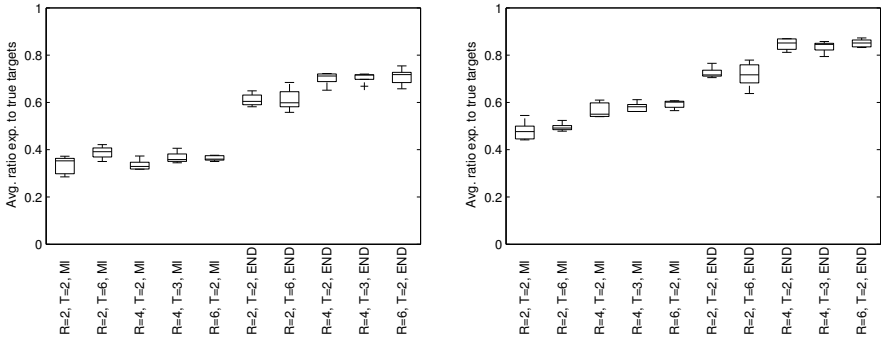
targets to true targets within the sensor FoV, respectively. In general, the fraction of targets tracked by the team depends much more on the motion model and the objective function than on the team size or planning horizon, despite the fact that larger teams and planning horizons cause the robots to observe a larger number of targets. Additionally, the ratio of the expected number of targets to the true number of targets within the sensor FoV is largely independent of the objective function, team size, or planning horizon.

Overall, the robot teams using the information based control objective (MI) estimate and track fewer targets than the teams using the END objective but each target is tracked with higher quality. Additionally, the teams using the GP-based motion model track more targets than those using the GRW motion model.

The reason for this difference is due to the emergent behavior of the different control objectives. Robots using the MI objective tend to stay closer to the ground in order to decrease uncertainty in the location of individual targets. On the other hand, robots using the END objective fly at a higher altitude, as Fig. 11 shows. Note that increasing the altitude decreases the probability of detection, while increasing the sensor FoV. Consequently, flying to the highest altitude is not necessarily optimal. Figure 12 shows the average target entropy. This is substantially lower for the teams using MI, indicating that the targets are being tracked with less uncertainty.

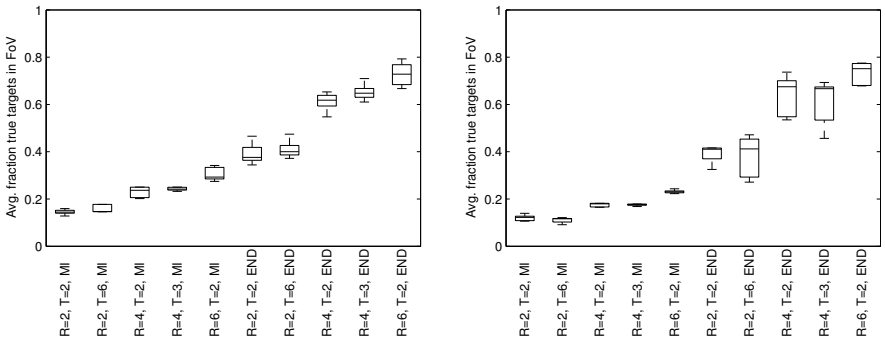
4.2 Static Targets

We also test the performance of our framework with static targets using a team of 4 robots with a planning horizon of 3. The simulation parameters are identical, except we replace the 80 taxi data traces with 80 randomly drawn static target locations. The resulting final estimated number of targets and target entropies are shown in Fig. 13. The final estimated number of targets is very close to 1 using both objective functions, indicating that the system is able to correctly determine the number of targets. The entropy is also lower than in the case of moving targets.



(a) Gaussian random walk **(b)** Gaussian process

Fig. 8 Average ratio of the expected number of targets to the true number of targets over a single run.

