

Approximate Representations for Multi-Robot Control Policies that Maximize Mutual Information

Benjamin Charrow
GRASP Lab
University of Pennsylvania
bcharrow@seas.upenn.edu

Vijay Kumar
GRASP Lab
University of Pennsylvania
kumar@seas.upenn.edu

Nathan Michael
The Robotics Institute
Carnegie Mellon University
nmichael@cmu.edu

Abstract—We address the problem of controlling a small team of robots to estimate the location of a mobile target using non-linear range-only sensors. Our control law maximizes the mutual information between the team’s estimate and future measurements over a finite time horizon. Because the computations associated with such policies scale poorly with the number of robots, the time horizon associated with the policy, and typical non-parametric representations of the belief, we design approximate representations that enable real-time operation. The main contributions of this paper include the control policy, an algorithm for approximating the belief state with provable error bounds, and an extensive study of the performance of these algorithms using simulations and real world experiments in complex, indoor environments.

I. INTRODUCTION

Several important applications of robotics like environmental monitoring, cooperative mapping, and search and rescue, require information to be gathered quickly and efficiently. In this work, we consider a related information gathering problem where a team of mobile robots must estimate the location of a non-adversarial mobile target using range-only sensors. Given the target’s mobility and the limited information range-only sensors provide, the team must quickly determine which movements will yield useful measurements.

These problems can be thought of as active perception problems where one must find a control policy that is optimal in an information-theoretic sense [6, 19]. Maximizing mutual information has been a particularly successful approach [12, 18, 7, 1, 16]. Most mutual information based control laws are computationally tractable because they are greedy maximizations over a single time step or are used in offline or single robot scenarios. However, we cannot make any of these assumptions in our problem. Given that the target is mobile the team cannot generate plans offline. The team must coordinate their movements because of the limited information that range measurements provide about the target’s state. Further, maximizing mutual information over a finite time horizon involves minimizing a cost functional over the set of possible trajectories, a problem that is computationally intractable. Even if the time horizon and the trajectory are discretized, calculating mutual information requires integration over the joint measurement space of the team, which grows with the time horizon of the plan and size of the team.

To address these difficulties we develop a new approximation and theoretical design criterion which enables us to build a real-time mutual information controller for our mobile target tracking problem. A primary contribution is a method to approximate mutual information by approximating the distribution over the target’s predicted location. While this approximation necessarily introduces error into the control law, we show that this error is bounded for Gaussian measurement models. Importantly, this bound yields insight into the degree to which we can speed up the calculation of mutual information via approximation without significantly affecting the team’s performance. The other primary contribution is a bound on the difference in mutual information between two control inputs as a function of the noise of the range sensors. This analysis further aids in the design of a real-time system by enabling us to appropriately design the set of control inputs.

We evaluate the proposed methodology and design trade-offs through simulations and real world experiments. Our results show that the approximation significantly decreases the time to compute mutual information and enables the team to quickly and successfully estimate the mobile target’s position.

In this work, we focus on range-only RF-based sensors as we are interested in addressing the problem of target localization and tracking in search and rescue scenarios where the location and motion of targets are unknown, but the target is equipped with an RF ranging device (as found in many modern mobile phones [5]). As we will show, these sensors enable operation in a variety of environments without requiring line-of-sight to the target, which is often the case in search and rescue scenarios. However, as the proposed mutual information based techniques are broadly applicable, the presentation of these methods in Sect. IV remains general.

II. RELATED WORK

There is extensive work on information based control. Grocholsky [6] and Stump et al. [19] developed controllers to localize static targets by maximizing the rate of change of the Fisher information matrix, but their work assumes a Gaussian belief. Hoffmann and Tomlin [7] maximized mutual information and used a particle filter for the belief, but limited coordination between robots to keep the observation space small. We previously built on these ideas by using an algorithm of Huber et al. [11] to approximate entropy [1].

TABLE I
NOTATION

Symbol	Description
x	Mobile target's 2D position
z	Range measurement
t	Current time
τ	Future time interval
c_τ	Trajectory for the team to follow
T	Length of future time interval
R	Size of team
M	Num. of particles for target prediction

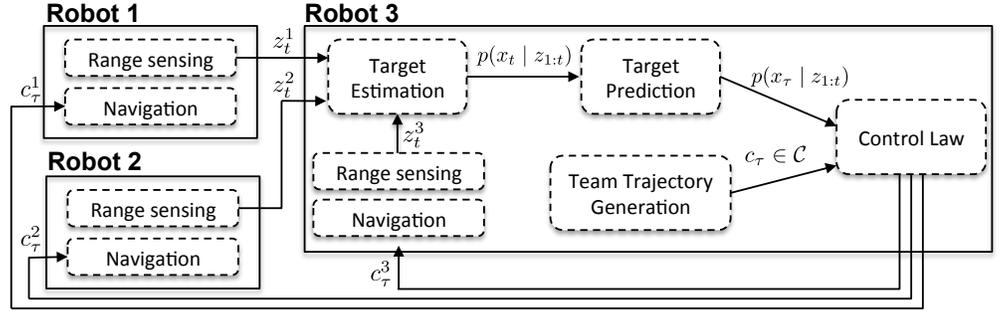


Fig. 1. System overview with three robots. In this example Robot 3 aggregates range measurements and estimates the location of the target. This estimate is used to predict future locations of the target over several time steps. The control law selects the trajectory of the team that maximizes the mutual information between the target's predicted location and measurements the team expects to make. This trajectory is sent to the other robots which follow it.

