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

Benjamin Charrow
Univ. of Pennsylvania
bcharrow@seas.upenn.edu

Vijay Kumar
Univ. of Pennsylvania
kumar@seas.upenn.edu

Nathan Michael
Carnegie Mellon Univ.
nmichael@cmu.edu

July 1, 2014

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, and an extensive study of the performance of these algorithms using simulations and real world experiments in complex, indoor environments.

1 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 [12, 29]. Maximizing mutual information has been a particularly successful approach [19, 28, 14, 3, 26], as this policy seeks to gather measurements that will cause the greatest decrease in the uncertainty (i.e., entropy) of the estimate. 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

Table 1: Notation

Symbol	Description
x	Mobile target's 2D position
z	Range measurement
t	Current time
σ_m^2	Measurement noise
σ_p^2	Process noise
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

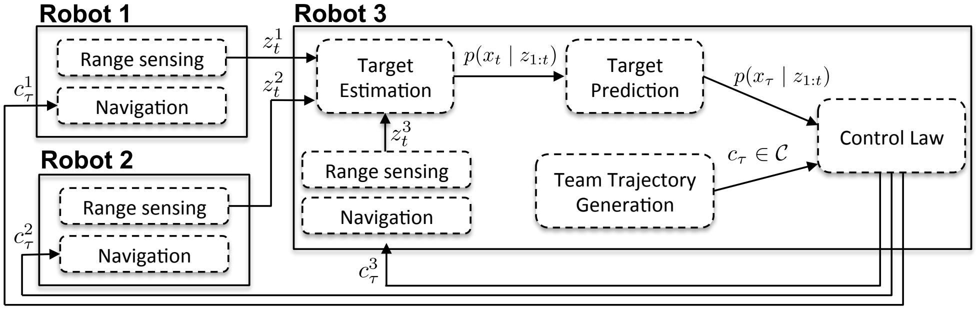


Figure 1: System overview with three robots. In this example, Robot 3 aggregates range measurements and estimates the location of the target. This estimate is used to predict the target's future locations over several time steps. The control law selects the trajectory that maximizes the mutual information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

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 [10]). 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. 4 remains general.

This article expands and refines our previous work on approximations for information-theoretic control with range-only sensors [2]. The primary addition in this work is a new theoretical bound on how the entropy of a Gaussian mixture model changes when the means of its components are perturbed (Sect. 4.1). This bound enables a better analysis of how approximating the belief of the target's location affects mutual information (Sect. 4.2) and the degree to which different control inputs will have different information content (Sect. 4.3). This paper also addresses an oversight in our previous work, in which we incorrectly argued that the approximations are guaranteed to introduce bounded error into the calculation of mutual information (Sect. 4.4). Finally, we present additional numerical studies to validate various approximations we make (Sect. 5).

2 Related Work

There is extensive work on information based control. Grocholsky [12] and Stump et al. [29] 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 [14] 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. [18] to approximate entropy [3]. Hollinger and Sukhatme [15] describe sampling based strategies for maximizing a variety of information metrics. Julian et al. [19] developed distributed algorithms for calculating mutual information to estimate the state of static environments using multi-robot teams. Ryan and Hedrick [26] 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. Recognizing that minimizing communication between robots can be important, Kassir et al. [20] developed a distributed information based control law to that explicitly modeled communication costs. Mutual information and entropy have also been applied to visual servoing problems [6] and active object modeling [32].

Krause and Guestrin [21] derived performance guarantees for greedy maximizations of mutual information and other submodular set functions. These results were applied to to mobile robot environmental monitoring problems by Singh et al. [28] and Binney et al. [1]. However, these performance guarantees only hold in offline settings where teams do not update their actions based on measurements they receive. Golovin and Krause [11] generalized these ideas to online settings, deriving performance guarantees for adaptive submodular functions. Unfortunately, mutual information is not an adaptive submodular function and the guarantees require all actions to be available to all robots at all times, making these results not directly applicable to this work.

The multi-robot range-only tracking problem we study is similar to probabilistic pursuit-evasion games [4]. The problem also relates to work by Hollinger et al. [16] and on mobile target estimation with fixed and uncontrollable range radios. Djugash and Singh [7] also developed a decentralized estimation algorithm for localizing static and mobile nodes using range-only sensors. In contrast, we focus on controlling mobile robots to localize a mobile target.

3 Mutual Information Based Control

In this section we describe a mutual information based control law for the mobile target tracking problem. The proposed approach considers a team of cooperating robots which concurrently measure the distance to the target via their RF sensors. The team shares observations (following Fig. 1) to jointly estimate the target's location (Sect. 3.1). By predicting where the target will go next and what measurements they are likely to receive (Sect. 3.2), the team chooses control directions that maximize the predicted reduction in entropy of the target's uncertainty over a finite time horizon (Sect. 3.3).

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

3.1 Target Estimation

Range measurements are a non-linear function of the target state and can easily lead to complex belief distributions such as rings and crescents [3, 7, 14]. Due to the limited information they provide, range measurements can also lead to multiple valid hypotheses. These beliefs cannot be well approximated by a single Gaussian and so we estimate the target's 2D position at time t using a particle filter with the typical recursive Bayesian filter equation [30]:

$$p(x_t \mid z_{1:t}) = \eta \, p(z_t \mid x_t) \int p(x_{t-1} \mid z_{1:t-1}) p(x_t \mid x_{t-1}) \, dx_{t-1} \tag{1}$$

where η is a normalization constant, x_t is the 2D position of the robot at time t, and z_t is the vector of range measurements the team receives at time t. For ease of reference, Tab. 1 describes all relevant variables. 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 un-cooperative target [26, 31] we use a Gaussian random walk for the process model: $p(x_t \mid x_{t-1}) = \mathcal{N}(x_t; x_{t-1}, \sigma_p^2 I)$ where I is the identity matrix and $\mathcal{N}(x; \mu, \Sigma)$ is a Gaussian with mean μ and covariance Σ.

Throughout this we assume that the measurement model, $p(z_t \mid x_t)$, can be modeled as the true distance between the target and measuring robot perturbed by Gaussian noise. Previous experimental work by Djugash and Singh [7] and the authors [3] 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.

Particle filters are expressive, but can also be computationally expensive, requiring many samples to accurately estimate a distribution. While we use KLD sampling [9] to keep the number of particles small, it is reasonable to ask whether the target's position should be estimated with a parametric filter such as the Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF). However, the EKF and UKF are only appropriate when the belief can be well approximated as a single Gaussian. This could be helpful once the team's estimate has converged, but at that point in time the particle filter should be able to represent the belief using a few hundred particles, making it fast for both estimation and control. Using a filter like the EKF also has its drawbacks. Hoffmann and Tomlin [14] describe how linearizing the measurement model introduces error into the calculation of mutual information, potentially hurting the team's overall performance.

3.2 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 $\boldsymbol{\tau} \overset{\triangle}{=} t+1 : t+T$, they need to estimate the target's future state: $x_{\boldsymbol{\tau}} = [x_{t+1}, \ldots, x_{t+T}]$. This can be done using the current belief and process model:

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

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

Over this time interval the team will receive range measurements $z_\tau(c_\tau) = [z_{t+1}(c_{t+1}), \ldots, z_{t+T}(c_{t+T})]$ 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 distribution can be calculated by marginalizing over the state and applying the conditional independence assumption of the measurements:

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

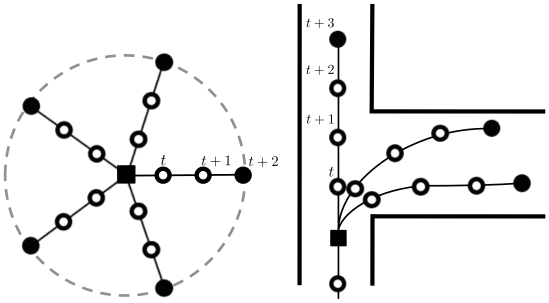
Because the measurement model is Gaussian, substituting (2) into (3) results in a Gaussian mixture model representation of the measurement density. If the team has R robots, then after applying the conditional independence assumption the j^{th} mixture component has a weight of w_j with a distribution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

3.3 Information Based Control

3.3.1 Control Law

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 plans how it will move over the time interval τ by considering a set of trajectories, $\mathcal{C}$, that they can follow. We define a trajectory as a discrete sequence of poses for each team member, so that a trajectory $c_\tau \in \mathcal{C}$ can be expressed as $c_\tau = [c_{t+1}, \ldots, c_{t+T}]$ where c^r_k is the 2D position of robot r at time k (we ignore orientation as it does not affect range measurements). After generating $\mathcal{C}$, the team selects the trajectory that maximizes mutual information:

$$c^*_{\boldsymbol{\tau}} = \underset{c_\tau \in \mathcal{C}}{\arg\max}\, \mathrm{I}[z_{\boldsymbol{\tau}}(c_\tau), x_{\boldsymbol{\tau}}] = \underset{c_\tau \in \mathcal{C}}{\arg\max}\, \mathrm{H}\left[z_{\boldsymbol{\tau}}(c_\tau)\right] - \mathrm{H}\left[z_{\boldsymbol{\tau}}(c_\tau) \mid x_{\boldsymbol{\tau}}\right] \tag{4}$$



(a) Points on a Circle (b) Indoor Planner

Figure 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.

where $\mathrm{I}[z_{\boldsymbol{\tau}}(c_\tau), x_{\boldsymbol{\tau}}]$ is the mutual information between future measurements and the state, which can be calculated as the difference between $\mathrm{H}\left[z_{\boldsymbol{\tau}}(c_{\boldsymbol{\tau}})\right]$, the differential entropy of future measurements, and $\mathrm{H}\left[z_{\boldsymbol{\tau}}(c_\tau) \mid x_{\boldsymbol{\tau}}\right]$, the conditional differential entropy of future measurements [5]. Although it is not explicit, both entropies are conditioned on measurements made up to time t. Both measurement distributions and their entropies change as a function of where the team travels.