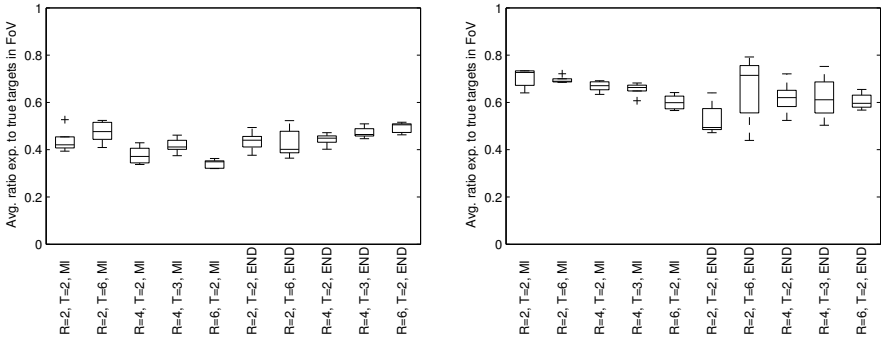


(a) Gaussian random walk **(b)** Gaussian process

Fig. 9 Average fraction of the number of true targets within the team's field of view over a single run.

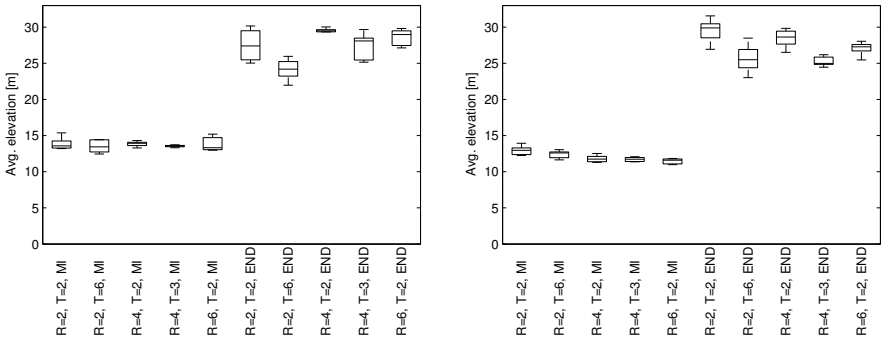
5 CONCLUSIONS

In this paper we describe a framework for detecting, localizing, and tracking an unknown number of moving targets using a team of mobile robots. The robot team uses the Probability Hypothesis Density filter to simultaneously estimate the number of targets and the states of the targets. The PHD filter is robust to false negative and false positive detections and sensor noise and does not require any explicit data association. Using the estimate of the target set from the PHD filter, the robots greedily select actions that maximize submodular control objectives. The two control objectives that we consider in this paper are the expected number of detected (END) targets by the team and the mutual information (MI) between the predicted targets and the future detections of the robots. We validate our framework through extensive simulations using a real-world dataset for target motion. Robot teams using the



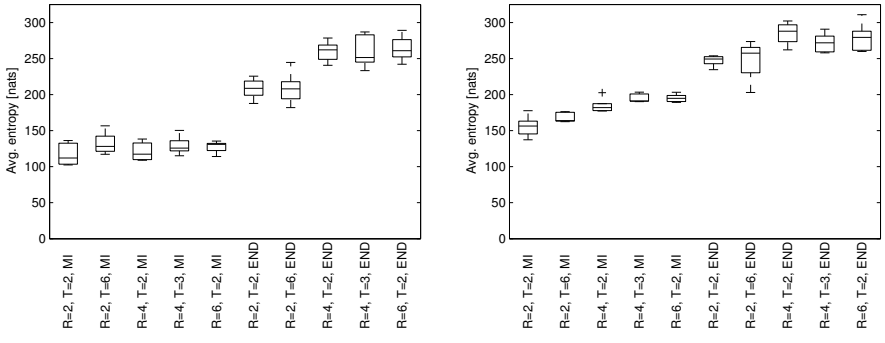
(a) Gaussian random walk **(b)** Gaussian process

Fig. 10 Average ratio of the expected number of targets to the true number of targets within the team's field of view over a single run.