Julian et al. [12] and Singh et al. [18] used mutual information to estimate the state of static environments using multi-robot teams. Krause and Guestrin [13] derived performance guarantees for greedy maximizations of mutual information. However, these guarantees only hold in offline settings where new measurements are *not* incorporated as robots move. Ryan and Hedrick [16] minimized entropy over a receding horizon to track a mobile car with a fixed wing plane. We use dynamically simpler ground robots, but our approximations enable us to calculate control inputs for multiple robots in real-time.

The multi-robot range-only tracking problem we study is similar to probabilistic pursuit-evasion games [2]. The problem also relates to work by Hollinger et al. [8] on mobile target estimation with fixed and uncontrollable range radii. We focus on controlling mobile robots to localize a mobile target.

III. MUTUAL INFORMATION BASED CONTROL

In this section we describe a mutual information based control law for the mobile target tracking problem. We first detail how to estimate the target's current location and predict where it will go next, enabling the team to estimate what measurements they will receive at different future locations. Our control law uses these predictions to determine which movements of the team will result in the biggest decrease of the estimate's uncertainty. Figure 1 outlines the entire process and Tab. I describes all relevant variables.

Our control and estimation strategies are centralized and require communication throughout the team. This limitation is not significant for our problem as typical range sensors are RF-based. If the team was in an environment where they could not communicate, they could not make measurements.

A. Target Estimation

Range measurements can easily lead to non-trivial multi-hypothesis belief distributions [1, 4, 7] and so we estimate the target's 2D position at time t using a particle filter with the typical recursive Bayesian filter equation [20]:

$$bel(x_t) = \eta p(z_t | x_t) \int bel(x_{t-1}) p(x_t | x_{t-1}) dx_{t-1} \quad (1)$$

where η is a normalization constant. This makes the standard assumptions that the target's state is a Markov process and that measurements made at different points in time or by different robots are conditionally independent given the state.

Similar to other work on predicting the motion of an uncooperative target [16, 21] we use a Gaussian random walk for the process model: $p(x_t | x_{t-1}) = \mathcal{N}(x_t; x_{t-1}, \sigma^2 I)$ where x_{t-1} is the mean and $\sigma^2 I$ is the covariance.

We assume that the measurement model, $p(z_t | x_t)$, can be modeled as the true distance between the target and measuring robot perturbed by Gaussian noise. Previous experimental work by Djughash and Singh [4] and the authors [1] shows this assumption is reasonable when the target and robot have line of sight and that there are techniques for mitigating it when they do not.

B. Target and Measurement Prediction

To determine how they should move, the team must predict where the target will go and what measurements they will receive. Specifically, if the team plans a trajectory over the time interval $\tau \triangleq t+1 : t+T$, they need to estimate the target's future state: $x_\tau = [x_{t+1}, \dots, x_{t+T}]$. This can be done using the current belief and process model:

$$p(x_\tau | z_{1:t}) = \int p(x_t | z_{1:t}) \prod_{i=t+1}^{t+T} p(x_i | x_{i-1}) dx_t \approx \sum_{j=1}^M w_j \delta(x_\tau - \tilde{x}_\tau^j) \quad (2)$$

Where $\delta(\cdot)$ is the Dirac delta function, \tilde{x}_τ^j is the trajectory of the j^{th} particle, and w_j is its weight.

Over this time interval the team will receive range measurements $z_\tau(c_\tau) = [z_{t+1}(c_{t+1}), \dots, z_{t+T}(c_{t+T})]$ where z_k is the vector of measurements the team receives at time k and c_τ is the trajectory the team follows. The measurement density can be calculated by marginalizing over the state and applying the conditional independence assumption of the measurements:

$$p(z_\tau(c_\tau) | z_{1:t}) = \int p(x_\tau | z_{1:t}) \prod_{i=t+1}^{t+T} p(z_i(c_\tau) | x_i) dx_\tau \quad (3)$$

Because the measurement model is Gaussian, substituting (2) into (3) results in a Gaussian mixture model representation of the measurement density. If the team has R robots, then after applying the conditional independence assumption the j^{th} mixture component has a distribution equal to $\prod_{i=t+1}^{t+T} \prod_{r=1}^R p(z_i^r(c_i^r) | x_i = \tilde{x}_i^j)$ and weight w_j . Individual range measurements are 1-dimensional, which means the measurement space is RT -dimensional.

C. Information Based Control

Our control law maximizes the mutual information between the future state of the target and future measurements made by the team. By considering multiple measurements over time, the control law produces a better trajectory than a greedy maximization would. Formally, at time t the team selects the trajectory to follow over the time interval τ via the functional:

$$c_\tau^* = \arg \max_{c_\tau \in \mathcal{C}} \mathbf{H}[z_\tau(c_\tau)] - \mathbf{H}[z_\tau(c_\tau) | x_\tau] \quad (4)$$

where $\mathbf{H}[z_\tau(c_\tau)]$ is the differential entropy of the measurement density, $\mathbf{H}[z_\tau(c_\tau) | x_\tau]$ is the conditional differential entropy of the measurements given the target's true state, and their difference is mutual information [3]. Although it is not explicit, both entropies are conditioned on measurements made up to time t . The domain of this objective is \mathcal{C} , the set of trajectories the team can follow. Both measurement distributions and their entropies change as a function of where the team travels. We now discuss how to generate trajectories and evaluate the objective.

1) *Trajectory Generation*: The team makes measurements at discrete points in time and so we treat the set of trajectories, \mathcal{C} , as a sequence of configurations. An element of \mathcal{C} can be expressed as $c_\tau = [c_{t+1}, \dots, c_{t+T}]$ where c_k^r is the 2D position of robot r at time k (we ignore orientation as it does not affect range measurements). To generate these configurations we use motion primitives. For example, in an open environment the set of trajectories could be composed of straight line trajectories whose final destinations are uniformly spaced on a circle (Fig. 2a). When non-convex obstacles are present (e.g., in an office building), \mathcal{C} could be generated by selecting final destinations and determining intermediate points by interpolating along the shortest valid path (Fig. 2b).