To generate $\mathcal{C}$, 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).

3.3.2 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:

$$\mathrm{H}\left[z_{\boldsymbol{\tau}}(c_\tau) \mid x_{\boldsymbol{\tau}}\right] \approx \sum_{i=1}^{M} w_i \sum_{j=t+1}^{t+T} \sum_{r=1}^{R} \mathrm{H}\left[z_j^r(c_j^r) \mid x_j = \tilde{x}_j^i\right] \tag{5}$$

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 $\Theta(MRT)$.

Unfortunately, as shown in Sect. 3.2 the measurement density, $p(z_\tau(c_\tau) \mid z_{1:t})$, 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. [18]. 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 0^{th} order approximation:

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

where $g(x) = \sum_{i=1}^{M} w_i \mathcal{N}(x; \mu_i, \Sigma_i)$. While this approximation is not tight, we show experimentally that it suffices for information based control.

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 [25, 17]. In Sect. 4.2 we describe a different approach which reduces the number of components by approximating the distribution over the target's future state and prove that such approximations result in similar distributions over measurements.

4 Approximate Representations

In Sect. 3 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. 4.2 where we analyze how an approximation of the target's future state affects the objective function, and in Sect. 4.3 where we develop a theoretical design criterion for generating trajectories. These analyses use an information theoretic relationship that we prove in Sect. 4.1.

4.1 Bounding Entropy Difference

Intuitively, if two Gaussian mixture models (GMMs) are similar, then their entropies should be similar as well. In particular, one would expect that if the means of individual components of a GMM are slightly perturbed, then the entropy of the GMM would not change substantially. Our primary theoretical contribution is proving that this intuition is correct. We do this by building on a previously known result that relates the L_1 distance between discrete distributions and their difference in entropies. For clarity of presentation, all proofs are deferred to the Appendix.

The L_1 distance is one way of measuring the difference between distributions. For two densities f and g, it is defined as $\|f - g\|_1 = \int |f(x) - g(x)|\, dx$. To concisely state the following results we also define $\phi(x) = \mathcal{N}(x; 0, 1)$ as the standard 1-dimensional Gaussian, $\Phi(x)$ as its cumulative distribution function, $\|x\|_\Sigma = \sqrt{x^T \Sigma^{-1} x}$ as the Mahalanobis distance, and $|\cdot|$ as the determinant of a matrix.

Theorem 1 (L_1 bound on entropy)**.** *Let f and g be probability density functions such that $\|f - g\|_1 \leq 1/2$, $0 \leq f(x) \leq 1$ and $0 \leq g(x) \leq 1$. Then*

$$|\,\mathrm{H}[f] - \mathrm{H}[g]\,| \leq -\|f - g\|_1 \log \|f - g\|_1 + \|f - g\|_1 \, \mathrm{H}\left[\frac{|f(x) - g(x)|}{\|f - g\|_1}\right] \tag{7}$$

Cover and Thomas [5, Theorem 17.3.3] prove this result for discrete distributions, but the proof extends to continuous distributions whose likelihood is between 0 and 1. Using the following lemmas, we use Theorem 1 to prove that the entropy of a GMM does not vary substantially if the means of its components are perturbed.

Lemma 1 (L_1 between Gaussians with same covariance)**.** *If $f(x) = \mathcal{N}(x; \mu_1, \Sigma)$ and $g(x) = \mathcal{N}(x; \mu_2, \Sigma)$ are k-dimensional Gaussians then*

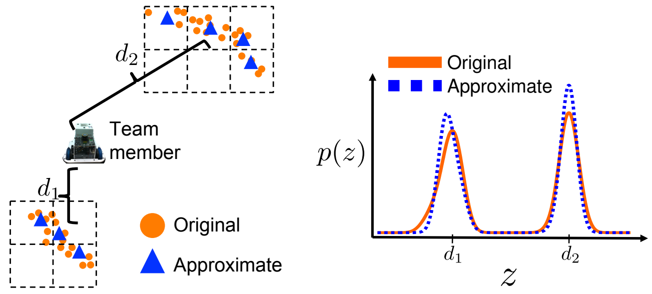
$$\|f - g\|_1 = 2\left(\Phi(\delta) - \Phi(-\delta)\right) = 2\left(2\Phi(\delta) - 1\right) \tag{8}$$

where $\delta = \|\mu_1 - \mu_2\|_\Sigma / 2$.

Lemma 2 (Entropy bound of normalized difference of Gaussians)**.** *If $f(x) = \mathcal{N}(x; \mu_1, \Sigma)$ and $g(x) = \mathcal{N}(x; \mu_2, \Sigma)$ are k-dimensional Gaussians with $\mu_1 \neq \mu_2$ then*

$$\mathrm{H}\left[\frac{|f - g|}{\|f - g\|_1}\right] \leq \log \sqrt{(2\pi e)^k |\Sigma| \left(1 + \delta^2 + \frac{2\delta\phi(\delta)}{2\Phi(\delta) - 1}\right)} \tag{9}$$

where $\delta = \|\mu_1 - \mu_2\|_\Sigma / 2$



(a) Distributions over target locations

(b) Distributions over range measurements

Figure 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)

Theorem 2 (Bound on Entropy Difference of GMMs). *Let $f(x) = \sum_{i=1}^{N} w_i f_i(x)$ be a k-dimensional GMM where $f_i(x) = \mathcal{N}(x; \mu_i, \Sigma_i)$ and $|\Sigma_i| \geq (2\pi)^{-k}$ so $0 \leq f_i(x) \leq 1$. If $g(x) = \sum_{i=1}^{N} w_i g_i(x)$ where $g_i(x) = \mathcal{N}(x; \mu_i + \Delta_i, \Sigma_i)$ is the GMM created by perturbing the means of components of f by Δ_i such that $\|f_i - g_i\|_1 \leq 1/2$ for all i then:*

$$|\,\mathrm{H}[f] - \mathrm{H}[g]\,| \leq \sum_{i=1}^{N} -w_i \|f_i - g_i\|_1 \log \frac{w_i \|f_i - g_i\|_1}{\sqrt{(2\pi e)^k |\Sigma_i| \left(1 + \delta_i^2 + \frac{2\delta_i \phi(\delta_i)}{2\Phi(\delta_i) - 1}\right)}} \tag{10}$$

where $\delta_i = \|\Delta_i\|_{\Sigma_i}/2$ and $\|f_i - g_i\|_1 = 2(2\Phi(\delta_i) - 1)$.

There are two primary limitations to Thm. 2, both of which come from Thm. 1. The first is that the bound only applies to GMMs where the determinant of the covariance of individual components is not too small; this is necessary to ensure that the maximum likelihood of each component is at most 1. This issue is not significant for our application as the covariance is determined by the noise of range measurements, which is typically quite large [3]. Another limitation is the L_1 constraint between components, which requires that each perturbation not be large relative to the covariance. This requirement can be relaxed by using a variant of Thm. 1. The L_1 constraint arises in the proof of Thm. 1 from an inequality of the entropy function, $h(t) = -t \log t$. Fannes [8] proved a similar inequality for $h(t)$ that is weaker, but does not have the L_1 constraint; this inequality can be substituted into the proof for Thm. 1 to eliminate the requirement that components be near each other.

There are other results that connect similarity metrics between distributions to the difference of their entropies. For example, Silva and Parada [27, Theorem 2] recently proved that in the limit as the KL divergence between bounded continuous distributions goes to 0, their entropies become equal. 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 bounds the entropy difference of continuous densities like GMMs whose support is unbounded.

