Information-Theoretic Motion Planning for Constrained Sensor Networks

Daniel Levine[*], Brandon Luders[†], and Jonathan P. How[‡]
Laboratory for Information and Decision Systems (LIDS), MIT, Cambridge, MA

Abstract

This paper considers the problem of online informative motion planning for a network of heterogeneous sensing agents, each subject to dynamic constraints, environmental constraints, and sensor limitations. Prior work has not yielded algorithms that are amenable to such general constraint characterizations. In this paper, we propose the Information-rich Rapidly-exploring Random Tree (IRRT) algorithm as a solution to the constrained informative motion planning problem that embeds metrics on uncertainty reduction at both the tree growth and path selection levels. IRRT possesses a number of beneficial properties, chief among them being the ability to find dynamically feasible, informative paths on short timescales, even subject to the aforementioned constraints. The utility of IRRT in efficiently localizing stationary targets is demonstrated in a progression of simulation results with both single-agent and multiagent networks. These results show that IRRT can be used in real-time to generate and execute information-rich paths in tightly constrained environments.

1 Introduction

Mobile sensing agents often serve a crucial role in seeking out and gathering information for intelligence, surveillance, and reconnaissance (ISR) missions, both in military and civilian applications. Tasks performed by such agents might include target classification, target localization, mapping, and search and track, among others. Central to many of these tasks is the notion of collecting information to reduce uncertainty. By deploying a team of cooperative sensing agents, complex information collection tasks can be completed effectively and efficiently.

Provided that mobile sensing incurs resource costs to the operator, when evaluating agent plans, one typically seeks to both maximize the information content and minimize the cost. This motivation is central to the informative motion planning problem [12, 15, 24, 32, 35], in which plans consist of trajectories for dynamically constrained sensing agents. To accommodate a variety of sensor platforms, the selected informative motion planning algorithm should be amenable to nonholonomic and/or differential dynamic constraints. Obstacles present a further challenge in that they can both constrain the vehicle motion and occlude observations. Finally, the limitations inherent in available sensing mechanism (e.g., narrow field of view) should be accounted for when anticipating informativeness.

Prior research has addressed this problem by using receding horizon, optimization-based techniques for short-term control sequences [11, 25, 28] or heuristically shaped, simple paths that are known to give optimal results in unconstrained settings [2, 30]. However, the inclusion of general dynamic, environmental, or sensor constraints may hinder or even prohibit real-time solution generation for such approaches. Target tracking solutions using the partially observable Markov decision process (POMDP) framework [33], while having very general constraint characterizations, are currently intractable for vehicle models with complex and/or nonholonomic dynamics.

[*] Ph.D. candidate, Dept. of Aeronautics and Astronautics, MIT; denu@mit.edu

[†] Ph.D. candidate, Dept. of Aeronautics and Astronautics, MIT; luders@mit.edu

[‡] Richard C. Maclaurin Professor of Aeronautics and Astronautics, MIT; jhow@mit.edu

In this paper, the Information-rich Rapidly-exploring Random Tree (IRRT) algorithm is presented as an online solution method that affords very general constraint characterizations for the informative motion planning problem. IRRT extends the RRT algorithm [10, 18, 19] by embedding information collection, as predicted using Fisher information matrices [8], at the tree expansion, path selection, and path execution levels. As IRRT is a sample-based motion planner, feasible solutions can easily be generated online and in real-time, and the number of discovered feasible solutions scales well with the available computational resources. That it extends the RRT algorithm is central to the breadth of constraints under which IRRT can operate. Furthermore, IRRT may be used in decentralized settings for multiple sensing agents to cooperatively generate efficient, informative motion plans. Simulated results have demonstrated that IRRT can generate and execute informative, dynamically-feasible motion plans in what would otherwise be considered prohibitively constrained settings. Furthermore, IRRT produces complex target localization behaviors from a simple model of the trade-off between information collection and resource consumption.

The flexibility to generate informative motion plans in real-time and under general feasibility constraints and cost functions distinguishes the IRRT from the related prior research, which is surveyed in Section 2. The remainder of the paper is structured as follows. Section 3 states the informative motion planning problem, and Section 4 motivates the use of sample-based planners. The IRRT algorithm is then presented in Section 5. A progression of simulation results in Section 6 demonstrate the utility of IRRT for constrained sensor networks, culminating with multiple Dubins car agents cooperatively and efficiently maneuvering to localize targets in a cluttered environment.

2 Related Work

Several solution strategies to effect information-rich path planning have been considered in the past decade. Analytical solutions often use the Fisher information matrix (FIM) to quantify trajectory information collection in an optimal control framework [22, 26]. Solutions seek to maximize, for example, a lower bound on the determinant of the FIM [22] or the $\log\det$ of the FIM at the conclusion of the trajectory [26]. While analytical solutions often have a simple form and perform optimally for low-dimensional, unconstrained simple problems, they typically are not immediately scalable to complicated scenarios (e.g., sensor platform dynamics of order higher than two).