2) *Calculating the Objective*: To evaluate the objective for a trajectory c_τ , we separately evaluate the conditional measurement entropy and measurement entropy.

Calculating the conditional entropy is straightforward. Using its definition and the assumed conditional independence of the measurements given the state:

$$\mathbf{H}[z_\tau(c_\tau) | x_\tau] \approx \sum_{i=1}^M w_i \sum_{j=t+1}^{t+T} \sum_{r=1}^R \mathbf{H}[z_j^r(c_j^r) | x_j = \tilde{x}_j^i] \quad (5)$$

M is the number of particles used to predict the target's future state (2) and the approximate equality is due to the particle representation. Because each measurement comes from a Gaussian distribution, the entropies in the summand can be calculated analytically, making the running time $\Theta(MRT)$.

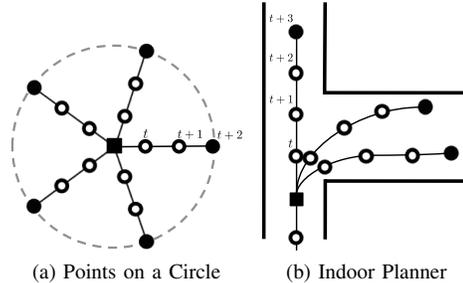


Fig. 2. Representative trajectories for a single robot using a finite set of motion primitives with a finite horizon. The black boxes are the starting locations, the black circles are the final destinations, and the hollow circles show intermediate points.

Unfortunately, as shown in Sect. III-B the measurement density, $p(z_\tau(c_\tau) | z_{1:t})$, is a Gaussian mixture model (GMM) and there is no analytic method for calculating its entropy. Numerical integration methods can be used, but the domain of integration is RT -dimensional, making it large even when there are a few robots over short time scales. This necessitates the use of algorithms like Monte Carlo integration which scale exponentially with the dimension, making them unsuitable for real-time performance.

However, there is a deterministic approximation for the entropy of a GMM developed by Huber et al. [11]. This algorithm uses a truncated Taylor series expansion of the logarithmic term in the entropy which allows the integration to be performed analytically. We use the 0th order approximation:

$$\mathbf{H}[g] \approx - \sum_{k=1}^M w_k \log \sum_{j=1}^M w_j \mathcal{N}(\mu_k; \mu_j, \Sigma_j) \quad (6)$$

where $g(x) = \sum_{i=1}^M w_i \mathcal{N}(x; \mu_i, \Sigma_i)$. While this approximation is not tight, our previous experimental work shows that it suffices for information based control [1].

The time complexity of (6) is $\Theta(M^2RT)$. The dependence on the number of particles which represent the target's future state, M , is problematic as it grows exponentially with the time horizon, T . While there are algorithms for reducing the number of components in a GMM, their time complexity is also quadratic in the number of components [15, 10]. In Sect. IV-B we describe a different approach which reduces the number of components by approximating the distribution over the *target's future state*, speeding up the entropy calculations without significantly affecting their value.

IV. APPROXIMATE REPRESENTATIONS

In Sect. III we outlined a general control law for a multi-robot team. However, it is computationally infeasible as the team's size and time horizon grow. It is also unclear which trajectories to consider. We address these problems in Sect. IV-B where we show how an approximation of the target's future state affects the objective function, and in Sect. IV-C where we develop a theoretical design criterion for generating trajectories. These analyses use an information theoretic relationship that we prove in Sect. IV-A.

A. Kullback-Leibler divergence and Entropy Difference

The Kullback-Leibler divergence (KL divergence or KLD) is one way of measuring the “distance” between two densities. For two densities p and q , it is defined as $\mathbf{KL}[p \parallel q] = \mathbf{E}_p[\log p(x)/q(x)]$. Intuitively, if two distributions are similar (i.e., their KL divergence is small) then their entropies (i.e., uncertainty) should also be similar. We show that this intuition is correct for Gaussian distributions and use that result to bound the difference of entropies between Gaussian mixture models (GMMs) with identically weighted components. For clarity of presentation, we defer all proofs to the Appendix.

The style of these information theoretic inequalities is not novel. Silva and Parada [17, Theorem 2] recently proved that in the limit as the KL divergence between bounded distributions goes to 0, their entropies become equal. Cover and Thomas [3, Theorem 17.3.3] showed that if the L_1 norm between discrete distributions is bounded, they have similar entropies. Research on rate distortion theory often looks at similar problems (e.g., how to find approximations of distributions without exceeding a certain distortion). However, we are not aware of prior work which directly relates KL divergence and entropy difference for continuous distributions with unbounded support. In the remainder of this section we also apply these bounds to our robotics problem, which we view as our primary contribution.

Lemma 1 (Entropy Upper Bound for Mixture Models):

If $f(x) = \sum_i \pi_i f_i(x)$ is a mixture model density then $\mathbf{H}[f] \leq \mathbf{H}[\pi] + \sum_i \pi_i \mathbf{H}[f_i]$ where $\mathbf{H}[\pi]$ is the discrete entropy of the mixture weights (Proved by Huber et al. [11]).

Lemma 2 (Entropy Lower Bound for Mixture Models): If

$f(x) = \sum_i \pi_i f_i(x)$ is a mixture model, $\mathbf{H}[f] \geq \sum_i \pi_i \mathbf{H}[f_i]$.