4.2 Approximating Belief

As discussed in Sect. 3.3, the number of particles needed to predict the location of the target grows exponentially in time. This complicates calculating the entropy of the measurement distribution (3), as it depends on the predictive distribution of the target (2). 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 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 analyze the error introduced by this approximation, we assume that the range based measurement model has a constant variance σ_m^2 and is unbiased. With these assumptions, the conditional entropy (5) is $\mathrm{H}\left[z_{\boldsymbol{\tau}}(c_\tau) \mid x_{\boldsymbol{\tau}}\right] \approx (TR/2)\log(2\pi e \sigma_m^2)$, 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, $\mathrm{H}\left[z_{\boldsymbol{\tau}}(c_\tau)\right]$. We use Thm. 2 to bound this error, by showing that the approximate measurement distribution is a perturbed version of the original measurement distribution. Let $\mathcal{X}$ be the original set of particles and $f(z) = \sum_{j=1}^{M} w_j \mathcal{N}(z; \mu^{f_j}, \sigma_m^2 I)$ be the Gaussian mixture model (GMM) it creates with the trajectory c_τ and measurement model, using superscripts to index components. Let $\mathcal{G}(j)$ be a map from the index of a particle in $\mathcal{X}$ to the index of the corresponding particle in the approximate particle set $\mathcal{Y}$. The density $\mathcal{Y}$ creates is $g(z) = \sum_{i=1}^{N} \pi_i \mathcal{N}(z; \mu^{g_i}, \sigma_m^2 I)$, where the i^{th} component's mean and weight is determined by the particles in $\mathcal{X}$ that map to the i^{th} particle in $\mathcal{Y}$. While $f(z)$ and $g(z)$ may have a different number of components, $M \geq N$, the expressions are related in that each component in $f(z)$ corresponds to a single component in $g(z)$. This means the approximated measurement distribution can be re-expressed as $g(z) = \sum_{j=1}^{M} w_j \mathcal{N}(z; \mu^{f_j} + \Delta^j, \sigma_m^2 I)$ where Δ^j is the difference between μ^{f_j} and $\mu^{g_{\mathcal{G}(j)}}$. Each Δ^j can be bounded. Note that 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 the farthest they can be apart is $\|\tilde{x}_k^{\mathcal{X}_j} - \tilde{x}_k^{\mathcal{Y}_i}\|_2 \leq \sqrt{2}L$ where L is the cell's length. Thus, each component of the perturbation is at most $\sqrt{2}L$. As the perturbation vector is RT dimensional for all j:

$$\|\Delta^j\|_{\sigma_m^2 I} \leq \frac{\sqrt{2}LRT}{\sigma_m} \tag{11}$$

This bounds the Mahalanobis distance between the mixtures corresponding means. When the difference is small, Thm. 2 bounds the difference in entropies. Importantly, the bound shows that if the length of a grid cell is below the standard deviation of the measurement model, then the error from the approximation will be negligible.

4.3 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 show 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 as their mutual information will be similar. 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 $f(z) = \sum_{i=1}^{M} w_i \mathcal{N}(z; \mu^{f_i}, \sigma_m^2 I)$ be the measurement density it determines when combined with the measurement model and $\mathcal{X}$, the particle set representing the predicted motion of the target. If α is a perturbation to the team's position at each point in time, $c + \alpha$ is a new trajectory with measurement density $g(z) = \sum_{i=1}^{M} w_i \mathcal{N}(z; \mu^{g_i}, \sigma_m^2 I)$. $\mathcal{X}$ is the same for c and $c + \alpha$, so $f(z)$ and $g(z)$ have the same number of components with identical weights. Similar to the last section, the conditional entropy of the measurements is the same for the original and perturbed trajectories. Consequently, to determine the difference in mutual information, we need to determine the difference in entropies between $f(z)$ and $g(z)$.

The entropy of $f(z)$ and $g(z)$ are not necessarily equal as the means of individual components may change. To bound this difference, consider the distance between the i^{th} particle and the r^{th} robot at time k for both trajectories: $\mu^{f_i}_{k,r} = \|\tilde{x}^i_k - c^r_k\|_2$ and $\mu^{g_i}_{k,r} = \|\tilde{x}^i_k - (c^r_k + \alpha^r_k)\|_2$. The triangle inequality tells us $|\mu^{f_i}_{k,r} - \mu^{g_i}_{k,r}| \leq \|\alpha^r_k\|_2$. Defining Δ^i as a perturbation to μ^{f_i} that transforms it to μ^{g_i}, and summing over all time steps and robots:

$$\|\Delta^i\|_{\sigma_m^2 I} = \|\mu^{g_i} - \mu^{f_i}\|_{\sigma_m^2 I} \leq \sum_{k=t+1}^{t+T} \sum_{r=1}^{R} \frac{\|\alpha^r_k\|_2}{\sigma_m} \tag{12}$$

When the change in a robot's position is small relative to the standard deviation of the measurement model, the mixture components will not be significantly perturbed. By Thm. 2, this means the entropy of the mixture will not change significantly.

4.4 Bounding Entropy Difference using Kullback-Leibler Divergence

The Kullback-Leibler divergence (KL divergence or KLD) is one way of measuring the "distance" between two densities. For two densities f and g, it is defined as $\mathrm{KL}[f \mid\mid g] = \mathbb{E}_f\left[\log f(x)/g(x)\right]$. In previous work, we argued that the KL divergence can be used to bound the difference in entropies of GMMs. However, this reasoning was flawed and we incorrectly concluded that the bounds provided useful insight into how the entropy of a GMM can change. To correct the record, we detail this prior result and clarify the source of the issue.

Lemma 3 (KLD Bound on Entropy for Gaussians)**.** *If $f(x)$ and $g(x)$ are two k-dimensional Gaussian densities:*

$$|\,\mathrm{H}\left[f\right] - \mathrm{H}\left[g\right]| < \min\{\mathrm{KL}[f \mid\mid g], \mathrm{KL}[g \mid\mid f]\} + \frac{k}{2}$$

Lemma 4 (Entropy Upper Bound for Mixture Models)**.** *If $f(x) = \sum_i w_i f_i(x)$ is a mixture model density then $\mathrm{H}\left[f\right] \leq \mathrm{H}\left[w\right] + \sum_i w_i \,\mathrm{H}\left[f_i\right]$ where $\mathrm{H}\left[w\right]$ is the discrete entropy of the mixture weights (Proved by Huber et al. [18]).*

Lemma 5 (Entropy Lower Bound for Mixture Models)**.** *If $f(x) = \sum_i w_i f_i(x)$ is a mixture model, $\mathrm{H}\left[f\right] \geq \sum_i w_i \,\mathrm{H}\left[f_i\right]$.*

Theorem 3 (Entropy Difference for Mixture Models)**.** *If $f(x) = \sum_{i=1}^N w_i f_i(x)$ and $g(x) = \sum_{i=1}^N w_i g_i(x)$ are mixture models whose components have equal weights then*

$$|\,\mathrm{H}\left[f\right] - \mathrm{H}\left[g\right]| \leq \mathrm{H}\left[w\right] + \sum_{i=1}^{N} w_i |\,\mathrm{H}\left[f_i\right] - \mathrm{H}\left[g_i\right]|$$

where $\mathrm{H}\left[w\right]$ is the discrete entropy of the mixture weights.

Theorem 3 does not connect the difference in entropies between f and g to their similarity as measured by KL divergence or a similar quantity. In previous work, we listed a slight variant of it, [2, Thm. 1], which used Lem. 3 to bound the summand in Thm. 3. While that is valid, it is not relevant for GMMs where each component has the same covariance because the summand is always 0: the entropy of a Gaussian only depends on its covariance, so when corresponding components have the same covariance, they have the same entropy, making their difference 0. Because the fixed covariance noise model results in GMMs whose components have identical covariance, it is incorrect to conclude, as we previously did, that a bounded KL divergence between components gives insight into the entropy difference between the GMMs.

A challenge to the pragmatic contribution of Thm. 3 is that the upper and lower bounds used to prove it are strong at different times. The upper bound on the entropy (Lem. 4) is tightest when each of the components is separated and their shared support is minimal [18]. In contrast, the lower bound (Lem. 5)

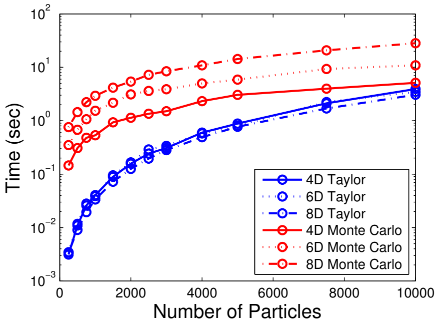


Figure 4: Monte Carlo integration vs. 0th order Taylor approximation for evaluating mutual information. The Taylor approximation is more efficient.

is tightest when each component is essentially identical. It is also worth noting that the bound on entropy difference is looser as the mixture weights come closer to each other. In the worst case, w is the uniform distribution meaning $\mathrm{H}[w] = \log N$ [5]. Note that Thm. 2 has no such offset term, making it a more insightful bound. Consequently, Thm. 2 in this paper enables us to reach the same conclusions we previously made, which we believe resolves the issue.