An opposite extremum of the analytical solution strategy is one that neglects transient localization behavior and adopts a heuristic path shape shown to perform well in steady state. Examples of such shapes are circles, ellipses, and spirals. For aerial platforms with side-mounted cameras, circular trajectories with optimal radii at fixed altitude [30] and varying altitude [2] have been proposed. While these heuristically constrained trajectories capture the physical and geometric intuition of bearings-only target tracking, that of reducing range and maximizing orthogonality of measurements, they artificially limit the performance of the informative motion planner when operating under realistic constraints.

Recent research has also considered trajectories constructed by performing receding-horizon control online. This class of solution strategies can be partitioned into discrete-space and continuous-space planners. The advent of the former involved a core robotics problem, simultaneous localization and mapping (SLAM), in which a vehicle must localize itself using measurements of features registered in a feature map of the environment, which is both constructed and refined online. Several prominent papers have addressed the SLAM-oriented problem of planning a vehicle path through the environment to maximize the information gain in a temporally local sense [4, 7, 36]; these strategies can be summarized as greedy, next-best-view methods that perform steepest ascent on the information. Realizing the need for information-theoretic multistep planners [13], Sim and Roy present a global planner for the SLAM framework which performs multistep planning as a pruned graph search [34]. However, as discrete-space methods require enumerating a subset of reachable states, vehicles with nontrivial, fundamentally continuous dynamic models would require a level of enumeration that may prove prohibitive.

Several continuous-space, receding-horizon planning strategies for generating information-rich trajectories have also been considered. Frew uses the determinant of the target estimation error covariance as the objective function in a trajectory-generating optimization for 2D ground robots with limited field of view [11]. Ousingsawat and Campbell formulate a receding horizon optimal control problem that attempts to maximize information, quantified using the FIM, while avoiding risk zones and satisfying terminal location and time constraints [25]. However, the results presented therein are limited to simple constraint sets: an omnidirectional sensor performs 2D target estimation, vehicles are modeled as point masses, and risk zones are elliptical. Ponda uses the A-optimality criterion (of the FIM) as the objective function to optimize the trajectory of a fixed-wing aircraft with a perfectly gimbaled camera [28]. The assumption of a gimbaled camera, together with the absence of obstacles, assumes the target to be visible from the entire flight volume. In reality, the existence of sensor occlusions limits the effectiveness of such a method. While many works embed a small set of apt constraints – e.g., Frew considers limited field of view sensing limitations [11], and Ponda explicitly handles dynamic constraints of the vehicle [28] – the available solution methods for such receding-horizon optimization problems are not extensible to the combination of sensor limitations, environmental constraints, and dynamic constraints often encountered in real-world scenarios. This is primarily due to the absence of receding-horizon methodologies that maintain tractability as the constraint set increases in size and/or complexity.

When generality is desired, the partially observable Markov decision process (POMDP) framework is widely acknowledged to be the most principled way of solving problems of planning under uncertainty with observations. Recent research has considered belief-space planning for both the target tracking problem and its inverse problem, that of localizing a vehicle through sensor measurements of perfectly known targets in a previously mapped environment. For example, He et al. use the Belief Roadmap (BRM) to plan vehicle trajectories that maximize the self-localization capability of a hovering vehicle operating in GPS-denied environments [12]. Using a prior map of the environment and the associated measurement samples for a laser range finder, a graph of the covariance propagation between samples can be formed, from which the BRM efficiently selects trajectories that mitigate egomotion drift and aid knowledge of goal arrival. Roy and He also explore forward-search in a POMDP framework to facilitate target tracking [33]. While the POMDP framework has shown promising results for simple vehicle models, POMDP solutions are currently intractable for vehicle models with complex dynamics. Furthermore, solving for the optimal policy over the set of reachable beliefs – using, for example, point-based value iteration (PBVI) [27] – is intractable for the observation models of interest and is not typically amenable to dynamically changing the dimension of the belief space, which would occur as targets are introduced into or dismissed from the mission.

3 Problem Statement

Consider a bounded, open set $X \subset \mathbb{R}^{d_x}$ partitioned into an obstacle region $X_{\text{obs}} \subset X$ and an obstacle-free region $X_{\text{free}} = X \setminus X_{\text{obs}}$. The obstacle map comprising X_{obs} is assumed to be available for use from either existing maps or a separate, offline mapping algorithm [6, 37], as it is not the focus of this work. Points in X_{obs} are said to be in collision, while those in X_{free} are said to be collision-free. Given an initial collision-free state $x_{\text{init}} \in X_{\text{free}}$ and a goal region $X_{\text{goal}} \subset X_{\text{free}}$, the feasible motion planning problem is to find a path $\sigma : [0, T] \to X_{\text{free}}$, i.e., that is collision-free at all points and that satisfies the specified initial state $\sigma(0) = x_{\text{init}}$ and terminal state $\sigma(T) \in X_{\text{goal}}$ constraints, for some $T \in \mathbb{R}_{>0}$. When many feasible paths exist, one is often concerned with finding the minimum-cost feasible path, where the cost function is assumed to be known; examples include Euclidean distance, control effort, elapsed time, and combinations thereof.