Lemma 3 (KLD Bound on Entropy for Gaussians): If

$f(x)$ and $g(x)$ are two k -dimensional Gaussian densities:

$$|\mathbf{H}[f] - \mathbf{H}[g]| < \min\{\mathbf{KL}[f \parallel g], \mathbf{KL}[g \parallel f]\} + \frac{k}{2}$$

Theorem 1 (KLD Bound on Entropy for GMMs): If

$f(x) = \sum_i \pi_i f_i(x)$ and $g(x) = \sum_i \pi_i g_i(x)$ are k -dimensional GMMs with M identically weighted components, then

$$|\mathbf{H}[f] - \mathbf{H}[g]| \leq \mathbf{H}[\pi] + \frac{k}{2} + \sum_{i=1}^M \pi_i \alpha_i$$

where $\alpha_i = \min\{\mathbf{KL}[f_i \parallel g_i], \mathbf{KL}[g_i \parallel f_i]\}$ and $\sum_i \pi_i = 1$.

B. Approximating Belief

As discussed in Sect. III-C, the number of particles needed to predict the location of the target grows exponentially in time. This complicates calculating the entropy of the measurement density, as it depends on the predictive distribution of the target (3). In this section we show that a simple approximation of the predictive distribution can speed up calculations of the objective (4), without significantly affecting its value.

Our approach for approximating a particle set is to replace subsets of similar particles with their average. Specifically, we partition the state space with a regular square grid and create an approximate particle set by replacing particles in the same cell with their weighted average. Figure 3 shows an example

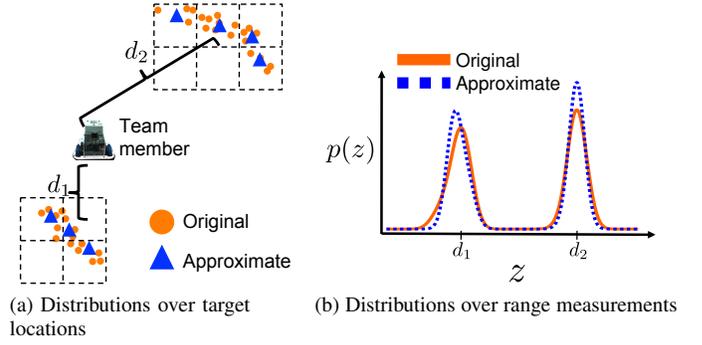


Fig. 3. (a) The original particle set (circles) can be approximated by a smaller set of particles (triangles) using a grid. (b) The different particle sets result in similar distributions over range measurements (e.g., peaks at d_1 and d_2)

of this process. In Fig. 3a the original particle set representing a multi-modal distribution over the target’s location is reduced via a grid. The new set is smaller, but it slightly changes the measurement distribution as shown in Fig. 3b.

To prove that this approximation introduces a bounded amount of error into the objective, we assume that the range based measurement model has a constant variance σ^2 and is unbiased. With these assumptions, the conditional entropy (5) is $\mathbf{H}[z_\tau(c_\tau) \mid x_\tau] \approx (TR/2) \log(2\pi e\sigma^2)$, which doesn’t depend on the number of particles or their location. Thus, the approximation does not introduce any error for this term.

The approximation does introduce error for the entropy of the measurement density, $\mathbf{H}[z_\tau(c_\tau)]$. To bound this error, we first determine the original and approximate measurement distributions. Let \mathcal{X} be the original set of particles and $f(z) = \sum_{j=1}^M w_j f_j(z)$ be the GMM it creates with the trajectory c_τ and measurement model. Additionally, let \mathcal{G}_i be the subset of \mathcal{X} which forms the i^{th} particle in the approximate particle set \mathcal{Y} . The density \mathcal{Y} creates is $g(z) = \sum_{i=1}^M \sum_{j \in \mathcal{G}_i} w_j g_i(z)$, where $g_i(z)$ is determined by the average of particles in \mathcal{G}_i . While $M \neq N$, the expressions for f and g have the same number of components with identical weights. However, the means of components f_j and g_i – which we denote as μ^{f_j} and μ^{g_i} – differ. Because measurements are unbiased, if c_k^r is robot r ’s position at time k and $\tilde{x}_k^{\mathcal{X}_j}$ is the position of the j^{th} particle in \mathcal{X} at time k , the corresponding element of μ^{f_j} is $\mu_{k,r}^{f_j} = \|c_k^r - \tilde{x}_k^{\mathcal{X}_j}\|_2$. Similarly, if $\tilde{x}_k^{\mathcal{Y}_i}$ is the particle in \mathcal{Y} that lies in the same cell as $\tilde{x}_k^{\mathcal{X}_j}$, then $\mu_{k,r}^{g_i} = \|c_k^r - \tilde{x}_k^{\mathcal{Y}_i}\|_2$. Because $\tilde{x}_k^{\mathcal{X}_j}$ and $\tilde{x}_k^{\mathcal{Y}_i}$ lie in the same cell, $\|\tilde{x}_k^{\mathcal{X}_j} - \tilde{x}_k^{\mathcal{Y}_i}\|_2 \leq \sqrt{2}L$ where L is the cell’s length. Applying the triangle inequality and summing over robots:

$$\|\mu_k^{f_j} - \mu_k^{g_i}\|_2^2 \leq 2RL^2$$

Substituting this into the expression for the KL divergence between Gaussians:

$$\mathbf{KL}[f_j \parallel g_i] = \sum_{k=t+1}^{t+T} \frac{\|\mu_k^{g_i} - \mu_k^{f_j}\|_2^2}{2\sigma^2} \leq \frac{TRL^2}{\sigma^2} \quad (7)$$

This bounds the KL divergence between components of $f(z)$ and $g(z)$, so Thm. 1 bounds the difference in entropies.