5 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++, run on an Intel Core i7 processor, and interfaced using the Robot Operating System [24].

5.1 Computational Performance

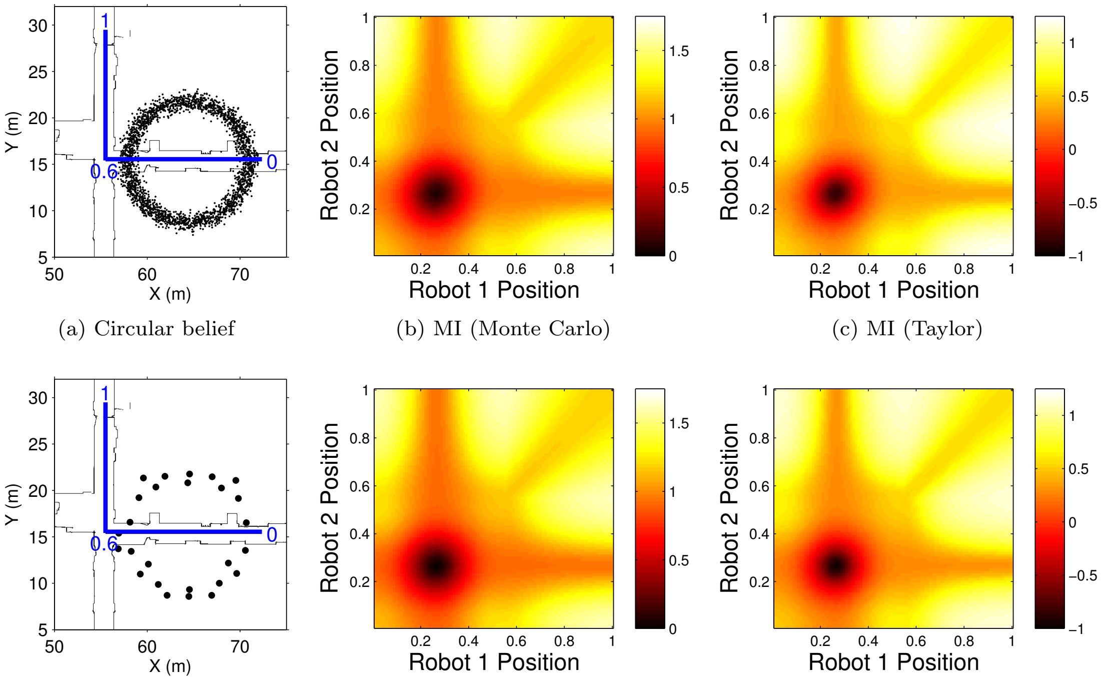
In Sect. 3.3 we discussed how the primary computational challenge of the objective is the calculation of the measurement density's entropy. We now study the performance of Huber's 0th order Taylor approximation and show that the approximation algorithm in Sect. 4.2 makes it tractable for longer time horizons. We also discuss the impact of using an alternative filter for estimation and calculating the objective.

To assess the performance of the Taylor approximation, we compare it to the Vegas algorithm – a Monte Carlo integration algorithm provided by the Cuba library [13] – 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.

5.2 Effects of Approximations

To understand the effect of the approximations on the team's decisions, we visualize their impact on the mutual information cost-surface in a few typical scenarios.

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 circular belief distribution represented by a particle filter (M = 2500, Fig. 5a), mutual information is computed using Monte Carlo integration for all possible locations of the team with range-only observations ($\sigma_p^2 = 3$ m, Fig. 5b). The value



(a) Circular belief

(b) MI (Monte Carlo)

(c) MI (Taylor)

(d) Approx. circular belief

(e) Approx. MI (Monte Carlo)

(f) Approx. MI (Taylor)

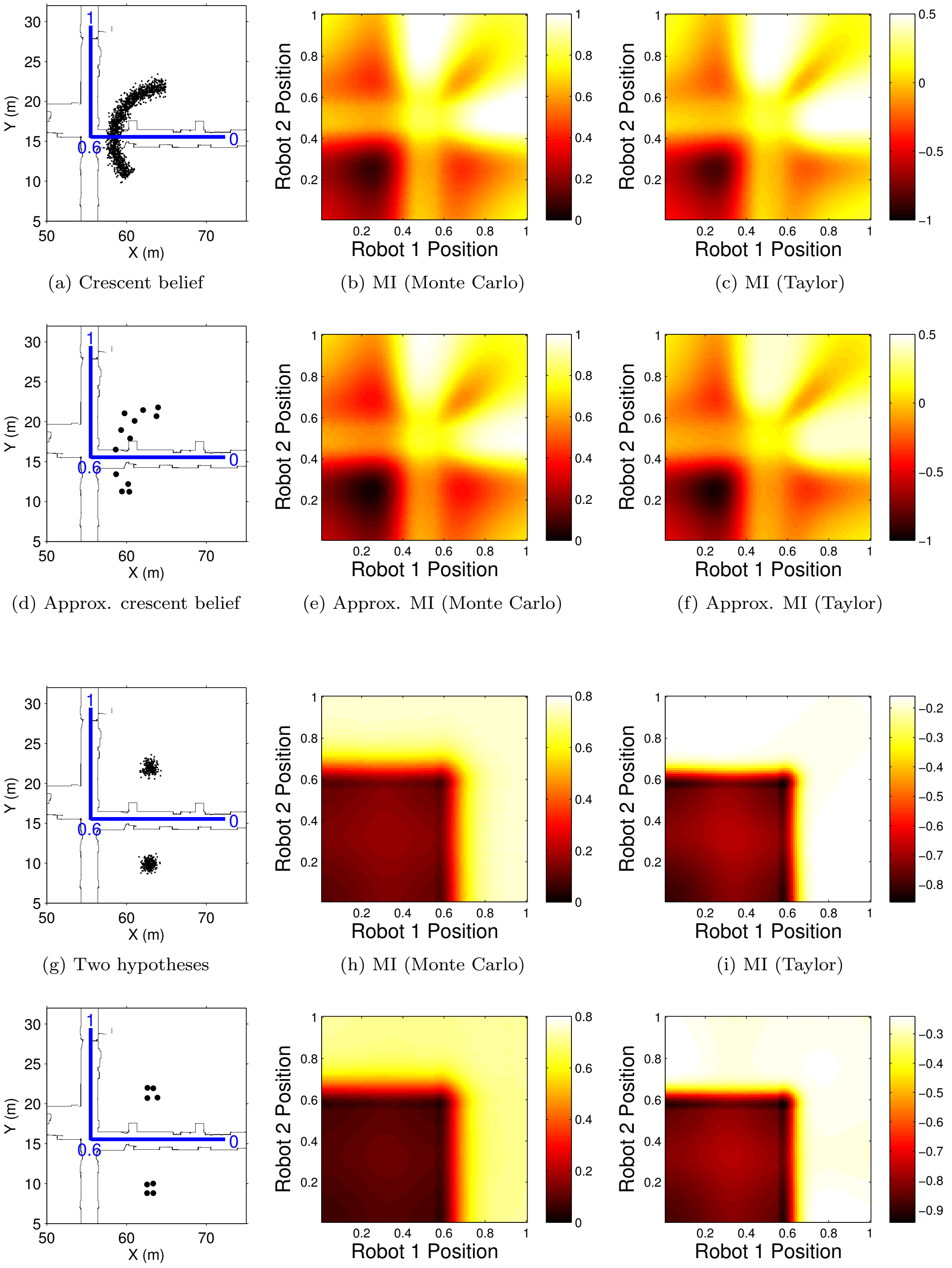
Figure 5: Effect of approximating belief on mutual information (MI) as two robots move along a parameterized line from [0,1] (blue, (a) and (d)). Approximating belief introduces limited error into mutual information when calculating it via Monte Carlo integration ((b) vs. (e)) or the Taylor approximation ((c) vs. (f)).

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 approximate belief representation with fewer particles (N = 26, Fig. 5d) corresponding to a grid length of 3.0 m yields mutual information values that are similar to the original distribution (Fig. 5e). The mean absolute error between the original and approximate mutual information is 0.03.