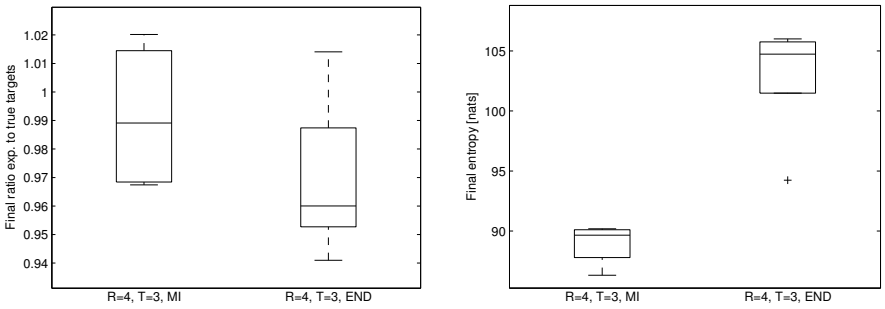
(a) Gaussian random walk **(b)** Gaussian process

Fig. 11 Average elevation of the robots over a single run.



(a) Gaussian random walk **(b)** Gaussian process

Fig. 12 Average entropy of the target set over a single run.



(a) Final ratio of the expected number of targets to the true number of targets.

(b) Final entropy of the estimated target set.

Fig. 13 Performance of our framework with static targets.

END objective track a higher fraction of the targets but do not localize the targets with high precision. Conversely, robot teams using MI track a smaller number of targets but have significantly lower uncertainty in the target positions.

Acknowledgements This work was funded by ONR MURI Grants N00014-07-1-0829, N00014-09-1-1051, and N00014-09-1-1031, ARO Grant W911NF-13-1-0350, NSF Grant IIS-1426840, and TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA. P. Dames was supported by the Department of Defense through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program.

References

1. M. Adams, B.-N. Vo, and R. Mahler, editors. *Advances in Probabilistic Modeling: Applications of Stochastic Geometry*, volume 21(2) of *IEEE Robot. Autom. Mag.* IEEE, June 2014.
2. N. Atanasov, M. Zhu, K. Daniilidis, and G. J. Pappas. Semantic Localization Via the Matrix Permanent. In *Robotics: Science and Systems*, 2014.
3. G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák. Maximizing a submodular set function subject to a matroid constraint. In *Integer programming and combinatorial optimization*, pages 182–196. Springer, 2007.
4. B. Charrow, N. Michael, and V. Kumar. Active control strategies for discovering and localizing devices with range-only sensors. In *Workshop on Algorithmic Foundations in Robotics (WAFR)*, 2014.
5. T. H. Chung, G. A. Hollinger, and V. Isler. Search and pursuit-evasion in mobile robotics. *Autonomous robots*, 31(4):299–316, 2011.
6. T. Cover and J. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.
7. P. Dames and V. Kumar. Autonomous Localization of an Unknown Number of Targets without Data Association Using Teams of Mobile Sensors. *IEEE Trans. Autom. Sci. Eng.*, 12(3):850–864, 2015.
8. E. W. Frew and S. M. Rock. Trajectory generation for constant velocity target motion estimation using monocular vision. In *IEEE Intl. Conf. on Rob. and Autom.*, pages 3479–3484, 2003.
9. T. Furukawa, F. Bourgault, B. Lavis, and H. F. Durrant-Whyte. Recursive Bayesian search-and-tracking using coordinated UAVs for lost targets. In *IEEE Intl. Conf. on Rob. and Autom.*, pages 2521–2526, 2006.