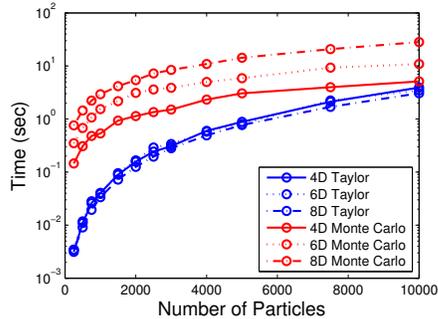


Fig. 4. Monte Carlo integration vs. 0th order Taylor approximation for evaluating mutual information. The Taylor approximation is more efficient.

Importantly, the bound shows that if the length of a grid cell is close to the standard deviation of the measurement model, then the error from the approximation will be negligible.

C. Selecting Motion Primitives

A central aspect of our approach is the use of motion primitives to generate trajectories. To select an informative trajectory quickly, we wish to generate as few as possible without ignoring important directions. In this section we prove that mutual information cannot change significantly when the team’s trajectory changes on the order of the standard deviation of the measurement model. This suggests that generating multiple trajectories that are close to one another is computationally wasteful. It also suggests that the intermediate waypoints from motion primitives (e.g., Fig. 2) should be spaced as a function of the standard deviation of the measurement model. We exploit both of these observations when designing the system.

Let c be a trajectory and $f(z) = \sum_i w_i f_i(z)$ be the measurement density it determines when combined with $p(z | x)$, and \mathcal{X} , the particle set representing the predicted motion of the target. If Δ is a perturbation to the team’s position at each point in time, $c + \Delta$ is a new trajectory with density $g(z) = \sum_i w_i g_i(z)$. \mathcal{X} is the same for c and $c + \Delta$, so $f(z)$ and $g(z)$ have the same number of components with identical weights. However, the means of the components (i.e., μ^{f_i} and μ^{g_i}) are not necessarily equal. Similar to before, $\mu_{k,r}^{f_i} = \|\tilde{x}_k^i - c_k^r\|_2$ and $\mu_{k,r}^{g_i} = \|\tilde{x}_k^i - (c_k^r + \Delta_k^r)\|_2$ where \tilde{x}_k^i is again the i^{th} particle in \mathcal{X} at time k . The triangle inequality tells us $|\mu_{k,r}^{f_i} - \mu_{k,r}^{g_i}| \leq \|\Delta_k^r\|_2$. Summing over all robots and time steps in the expression for KL divergence yields:

$$\text{KL}[f_i \parallel g_i] \leq \sum_{k=t+1}^{t+T} \sum_{r=1}^R \frac{\|\Delta_k^r\|_2^2}{2\sigma^2}$$

A direct application of Thm. 1 bounds the difference of entropies between f and g . As before, the conditional entropy in the objective is the same for both mixture models, and so no error is introduced in that term.

V. EXPERIMENTS

This section describes a series of simulations and real world experiments which evaluate the proposed methodology and design trade-offs. All of our software is written in C++.

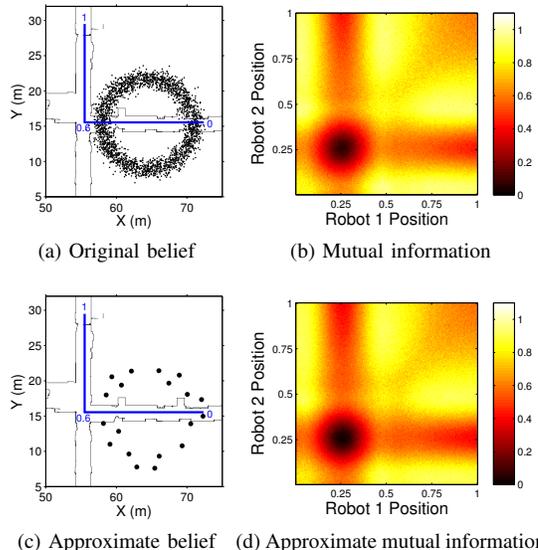


Fig. 5. Effect of approximating belief on mutual information as two robots move along a parameterized line from $[0,1]$ (blue, (a) and (c)). Approximating belief introduces limited error into the calculation of mutual information ((b) and (d)) according to the bound developed in Sect. IV.

A. Computational Performance

In Sect. III-C we discussed how the primary computational challenge of the objective is the calculation of the measurement density’s entropy. We now study the performance of Huber’s 0th order Taylor approximation and show that the approximation algorithm in Sect. IV-B makes it tractable for longer time horizons.

As a baseline, we compare the performance of the Taylor approximation to the Vegas algorithm – a Monte Carlo integration algorithm – on a variety of mixture models. To generate a mixture we sample particles and a team trajectory uniformly at random and then apply the measurement model. Figure 4 shows the time to calculate the entropy of a single mixture using different numbers of particles and measurement dimensions. Teams often have to evaluate dozens to hundreds of trajectories, so calculations longer than a fraction of a second make the control law infeasible. The data show that unlike Monte Carlo integration, the Taylor approximation is feasible in higher dimensional spaces when the number of particles is not large. Its lack of growth with dimension is not surprising as its dependence on dimension is from simple matrix operations.

Figure 5 depicts the effect of approximating the belief on the calculation of mutual information as two robots are constrained to move along a parameterized line from $[0, 1]$. Given a belief distribution represented by a particle filter ($M = 2500$, Fig. 5a), mutual information is computed for all possible locations of the team with range-only observations ($\sigma = 1.0$ m, Fig. 5b). The value of mutual information matches intuition; as the robots move away from the center of the belief (0.3 on the line), measurements become increasingly informative with corresponding increases in mutual information. An ap-

proximate belief representation with fewer particles ($M = 17$) corresponding to a grid length of 4.0 m (Fig. 5c) yields mutual information values that are similar to the original distribution. The mean absolute error (0.037 as depicted) is lower than Thm. 1 suggests, indicating that it can be a conservative bound.