It is also important to understand the effect of approximating the belief on the Taylor approximation, as that is how we calculate mutual information in simulations and experiments. Comparing Fig. 5c to Fig. 5f, it is clear that the approximation introduces limited error for the Taylor approximation as well; the mean absolute error is 0.06. While this is larger than it was for numerical integration, it still represents a small change.

To further verify that the approximation introduces limited error, Fig. 6 shows its effect when the belief is distributed as a crescent or two separate hypotheses. These distributions are useful examples as they often arise when using range-only measurements [7, 3]. As in Fig. 5, it is clear that the belief approximation introduces limited error for both Monte Carlo integration (mean absolute error below 0.04) and the Taylor approximation (mean absolute error below 0.09). It also significantly reduces the number of particles resulting in a substantial computational speedup ($M = 1500$ to $N = 12$ for the crescent and $M = 500$ to $N = 8$ for two hypotheses).

Looking at the Taylor approximation in isolation might suggest that it performs poorly. For example, it sometimes results in a negative value for mutual information which is a non-negative quantity [5]. However, for the purposes of selecting a control action, the location of the maxima matter more than their value. Given this, by comparing Fig. 5b and Fig. 5c it is clear that using the Taylor approximation will cause similar actions to be selected: the maxima occur at the same place in both plots, demonstrating that the Taylor approximation performs well.



(a) Crescent belief

(b) MI (Monte Carlo)

(c) MI (Taylor)

(d) Approx. crescent belief

(e) Approx. MI (Monte Carlo)

(f) Approx. MI (Taylor)

(g) Two hypotheses

(h) MI (Monte Carlo)

(i) MI (Taylor)

(j) Approx. two hypotheses

(k) Approx. MI (Monte Carlo)

(l) Approx. MI (Taylor)

Figure 6: Effect of approximating different beliefs on mutual information (MI) as two robots move along a parameterized line from [0,1] (blue, (a), (d), (g) and (j). The approximation introduces limited error for Monte Carlo integration and the Taylor approximation.

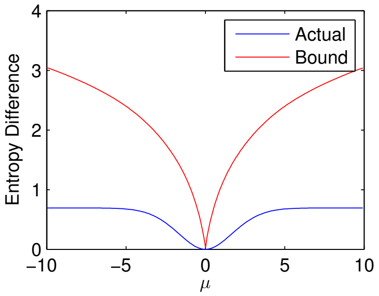


Figure 7: Comparison of actual entropy difference to bound from Thm. 2 for mixture models that differ by one component. The theoretical bound is not sharp, but becomes tighter as the components become more similar.

The previous plots demonstrate that the approximation introduces relatively small amounts of error in typical scenarios. However, they do not illustrate the theoretical bound on entropy difference (Thm. 2). A simple example comparing the entropy of a single Gaussian to a two-component mixture model makes the bound clearer. Let $f(x) = \mathcal{N}(x; 0, 1)$ be the standard 1D Gaussian and $g(x; \mu) = 0.5\mathcal{N}(x; 0, 1) + 0.5\mathcal{N}(x; \mu, 1)$ be a 2-component mixture model where the mean of the second component is μ. Figure 7 shows the actual entropy difference, $|\,\mathrm{H}[f] - \mathrm{H}[g]\,|$, between these mixtures along with the upper bound from Thm. 2 for various values of μ. When μ is close to 0 the mixtures are similar and have similar entropies, which the bound captures. As μ increases, the bound is weaker, becoming much larger than the true entropy difference. Despite this fact, Thm. 2 still provides useful theoretical insight into the effect of the approximation.

5.3 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. 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.

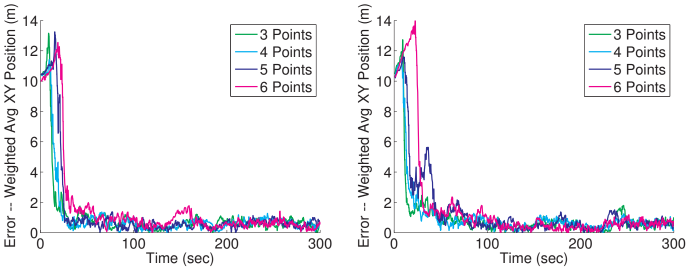
At each point in time, the team plans a 15 second trajectory – simulating 3 target steps – and evaluates mutual information at 3 points along each trajectory. Although each robot will make significantly more than 3 measurements while executing its trajectory, it is not necessary to evaluate mutual information at each position that a measurement will be received. As described in Sect. 4.3, mutual information at nearby locations must be similar, so densely evaluating it will not further distinguish which trajectories are more informative than others.

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 $\sigma_p = 0.3$, 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 $\sigma_p = 0.25$.

The team "acquires" the target once they sustain an estimate for 10 seconds with a mean error below 1.5 mand 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 2 shows acquisition times for 5 trials with a variable number of robots and destination points. Surprisingly, the 3 robot team does worse with more

Table 2: 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)



(a) Random walk

(b) Back and forth

Figure 8: 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.

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 8a shows the estimate's error over time. While the target is acquired at different points in time, the final error is the same 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 8b 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 the estimation and control approach work well.

5.4 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. 9. 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 [22]. The nanoPAN 5375's measurements often have an error larger than 2.0 m [3].

In our first experiment, the target traverses its loop once. Figure 10a 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

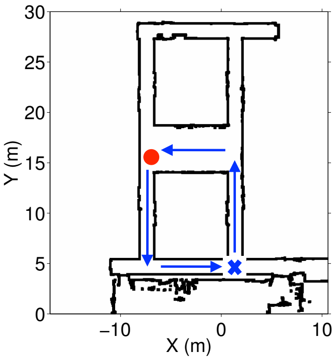
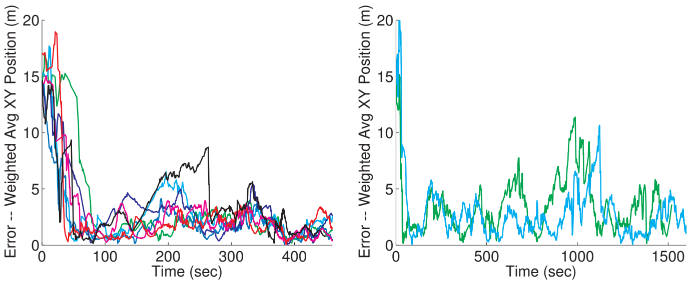


Figure 9: Indoor experimental setup. The target starts at the blue "X" and moves in a loop. The team starts at the red circle.



(a) 1 loop (7 trials)

(b) 4 loops (2 trials)

Figure 10: Distance from the estimate's mean to the target's true location as the target drives around.

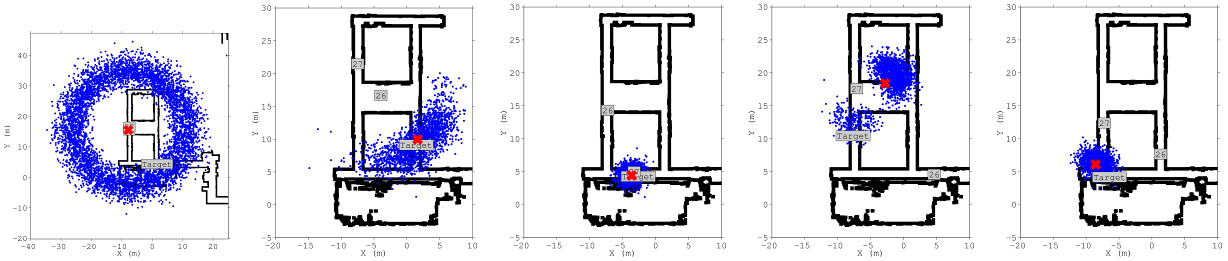
same sensor [3]. Given that the target is moving in this experiment, we believe these results indicate that the team performs well.

In our second experiment, we assess the longer term performance of our approach by having the target traverse the same loop 4 times. Figure 10b shows the error of 2 different trials and Fig. 11 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 11d and Fig. 10b 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.

6 Conclusion

There are many interesting directions for future research. For instance, in this work the team follows a sequence of waypoints at a fixed velocity, but could presumably increase their performance by moving faster or slower at various points in time. We would also like to investigate integrating other sensors. Finally, we are motivated by the theoretical results of this work and seek to prove stronger relationships between similarity measures of continuous distributions and the difference of their entropies.

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 that the entropy



(a) $t = 2.29$s (b) $t = 47.08$s (c) $t = 349.64$s (d) $t = 1028.41$s (e) $t = 1091.81$s

Figure 11: 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.

of a Gaussian mixture model cannot change substantially when its components are perturbed, we argue that these approximations do not significantly affect the teams behavior. 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.