10. N. R. Gans, G. Hu, K. Nagarajan, and W. E. Dixon. Keeping multiple moving targets in the field of view of a mobile camera. *IEEE Trans. Robot.*, 27(4):822–828, 2011.
11. H. Grabner, T. T. Nguyen, B. Gruber, and H. Bischof. On-line boosting-based car detection from aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(3):382–396, 2008.
12. G. A. Hollinger, J. Djugash, and S. Singh. Target tracking without line of sight using range from radio. *Autonomous Robots*, 32(1):1–14, 2012.
13. J. Joseph, F. Doshi-Velez, A. S. Huang, and N. Roy. A bayesian nonparametric approach to modeling motion patterns. *Autonomous Robots*, 31(4):383–400, 2011.
14. C.-Y. Kim, D. Song, Y. Xu, J. Yi, and X. Wu. Cooperative search of multiple unknown transient radio sources using multiple paired mobile robots. *IEEE Trans. Robot.*, 30(5):1161–1173, Oct 2014.
15. D. Kotz and T. Henderson. CRAWDAD: A community resource for archiving wireless data at dartmouth. *IEEE Pervasive Comput.*, 4(4):12–14, 2005.
16. A. Krause and C. E. Guestrin. Near-optimal Nonmyopic Value of Information in Graphical Models. *Conference on Uncertainty in Artificial Intelligence*, 2005.
17. C. Kreucher, K. Kastella, and A. O. Hero III. Sensor management using an active sensing approach. *Signal Processing*, 85(3):607–624, 2005.
18. K. Leung, F. Inostroza, and M. Adams. Evaluating Set Measurement Likelihoods in Random-Finite-Set SLAM. In *IEEE Intl. Conf. on Inf. Fusion*, pages 1–8. IEEE, 2014.
19. X. Li, R. Lu, X. Liang, X. Shen, J. Chen, and X. Lin. Smart community: an internet of things application. *IEEE Commun. Mag.*, 49(11):68–75, 2011.
20. X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking. part i. dynamic models. *IEEE Trans. Aerosp. Electron. Syst.*, 39(4):1333–1364, 2003.
21. A. Logothetis, A. Isaksson, and R. J. Evans. An information theoretic approach to observer path design for bearings-only tracking. In *IEEE Intl. Conf. on Decision and Control*, volume 4, pages 3132–3137, 1997.
22. R. Mahler. Multitarget bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.*, 39(4):1152–1178, Oct. 2003.
23. M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser. CRAWDAD data set epfl/mobility (v. 2009-02-24). Downloaded from http://crawdad.org/epfl/mobility/, Feb. 2009.
24. C. Rasmussen and C. Williams. Gaussian processes for machine learning. *Gaussian Processes for Machine Learning*, 2006.
25. S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer. The labeled multi-bernoulli filter. *IEEE Trans. Signal Process.*, 62(12):3246–3260, 2014.
26. B. Ristic, B.-N. Vo, and D. Clark. A note on the reward function for PHD filters with sensor control. *IEEE Trans. Aerosp. Electron. Syst.*, 47(2):1521–1529, 2011.
27. D. Song, C.-Y. Kim, and J. Yi. Simultaneous localization of multiple unknown and transient radio sources using a mobile robot. *IEEE Trans. Robot.*, 28(3):668–680, 2012.
28. J. R. Spletzer and C. J. Taylor. Dynamic sensor planning and control for optimally tracking targets. *Intl. J. Robot. Research*, 22(1):7–20, 2003.
29. L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell. *Bayesian Multiple Target Tracking*. Artech House, 2013.
30. P. Tokekar, V. Isler, and A. Franchi. Multi-target visual tracking with aerial robots. In *IEEE/RSJ Intl. Conf. on Intell. Robots and Syst.*, pages 3067–3072. IEEE, 2014.
31. B.-N. Vo, S. Singh, and A. Doucet. Sequential monte carlo methods for multi-target filtering with random finite sets. *IEEE Trans. Aerosp. Electron. Syst.*, 41(4):1224–1245, Oct. 2005.
32. Z. Xu, R. Fitch, J. Underwood, and S. Sukkarieh. Decentralized coordinated tracking with mixed discrete–continuous decisions. *Journal of Field Robotics*, 30(5):717–740, 2013.
33. T. Zhao and R. Nevatia. Car detection in low resolution aerial images. *Image and Vision Computing*, 21(8):693–703, 2003.
34. K. Zhou and S. I. Roumeliotis. Optimal motion strategies for range-only constrained multisensor target tracking. *IEEE Trans. Robot.*, 24(5):1168–1185, 2008.
35. K. Zhou and S. I. Roumeliotis. Multirobot active target tracking with combinations of relative observations. *IEEE Trans. Robot.*, 27(4):678–695, 2011.