B. Flexibility of Motion and Number of Robots

The team can presumably gather better measurements by evaluating more trajectories, but this adds computational expense, and additional trajectories may not be helpful. To examine these trade-offs in a controlled setting, we run simulations in an open environment and vary the number of robots in the team and destination points they use to generate trajectories (see Fig. 2a). We evaluate 1) how quickly the team obtains a stable and accurate estimate and 2) how accurate the estimate is over longer periods of time.

For each of these experiments the team moves at 0.2 m/s while the target moves at 0.15 m/s. The team plans a 15 second trajectory – simulating 3 target steps – and evaluates mutual information at 3 points along each trajectory. All members of the team start in the same location with the target 10 m to their right. The target repeatedly drives straight for 5 m and then selects a new direction uniformly at random. To make meaningful comparisons across experiments, the target always follows the same random path. The simulator generates range measurements between the target and each team member at 10 Hz using a Gaussian whose mean and variance are equal to the true distance. The simulator is asynchronous; the target moves even if the team is planning.

To ensure that the motion model of the filter (1) generates samples in all areas where the target could be, we set the standard deviation to be $\sigma = 0.3$, a small multiple of the maximum speed of the target multiplied by the sampling interval. To predict the target’s future location (2), the team simulates a step every 5 seconds with $\sigma = 0.25$.

The team “acquires” the target once they sustain an estimate for 10 seconds with a mean error below 1.5 m and a *spread* (i.e., the square root of the determinant of the covariance) below 1.5. We chose these values as they are close to the best the estimator can do. Table II shows acquisition times for 5 trials with a variable number of robots and destination points. Surprisingly, the 3 robot team does worse with more destination points. This is a direct consequence of the exponential growth in the number of trajectories the control law evaluates, which makes the robots spend a long time planning their initial trajectories. In contrast, the 2 robot team’s acquisition time improves by considering more than 3 destination points, after which point it does not change. Planning time is not an issue for the 2 robot team – they have at most 36 trajectories – which suggests that they are primarily limited by their team size. This is emphasized by the 10 second difference in performance between the best 2 robot time and the best 3 robot time.

To assess the long term performance of the team, we ran the same set of experiments for 5 minutes. Figure 6a shows the estimate’s error over time. While the target is acquired at different points in time, the final error is the same

TABLE II
EFFECT OF MOTION PRIMITIVES AND TEAM SIZE ON THE TIME TO ACQUIRE THE TARGET. MEAN AND STD. DEV. ARE SHOWN FOR 5 TRIALS.

	2 Robots	3 Robots
3 Points	80.55s (22.68)	42.57s (3.55)
4 Points	56.00s (6.07)	49.00s (15.24)
5 Points	56.17s (7.31)	51.83s (15.94)
6 Points	55.10s (12.24)	69.71s (9.94)

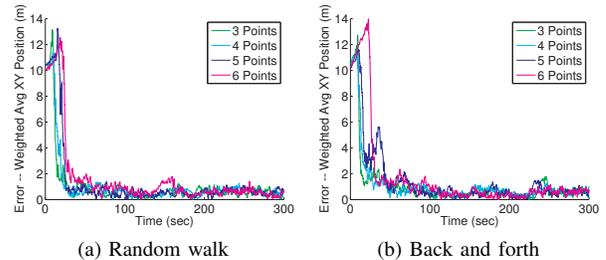


Fig. 6. Mean error of estimate for various motion primitives with 3 robots. The number of primitives do not affect the long term error regardless of how the target moves.

for all parameters. The computational difficulties that teams with more trajectories have are not present once the filter has converged, as the belief can be represented with fewer particles. Figure 6b shows the long term error from a separate experiment where the target moves 15 m to the right of its starting location and then back. This motion makes it harder to obtain and maintain an accurate estimate because the target moves away from the team for a long time without periodically stopping and turning. The similar results across experiments indicate that our estimation and control approach work well.

C. Indoor Experiment

To study the real world performance of a team we run two real world experiments where two mobile robots track a third mobile robot as it moves in a loop through an office building as shown in Fig. 8. The target’s path is 55 m long and it moves at 0.12 m/s, traversing the loop in about 8 minutes. The team starts across the map, travels at 0.2 m/s, and plans using destination points that are 6.0m away. Each robot is equipped with a Hokuyo-URG04LX laser scanner for localization, a 802.11s wireless mesh card for communication, and a range sensor that is commercially available as part of the nanoPAN 5375 Development Kit [14]. The nanoPAN 5375’s measurements often have an error larger than 2.0m [1].

In our first experiment, the target traverses its loop once. Figure 9a shows the error of the estimate over 7 trials. Overall, the team is able to obtain a good estimate quickly and maintain it. The mean root mean square error (RMSE) from $t = 60$ s to the end of the experiment was 2.43 m with a standard deviation of 0.68. The authors previously achieved errors of 1.0-2.0 m when localizing *stationary* nodes with the same sensor [1]. Given that the target is moving in this experiment, we believe these results indicate that the team performs well.

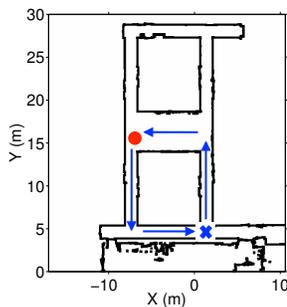


Fig. 8. Indoor experimental setup. The target starts at the blue “X” and moves in a loop. The team starts at the red circle.