7 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] Jonathan Binney, Andreas Krause, and Gaurav S Sukhatme. Optimizing waypoints for monitoring spatiotemporal phenomena. *Int. J. Robot. Research*, 32(8):873–888, 2013.

[2] Benjamin Charrow, Vijay Kumar, and Nathan Michael. Approximate representations for multi-robot control policies that maximize mutual information. In *Robotics: Science and Systems*, Berlin, Germany, June 2013.

[3] Benjamin Charrow, Nathan Michael, and Vijay Kumar. Cooperative multi-robot estimation and control for radio source localization. *Int. J. Robot. Research*, April 2014.

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

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

[6] Amaury Dame and Eric Marchand. Mutual information-based visual servoing. *IEEE Trans. on Robot.*, 27(5):958–969, 2011.

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

[8] Mark Fannes. A continuity property of the entropy density for spin lattice systems. *Communications in Mathematical Physics*, 31(4):291–294, 1973.

[9] Dieter Fox. Adapting the sample size in particle filters through KLD-sampling. *Int. J. Robot. Research*, 22(12):985–1003, 2003.

[10] Jun geun Park, Benjamin 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.

[11] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. of AI Research*, 42(1):427–486, 2011.

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

[13] Thomas Hahn. Cuba. http://www.feynarts.de/cuba/, January 2013.

[14] 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.

[15] Geoffrey Hollinger and Gaurav Sukhatme. Sampling-based motion planning for robotic information gathering. In *Robotics: Science and Systems*, Berlin, Germany, June 2013.

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

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

[18] 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.

[19] 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.

[20] Abdallah Kassir, Robert Fitch, and Salah Sukkarieh. Decentralised information gathering with communication costs. In *Proc. of the IEEE Int. Conf. on Robot. and Autom.*, pages 2427–2432, Saint Paul, USA, May 2012.

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

[22] nanoPAN 5375 Development Kit. http://nanotron.com/EN/pdf/Factsheet_nanoPAN_5375_Dev_Kit.pdf, September 2013.

[23] Donald Owen. A table of normal integrals. *Communications in Statistics-Simulation and Computation*, 9(4):389–419, 1980.

[24] ROS. http://www.ros.org/wiki/, January 2013.

[25] Andrew Runnals. Kullback-leibler approach to gaussian mixture reduction. *IEEE Trans. on Aero. and Elec. Sys.*, 43(3):989–999, July 2007.

[26] Allison Ryan and J. Karl Hedrick. Particle filter based information-theoretic active sensing. *Robot. and Autonomous Syst.*, 58(5):574–584, May 2010.

[27] 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.

[28] 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.

[29] 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.

[30] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics.* MIT Press, 2008.

[31] Réne 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.

[32] Peter Whaite and Frank P Ferrie. Autonomous exploration: Driven by uncertainty. *IEEE Trans. Pattern Analysis and Machine Intell.*, 19(3):193–205, 1997.

A Proofs

A.1 Integrating Gaussians over a half-space

Proving Lems. 1 and 2 requires integrating Gaussian functions over a half-space. The following identities will be used several times. Owen [23] gives these identities for 1-dimensional Gaussians, but we have been unable to find a reference for multivariate case. For completeness, we prove them here.

Lemma 6. *If $f(x) = \mathcal{N}(x; \mu, \Sigma)$ is a k-dimensional Gaussian and $A = \{x : a^T x + b > 0\}$ is a half-space then*

$$\int_A f(x)\, dx = \Phi(p) \tag{13}$$

$$\int_A x f(x)\, dx = \Phi(p)\, \mu + \phi(p) \frac{\Sigma a}{\sqrt{a^T \Sigma a}} \tag{14}$$

$$\int_A x x^T f(x)\, dx = \Phi(p)\, (\Sigma + \mu\mu^T) + \phi(p) \left(\frac{\Sigma a \mu^T + \mu a^T \Sigma}{\sqrt{a^T \Sigma a}} - p \frac{\Sigma a a^T \Sigma}{a^T \Sigma a} \right) \tag{15}$$

where $p = (a^T \mu + b)/\sqrt{a^T \Sigma a}$ is a scalar, $\phi(x) = \mathcal{N}(x; 0, 1)$ is the PDF of the standard 1-dimensional Gaussian and $\Phi(x)$ is its CDF.

Proof of (13). All of these integrals are evaluated by making the substitution $x = Ry$, where R is a rotation matrix that makes the half-space A axis aligned. Specifically, define R such that $a^T R = \|a\| e_1^T$ where e_1 is a k-dimensional vector whose first entry is 1 and all others are 0. This substitution is advantageous, because it makes the limits of integration for all components of y except y_1 $[-\infty, \infty]$.

Because $R^T x = y$, y is a k-dimensional Gaussian with density $q(y) = \mathcal{N}(y; R^T \mu, R^T \Sigma R)$. The determinant of the Jacobian of y is the determinant of a rotation matrix: $|\partial y/\partial x| = |R^T| = 1$. Substituting:

$$\int_{a^T x + b > 0} f(x)\, dx = \int_{(a^T R) y + b > 0} |R^T| q(y)\, dy = \int_{\|a\| y_1 + b > 0} q(y)\, dy = \int_{-b/\|a\|}^{\infty} q_1(y_1)\, dy_1 \tag{16}$$

Where $q_1(y_1) = \mathcal{N}(y_1; e_1^T(R^T \mu), e_1^T(R^T \Sigma R) e_1)$ is the density for the first component of y. The final step follows as the limits of integration marginalize $q(y)$. To simplify the remaining integral, apply the definition of R, $q_1(y_1) = \mathcal{N}(y_1; \mu^T a/\|a\|, a^T \Sigma a/\|a\|^2)$, and use $1 - \Phi(-x) = \Phi(x)$:

$$\int_{-b/\|a\|}^{\infty} q_1(y_1)\, dy_1 = 1 - \Phi\left(\frac{-b - \mu^T a}{\sqrt{a^T \Sigma a}} \right) = \Phi\left(\frac{a^T \mu + b}{\sqrt{a^T \Sigma a}} \right) \tag{17}$$

□

Proof of (14). First, we perform a change of variables so that the integral is evaluated over the standard multivariate Gaussian $g(x) = \mathcal{N}(x; 0, I)$. This substitution is $x = \Sigma^{1/2}y + \mu$ which can be seen by noting that 1) $|\partial y/\partial x| = |\Sigma|^{-1/2}$ and 2) $f(x) = |\Sigma|^{-1/2}g(\Sigma^{-1/2}(x - \mu))$:

$$\int_A xf(x)\,dx = \int_B (\Sigma^{1/2}y + \mu)g(y)\,dy = \Sigma^{1/2}\int_B yg(y)\,dy + \mu\int_B g(y)\,dy \tag{18}$$

where $B = \{y : (a^T\Sigma^{1/2})y + (a^T\mu + b) > 0\}$ is the transformed half-space.

To evaluate the first term in (18), we calculate $\int_C yg(y)\,dy$, where $C = \{y : c^Ty + d > 0\}$ is a new generic half-space. This integral is easier than the original problem as $g(y)$ is the density of a zero-mean Gaussian with identity covariance. To proceed, substitute $y = Rz$ where R is a rotation matrix that makes C axis-aligned (i.e., $c^TR = \|c\|e_1^T$) and observe that $g(Rz) = g(z)$:

$$\int_C yg(y)\,dy = \int_{c^TRz+d>0} Rzg(Rz)|R^T|\,dz = Re_1\int_{-d/\|c\|}^{\infty} z_1\phi(z_1)\,dz_1 = Re_1\phi\left(\frac{-d}{\|c\|}\right) = \frac{c}{\|c\|}\phi\left(\frac{d}{\|c\|}\right) \tag{19}$$

e_1 appears because $g(x)$ is 0-mean; the only non-zero component of the integral is from z_1. The 1-dimensional integral follows as $d\phi(x)/dx = -x\phi(x)$.

To finish, (18) can be evaluated using the formula in (13) and (19):

$$\int_{a^Tx+b>0} xf(x)\,dx = \Sigma^{1/2}\left(\frac{\Sigma^{1/2}a}{\sqrt{a^T\Sigma a}}\right)\phi\left(\frac{a^T\mu + b}{\sqrt{a^T\Sigma a}}\right) + \mu\Phi\left(\frac{a^T\mu + b}{\sqrt{a^T\Sigma a}}\right) \tag{20}$$

□

Proof of (15). Similar to the last proof, make the substitution $x = \Sigma^{1/2}y + \mu$ with $g(y)$ as the standard multivariate Gaussian and expand terms.

$$\int_A xx^Tf(x)\,dx = \Sigma^{1/2}\int_B yy^Tg(y)\,dy\Sigma^{1/2} + \Sigma^{1/2}\int_B yg(y)\,dy\mu^T + \mu\int_B y^Tg(y)\,dy\Sigma^{1/2} + \mu\mu^T\int_B g(y)\,dy \tag{21}$$

where $B = \{y : (a^T\Sigma^{1/2})y + (a^T\mu + b) > 0\}$ is the transformed half-space.

To evaluate (21), we only need a formula for the first integral; the previous proofs have expressions for the other 3 integrals. To evaluate the first integral, let $C = \{y : c^Ty + d > 0\}$ be a new half-space and use the same rotation substitution $y = Rz$ as in the last proof.

$$\int_C yy^Tg(y)\,dy = R\left(\int_{z_1\|c\|+d>0} zz^Tg(z)\,dz\right)R^T = R\left(\Phi\left(\frac{d}{\|c\|}\right)I - \frac{d}{\|c\|}\phi\left(\frac{d}{\|c\|}\right)e_1e_1^T\right)R^T \tag{22}$$

$$= \Phi\left(\frac{d}{\|c\|}\right)I - \frac{d}{\|c\|}\phi\left(\frac{d}{\|c\|}\right)\frac{cc^T}{\|c\|^2} \tag{23}$$

The previous integral can be evaluated by analyzing each scalar component. g is the standard multivariate Gaussian, so $g(z) = \prod_{i=1}^k \phi(z_i)$. There are three types of terms:

- z_1^2: The limits of integration marginalize g and the resulting integral can be solved using integration by parts:

$$\int_{z_1\|c\|+d>0} z_1^2g(z)\,dz = \int_{-d/\|c\|}^{\infty} z_1^2\phi(z_1)\,dz_1 = \Phi\left(\frac{d}{\|c\|}\right) - \frac{d}{\|c\|}\phi\left(\frac{d}{\|c\|}\right) \tag{24}$$

- $z_i^2; (i > 1)$:

$$\int_{z_1\|c\|+d>0} z_i^2g(z)\,dz = \left(\int_{-d/\|c\|}^{\infty}\phi(z_1)\,dz_1\right)\left(\int_{-\infty}^{\infty} z_i^2\phi(z_i)\,dz_i\right) = \Phi\left(\frac{d}{\|c\|}\right) \tag{25}$$

The integral over z_i is 1 because it's the variance of the standard normal.

- $z_i z_j$ $(i \neq j, i \neq 1)$: These indices cover all non-diagonal elements.

$$\int_{z_1\|c\|+d>0} z_i z_j g(z)\, dz = \left(\int_{\alpha}^{\beta} z_j \phi(z_j)\, dz_j\right)\left(\int_{-\infty}^{\infty} z_i \phi(z_i)\, dz_i\right) = 0 \tag{26}$$

$z_i \neq 1$, so its limits are the real line. Because g is 0 mean the integral is 0.

We now have formula for all of the terms in (21). Recall that $B = \{x : (\Sigma^{1/2}a)^T x + (a^T\mu + b) > 0\}$. Using $p = (a^T\mu + b)/\sqrt{a^T\Sigma a}$, (13), (14), and (23):

$$\begin{aligned}\int_{a^Tx+b} xx^T f(x)\, dx =& \Sigma^{1/2}\left(\Phi(p)\, I - p\phi(p)\, \frac{\Sigma^{1/2}aa^T\Sigma^{1/2}}{\|a\|^2}\right)\Sigma^{1/2} + \\ & \Sigma^{1/2}\left(\phi(p)\, \frac{\Sigma^{1/2}a}{\sqrt{a^T\Sigma a}}\right)\mu^T + \mu\left(\phi(p)\, \frac{\Sigma^{1/2}a}{\sqrt{a^T\Sigma a}}\right)^T \Sigma^{1/2} + \\ & \Phi(p)\, \mu\mu^T & (27) \\ =& \Phi(p)\left(\Sigma + \mu\mu^T\right) + \phi(p)\left(\frac{\Sigma a\mu^T + \mu a^T\Sigma}{\sqrt{a^T\Sigma a}} - p\frac{\Sigma aa^T\Sigma}{\|a\|^2}\right) & (28)\end{aligned}$$

which is the formula we sought.

□

A.2 Proof of Lem. 1

The norm can be evaluated by splitting the integral up into regions where the absolute value disappears. Let A be the set where $f(x) > g(x)$ and A^c be the complement of A where $f(x) \leq g(x)$. Noting that $\int_A g(x) = 1 - \int_{A^c} g(x)$:

$$\int |f(x) - g(x)|\, dx = \int_A f(x) - g(x)\, dx + \int_{A^c} g(x) - f(x)\, dx = 2\left(\int_A f(x) - g(x)\, dx\right) \tag{29}$$

f and g have the same covariance, so f is bigger than g on a half-space $A = \{x : a^T x + b > 0\}$ where $a = \Sigma^{-1}(\mu_1 - \mu_2)$ and $b = (\mu_1 + \mu_2)\Sigma^{-1}(\mu_2 - \mu_1)/2$. This means (29) can be evaluated using Lem. 6. To do this, we need to evaluate p for $\int_A f(x)\, dx$ and $\int_A g(x)\, dx$. There are three relevant terms:

$$\mu_1^T a + b = \mu_1^T\Sigma^{-1}(\mu_1 - \mu_2) + \frac{(\mu_1 + \mu_2)\Sigma^{-1}(\mu_2 - \mu_1)}{2} = \frac{\|\mu_1 - \mu_2\|_\Sigma^2}{2} \tag{30}$$

$$\mu_2^T a + b = \mu_2^T\Sigma^{-1}(\mu_1 - \mu_2) + \frac{(\mu_1 + \mu_2)\Sigma^{-1}(\mu_2 - \mu_1)}{2} = -\frac{\|\mu_1 - \mu_2\|_\Sigma^2}{2} \tag{31}$$

$$\sqrt{a^T\Sigma a} = \sqrt{(\mu_1 - \mu_2)^T\Sigma^{-1}\Sigma\Sigma^{-1}(\mu_1 - \mu_2)} = \|\mu_1 - \mu_2\|_\Sigma \tag{32}$$

Using $\delta = \|\mu_1 - \mu_2\|_\Sigma/2$ we get $(a^T\mu_1 + b)/\sqrt{a^T\Sigma a} = \delta$ and $(a^T\mu_2 + b)/\sqrt{a^T\Sigma a} = -\delta$. Plugging these values into Lem. 6 completes the proof.

A.3 Proof of Lem. 2

Let X be a random variable whose density is $|f(x) - g(x)|/\|f - g\|_1$. Calculating the entropy of X is difficult as the expression involves the log of the absolute value of the difference of exponentials. Fortunately, the covariance of X can be calculated. This is useful because the maximum entropy of any distribution with covariance Σ is $(1/2)\log\left((2\pi e)^k|\Sigma|\right)$, the entropy of the multivariate Gaussian [5, Theorem 8.6.5]. By explicitly calculating the determinant of the covariance of X, we prove the desired bound.

To calculate X's covariance, use the formula $\operatorname{cov} X = \mathbb{E}\left[XX^T\right] - \mathbb{E}\left[X\right]\mathbb{E}\left[X\right]^T$. Similar to the proof of Lem. 1, we evaluate the mean by breaking the integral up into a region A where $f(x) > g(x)$ and A^C where $g(x) \geq f(x)$ vice-versa:

$$\mathbb{E}\left[X\right] = \frac{1}{\|f-g\|_1}\left(\int_A x(f(x)-g(x))\,dx + \int_{A^c} x(g(x)-f(x))\,dx\right) \tag{33}$$

As before, f and g have the some covariance, so $A = \{x : a^Tx + b > 0\}$ and $A^c = \{x : (-a)^Tx + (-b) \geq 0\}$ are half-spaces with $a = \Sigma^{-1}(\mu_1 - \mu_2)$ and $b = (\mu_1 + \mu_2)\Sigma^{-1}(\mu_2 - \mu_1)/2$.