In our second experiment, we assess the longer term performance of our approach by having the target traverse the same loop 4 times. Figure 9b shows the error of 2 different trials and Fig. 7 shows the filter’s evolution in trial 1. Again, the team does a good job tracking the target and obtains RMSEs of 3.88 m and 2.83 m. These slightly higher error rates are due to multiple hypotheses that sometimes arise as a result of the limited information range measurements provides. Given the limited travel directions in this environment, it is also difficult for the team to obtain measurements that remove ambiguities. Figure 7d and Fig. 9b at $t \approx 1000$ s show this happening. However, whenever the error increases, it eventually decreases. It is also important to note that when the estimate’s error increases, its covariance also increases; the rise in error is not indicative of the team being overly-confident of a bad estimate.

VI. CONCLUSION

In this paper we present a control policy for maximizing mutual information over a finite time horizon to enable a team of robots to estimate the location of a mobile target using range-only sensors. To enable calculation of our policy in real-time, we develop an approximation of the belief. By proving a relationship between the Kullback-Leibler divergence of certain distributions and their entropies, we show that the errors introduced by our approximation are bounded for range based Gaussian measurement models. We also develop a theoretical design criterion for generating motion primitives by connecting the incremental motions the team makes to the variance of the measurement model. We validated our approach using simulations and real world experiments in which a team of robots successfully track a mobile target.

VII. ACKNOWLEDGMENTS

We gratefully acknowledge the support of ONR Grant N00014-07-1-0829, ARL Grant W911NF-08-2-0004, and AFOSR Grant FA9550-10-1-0567. Benjamin Charrow was supported by a NDSEG fellowship from the Department of Defense.

REFERENCES

[1] Benjamin Charrow, Nathan Michael, and Vijay Kumar. Cooperative multi-robot estimation and control for radio

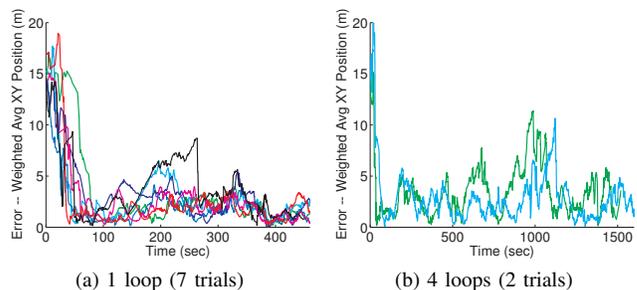


Fig. 9. Distance from the estimate’s mean to the target’s true location as the target drives around.

source localization. *Int. J. Robot. Research*. Special Issue from ISER 2012, to appear.

[2] Timothy Chung, Geoff Hollinger, and Volkan Isler. Search and pursuit-evasion in mobile robotics. *Autonomous Robots*, 31(4):299–316, 2011.

[3] Thomas M. Cover and Joy. A. Thomas. *Elements of Information Theory*. Wiley Online Library, 2004.

[4] Joseph Djughash and Sanjiv Singh. Motion-aided network slam with range. *Int. J. Robot. Research*, 31(5):604–625, 2012.

[5] Jun geun Park, Ben Charrow, Jonathan Battat, Dorothy Curtis, Einat Minkov, Jamey Hicks, Seth Teller, and Jonathan Ledlie. Growing an organic indoor location system. In *Proc. Intl. Conf. on Mobile Systems, Applications, and Services*, San Francisco, CA, June 2010.

[6] Ben Grocholsky. *Information-theoretic control of multiple sensor platforms*. PhD thesis, Australian Centre for Field Robotics, 2002.

[7] Gabriel Hoffmann and Claire Tomlin. Mobile sensor network control using mutual information methods and particle filters. *IEEE Trans. Autom. Control*, 55(1):32–47, 2010.

[8] Geoff Hollinger, Joseph Djughash, and Sanjiv Singh. Target tracking without line of sight using range from radio. *Autonomous Robots*, 32(1):1–14, 2011.

[9] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

[10] Marco Huber and Uwe Hanebeck. Progressive gaussian mixture reduction. In *Intl. Conf. on Inform. Fusion*, 2008.

[11] Marco Huber, Tim Bailey, Hugh Durrant-Whyte, and Uwe Hanebeck. On entropy approximation for gaussian mixture random vectors. In *Multisensor Fusion and Integr. for Intell. Syst.*, pages 181–188, Seoul, Korea, August 2008.

[12] Brian J. Julian, Michael Angermann, Mac Schwager, and Daniela Rus. A scalable information theoretic approach to distributed robot coordination. In *Proc. of the IEEE/RSJ Int. Conf. on Intell. Robots and Syst.*, pages 5187–5194, San Francisco, USA, September 2011.

[13] Andreas Krause and Carlos Guestrin. Near-optimal nonmyopic value of information in graphical models. In *Uncertainty in AI*, pages 324–331, 2005.

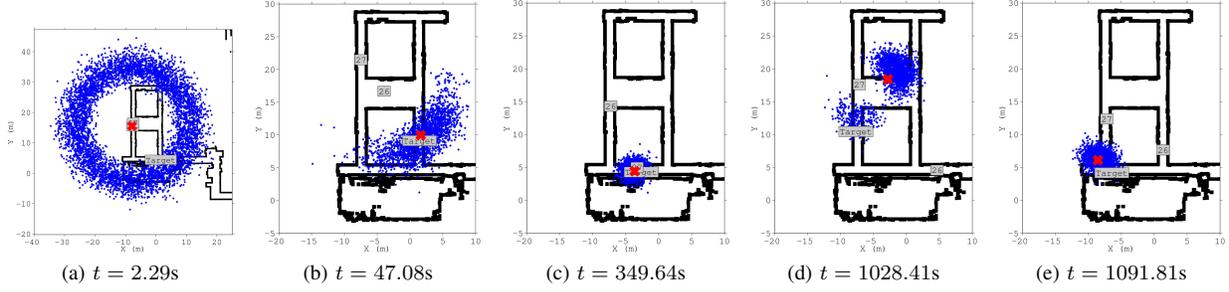


Fig. 7. Evolution of particle filter for trial 1 of the 4 loop experiment. Blue dots represent particles and the red “X” shows the filter’s mean. Gray boxes with numbers show the team’s position. a) Initially, the target’s location is uncertain. b-c) By moving intelligently, the team accurately estimates the target’s location. d-e) Multiple hypotheses can arise due to the target’s motion and the limited information of range measurements, but they are eventually eliminated.