Each of the terms in (33) are Gaussian functions integrated over a half-space, which can be evaluated using Lem. 6. To simplify the algebra, we evaluate the integrals involving f and g separately. This involves calculating a few terms, three of which are repeats: (30), (31), and (32). The other term is $\Sigma a = \mu_1 - \mu_2$. As the difference of the means will arise repeatedly, define $\Delta = \mu_1 - \mu_2$. Noting $2\delta = \|\Delta\|_\Sigma$ and $\phi(x) = \phi(-x)$, the integrals involving f are:

$$\int_A xf(x)\,dx - \int_{A^c} xf(x)\,dx = \Phi\left(\delta\right)\mu_1 + \frac{\phi\left(\delta\right)}{2\delta}\Delta - \left(\Phi\left(-\delta\right)\mu_1 + \frac{\phi\left(-\delta\right)}{2\delta}(-\Delta)\right) \tag{34}$$

$$= (\Phi\left(\delta\right) - \Phi\left(-\delta\right))\mu_1 + \frac{\phi\left(\delta\right)}{\delta}\Delta \tag{35}$$

Next, evaluate the integrals in (33) involving g. This can be done by pattern matching from (35). The main change is that μ_1 becomes μ_2 and the sign of p in Lem. 6 flips, meaning the sign of the arguments to $\phi\left(\cdot\right)$ and $\Phi\left(\cdot\right)$ flip (see (30) and (31)).

$$\int_A xg(x)\,dx - \int_{A^c} xg(x)\,dx = (\Phi\left(-\delta\right) - \Phi\left(\delta\right))\mu_2 + \frac{\phi\left(-\delta\right)}{\delta}\Delta \tag{36}$$

Subtracting (36) from (35), recognizing $\phi(x) = \phi(-x)$, and dividing by $\|f-g\|_1 = 2(\Phi\left(\delta\right) - \Phi\left(-\delta\right))$ simplifies (33):

$$\mathbb{E}\left[X\right] = \frac{\mu_1 + \mu_2}{2} \tag{37}$$

We now evaluate X's second moment. Split the integral up over A and A^c:

$$\mathbb{E}\left[XX^T\right] = \frac{1}{\|f-g\|_1}\left(\int_A xx^T(f(x)-g(x))\,dx + \int_{A^c} xx^T(g(x)-f(x))\,dx\right) \tag{38}$$

Once again, we evaluate this expression by separately evaluating the integrals involving f and g.

Starting with f and using Lem. 6 with Δ and δ:

$$\int_A xx^Tf(x)\,dx = \Phi\left(\delta\right)(\Sigma + \mu_1\mu_1^T) + \phi\left(\delta\right)\left(\frac{\Delta\mu_1^T + \mu_1\Delta^T}{2\delta} - \delta\frac{\Delta\Delta^T}{4\delta^2}\right) \tag{39}$$

$$\int_{A^c} xx^Tf(x)\,dx = \Phi\left(-\delta\right)(\Sigma + \mu_1\mu_1^T) - \phi\left(\delta\right)\left(\frac{\Delta\mu_1^T + \mu_1\Delta^T}{2\delta} - \delta\frac{\Delta\Delta^T}{4\delta^2}\right) \tag{40}$$

Taking the difference of (39) and (40):

$$\int_A xx^Tf(x)\,dx - \int_{A^c} xx^Tf(x)\,dx = (\Phi\left(\delta\right) - \Phi\left(-\delta\right))(\Sigma + \mu_1\mu_1^T) + \frac{\phi\left(\delta\right)}{\delta}\left(\Delta\mu_1^T + \mu_1\Delta^T - \frac{\Delta\Delta^T}{2}\right) \tag{41}$$

To evaluate the integrals in (38) involving g, we can pattern match using (39) and (40). As in the derivation of (36), this involves negating the p terms and replacing μ_1 with μ_2.

$$\int_A xx^Tg(x)\,dx - \int_{A^c} xx^Tg(x)\,dx = (\Phi\left(-\delta\right) - \Phi\left(\delta\right))(\Sigma + \mu_2\mu_2^T) + \frac{\phi\left(\delta\right)}{\delta}\left(\Delta\mu_2^T + \mu_2\Delta^T + \frac{\Delta\Delta^T}{2}\right) \tag{42}$$

Note the sign difference of $\Delta\Delta^T$ compared to (41).

To finish calculating the second moment, subtract (42) from (41) and divide by $\|f-g\|_1$, simplifying (38)

$$\mathbb{E}\left[XX^T\right] = \frac{1}{2}(2\Sigma + \mu_1\mu_1^T + \mu_2\mu_2^T) + \frac{\phi(\delta)}{\delta\|f-g\|_1}\Delta\Delta^T \tag{43}$$

We can now express the covariance of X.

$$\mathbb{E}\left[XX^T\right] - \mathbb{E}\left[X\right]\mathbb{E}\left[X\right]^T = \Sigma + \frac{\Delta\Delta^T}{4} + \frac{\phi(\delta)}{\delta\|f-g\|_1}\Delta\Delta^T = \Sigma + \left(\delta^2 + \frac{2\phi(\delta)\delta}{2\Phi(\delta)-1}\right)\frac{\Delta\Delta^T}{4\delta^2} \tag{44}$$

Which follows as $(\mu_1\mu_1^T + \mu_2\mu_2^T)/2 - (\mu_1+\mu_2)(\mu_1+\mu_2)^T/4 = \Delta\Delta^T/4$.

To calculate the determinant of the covariance, factor Σ out of (44) and define $\alpha = \delta^2 + \frac{2\phi(\delta)\delta}{2\Phi(\delta)-1}$:

$$|\operatorname{cov} X| = \left|\Sigma\left(I + \alpha\frac{\Sigma^{-1}\Delta\Delta^T}{4\delta^2}\right)\right| = |\Sigma|\,(1+\alpha) \tag{45}$$

The last step follows from the eigenvalues. $\Sigma^{-1}\Delta\Delta^T$ only has one non-zero eigenvalue; it is a rank one matrix as Σ^{-1} is full rank and $\Delta\Delta^T$ is rank one. The trace of a matrix is the sum of its eigenvalues, so $\operatorname{tr}(\Sigma^{-1}\Delta\Delta^T) = \operatorname{tr}(\Delta^T\Sigma^{-1}\Delta) = \|\Delta\|_\Sigma^2 = 4\delta^2$ is the non-zero eigenvalue. Consequently, the only non-zero eigenvalue of $\alpha\Sigma^{-1}\Delta\Delta^T/4\delta^2$ is α. Adding I to a matrix increases all its eigenvalues by 1 so the only eigenvalue of $I+\Sigma^{-1}\Delta\Delta^T/4\delta^2$ that is not 1 has a value of $1+\alpha$. The determinant of a matrix is the product of its eigenvalues, so $\det(I+\Sigma^{-1}\Delta\Delta^T/4\delta^2) = 1+\alpha$. As discussed at the beginning of the proof, substituting (45) into the expression for the entropy of a multivariate Gaussian achieves the desired upper bound.

A.4 Proof of Theorem 2

To prove the theorem, we build on Thm. 1. However, it cannot be directly applied because it is difficult to evaluate 1) the L_1 norm between GMMs and 2) the entropy of the normalized difference of mixture models. However, from Lem. 1 and Lem. 2 provide a way to evaluate these quantities when the densities are individual Gaussians. To exploit this fact, we split the problem up and analyze the difference in entropies between GMMs that only differ by a single component.

To start, define $d_j(x) = \sum_{i=1}^{j} w_i g_i(x) + \sum_{i=j+1}^{n} w_i f_i(x)$. d_j is a GMM whose first j components are the first j components in g and last $n-j$ components are the last $n-j$ components in f. Note that $d_0 = f$ and $d_n = g$. Using the triangle inequality:

$$|\,\mathrm{H}[f] - \mathrm{H}[g]\,| = |\,\mathrm{H}[d_0] - \mathrm{H}[d_1] + \mathrm{H}[d_1] - \mathrm{H}[d_n]\,| \le |\,\mathrm{H}[d_0] - \mathrm{H}[d_1]\,| + |\,\mathrm{H}[d_1] - \mathrm{H}[d_n]\,| \le \sum_{j=1}^{N} |\,\mathrm{H}[d_{j-1}] - \mathrm{H}[d_j]\,| \tag{46}$$

where the last step applied the same trick $n-2$ more times. Because $d_{j-1} - d_j = w_j(f_j - g_j)$ we can use Thm. 1 more easily:

$$\begin{aligned}\sum_{j=1}^{N} |\,\mathrm{H}[d_{j-1}] - \mathrm{H}[d_j]\,| &\le \sum_{j=1}^{N} q(\|d_{j-1}-d_j\|_1) + \|d_{j-1}-d_j\|_1\,\mathrm{H}\left[\frac{|d_{j-1}-d_j|}{\|d_{j-1}-d_j\|_1}\right] && (47)\\ &= \sum_{j=1}^{N} q(\|w_j(f_j-g_j)\|_1) + \|w_j(f_j-g_j)\|_1\,\mathrm{H}\left[\frac{|f_j-g_j|}{\|f_j-g_j\|_1}\right] && (48)\end{aligned}$$

By definition, f_j and g_j are Gaussians whose covariances are equal and whose means differ by Δ_i. Consequently, both Lem. 1 and Lem. 2 can be applied to (48), which completes the proof.