- [14] nanoPAN 5375 Development Kit. http://www.nanotron.com/EN/PR_tools.php#03, January 2013.
- [15] Andrew Runnals. Kullback-leibler approach to gaussian mixture reduction. *IEEE Trans. on Aero. and Elec. Sys.*, 43(3):989–999, July 2007.
- [16] Allison Ryan and J. Karl Hedrick. Particle filter based information-theoretic active sensing. *Robot. and Autonomous Syst.*, 58(5):574–584, May 2010.
- [17] Jorge F. Silva and Patricio Parada. Sufficient conditions for the convergence of the shannon differential entropy. In *IEEE Info. Theory Workshop*, pages 608–612, Paraty, Brazil, October 2011.
- [18] Amarjeet Singh, Andreas Krause, Carlos Guestrin, and William J. Kaiser. Efficient informative sensing using multiple robots. *J. of AI Research*, 34(1):707–755, 2009.
- [19] Ethan Stump, Vijay Kumar, Ben Grocholsky, and Pedro Shiroma. Control for localization of targets using range-only sensors. *Int. J. Robot. Research*, 28(6):743, 2009.
- [20] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT Press, 2008.
- [21] René Vidal, Omid Shakernia, H. Jin Kim, David Shim, and Shankar Sastry. Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation. *IEEE Trans. on Robot. and Autom.*, 18(5):662–669, 2002.

APPENDIX

A. Proof of Lem. 1 (Entropy Upper Bound for Mixture Models)

See Huber et al. [11, Theorem 3]. They use Gaussian densities, but their proof works for general mixture models.

B. Proof of Lem. 2 (Entropy Lower Bound for Mixture Models)

Starting from the differential entropy of f and expanding:

$$\begin{aligned} \mathbf{H}[f] &= \int \sum_i \pi_i f_i(x) \log \frac{1}{\sum_j \pi_j f_j(x)} dx \\ &= \sum_i \pi_i \int \left(f_i(x) \log \frac{f_i(x)}{\sum_j \pi_j f_j(x)} + f_i(x) \log \frac{1}{f_i(x)} \right) dx \\ &= \sum_i \pi_i (\mathbf{KL}[f_i || f] + \mathbf{H}[f_i]) \end{aligned} \quad (8)$$

$\mathbf{KL}[f_i || f]$ is non-negative [3]; dropping it obtains the bound.

C. Proof of Lem. 3 (KLD Bound on Entropy for Gaussians)

Let $f(x) = \mathcal{N}(x; \mu_f, \Sigma_f)$ and $g(x) = \mathcal{N}(x; \mu_g, \Sigma_g)$. The entropy of a Gaussian with covariance Σ is $1/2 \log((2\pi e)^k \det \Sigma)$ and so:

$$\mathbf{H}[f] - \mathbf{H}[g] = \frac{1}{2} \log \det \Sigma_g^{-1} \Sigma_f \quad (9)$$

Both Σ_g^{-1} and Σ_f are symmetric positive definite (SPD) so all the eigenvalues of $\Sigma_g^{-1} \Sigma_f$ are strictly positive [9, Theorem 7.6.3]. Defining $\lambda_i > 0$ as the i^{th} eigenvalue of $\Sigma_g^{-1} \Sigma_f$ we rewrite (9) as $1/2 \sum_i \log \lambda_i$.

The KL divergence between f and g bounds this quantity. Starting from the closed form expression:

$$\begin{aligned} 2 \mathbf{KL}[f || g] + k &= \text{tr}(\Sigma_g^{-1} \Sigma_f) - \log \det(\Sigma_g^{-1} \Sigma_f) \\ &\quad + (\mu_g - \mu_f)^T \Sigma_g^{-1} (\mu_g - \mu_f) \\ &\geq \text{tr}(\Sigma_g^{-1} \Sigma_f) - \log \det(\Sigma_g^{-1} \Sigma_f) \\ &= \sum_i (\lambda_i - \log \lambda_i) \end{aligned} \quad (10)$$

The inequality holds because Σ_g^{-1} is SPD. Because $\log \lambda_i < \lambda_i - \log \lambda_i$ we know $\mathbf{KL}[f || g] + k/2$ upper bounds the difference of entropies. Similarly, we can apply $-(\lambda_i - \log \lambda_i) < \log \lambda_i$ to obtain a lower bound. Combining the bounds we have $|\mathbf{H}[f] - \mathbf{H}[g]| \leq \mathbf{KL}[f || g] + k/2$.

KL divergence is not symmetric [3], so we complete the proof by repeating the same steps for $\mathbf{KL}[g || f]$.

D. Proof of Thm. 1 (KLD Bound on Entropy for GMMs)

From Lems. 2 and 1:

$$\mathbf{H}[f] - \mathbf{H}[g] \leq \mathbf{H}[\pi] + \sum_{i=1}^M \pi_i (\mathbf{H}[f_i] - \mathbf{H}[g_i]) \quad (11)$$

Swapping the order of f and g and repeating the same steps results in a lower bound on the entropy difference:

$$-\mathbf{H}[\pi] + \sum_{i=1}^n \pi_i (\mathbf{H}[f_i] - \mathbf{H}[g_i]) \leq \mathbf{H}[f] - \mathbf{H}[g] \quad (12)$$

Taking absolute values of (11) and (12) and applying Lem. 3 completes the proof.