The obstacle-free region is traversable by a connected network $\mathcal{Q}$ of heterogeneous mobile sensing agents, where each agent $q \in \mathcal{Q}$ is assumed to have a fully observable state $\mathbf{x}^{[q]}$, carry a set of sensors $S^{[q]}$, and have a known model of resource consumption rates (i.e., cost function) for actions it can execute. The sensing platforms are subject to various dynamic and environmental constraints, and the individual sensors $s \in S^{[q]}$ are characterized by a set of limitations (e.g., narrow field of view). In addition to constraining vehicle motion, the obstacles comprising X_{obs} may occlude observability of measurements.

Suppose also the existence of a known number of independent features, each of which has a partially observable, continuous state $\mathbf{y}_i \in Y$, where $Y \subseteq \mathbb{R}^{d_y}$. We take a Bayesian perspective and model $\mathbf{y}_i$ as the realization of a random vector $\mathsf{y}_i \in Y$, for which we instantiate a prior belief $p_{\mathsf{y}_i}(\cdot)$. It is assumed that the set of features $\{\mathsf{y}_i | i \in I\}$ is independent from the parameterization of X_{obs}. Given a prior belief $p_{\mathsf{y}_i}(\cdot)$ and likelihood function $p_{\mathbf{z}|\mathbf{y}_i}(\mathcal{Z}|\cdot)$ for a set of observations $\mathcal{Z}$, the posterior belief $p_{\mathbf{y}_i|\mathbf{z}}(\cdot|\mathcal{Z})$ may be computed using Bayes' rule.

It is assumed that during the online estimation of $\mathbf{y}_i$, all beliefs $b_{\mathbf{y}_i}(\cdot)$ can be approximated by K_i-component Gaussian mixture models [3], i.e.,

$$b_{\mathbf{y}_i}(\cdot) \approx \sum_{k=1}^{K_i} w_{i,k}\ \mathcal{N}(\cdot; \mu_{i,k}, \Lambda_{i,k}), \qquad \sum_{k=1}^{K_i} w_{i,k} = 1, \quad w_{i,k} > 0, \tag{1}$$

where $\mu_{i,k}$ and $\Lambda_{i,k}$ are the mean vector and covariance matrix, respectively, of the kth mixture component when estimating $\mathbf{y}_i$. For such Gaussian mixture models, the *mixture* mean μ_i and covariance Λ_i are, respectively,

$$\mu_i = \sum_{k=1}^{K_i} w_{i,k}\ \mu_{i,k}, \tag{2}$$

$$\Lambda_i = \sum_{k=1}^{K_i} w_{i,k} \left[\Lambda_{i,k} + (\mu_i - \mu_{i,k})(\mu_i - \mu_{i,k})^T\right]. \tag{3}$$

For concreteness, we will refer in the remainder of this section and throughout this paper to the example problem of target localization, in which a partially observable state $\mathbf{y}_i \in Y$ represents the location of target i in physical space. However, the solution strategy proposed by this paper is extensible to other, more general uncertainty reduction problems. We will proceed by first specializing to the stationary target case and then later discussing the implications of localizing nonstationary targets.

The objective for agent q is to safely arrive at its goal region $X_{\text{goal}}^{[q]}$, via $\sigma^{[q]}$, with minimum time and with minimum residual uncertainty on the target location beliefs $b_{\mathbf{y}_i}$, where the uncertainty metric is a convex combination $\sum_i \gamma_i f(\Lambda_i)$ of scalar functions $f(\cdot)$ on target covariances Λ_i, with $\sum_i \gamma_i = 1$ and the $\gamma_i > 0$ known. These competing objectives necessitate a trade-off between information collection and timely goal arrival.

For the constrained sensor problems of interest, the optimal solution is unlikely to be found, and the objective becomes identifying the least suboptimal feasible solution. Furthermore, for such problems, feasibility of sensor platform configurations is paramount. Our solution strategy is predicated on rapidly and continually identifying a collection of feasible paths, while executing the minimum-cost feasible path from that collection.

4 Motion Planning with RRTs

Before presenting the solution to the informative motion planning problem, this section highlights several properties of path planning algorithms used within this work. For the general motion planning problem, it has been shown that sample-based methods, which seek to approximate connectivity in X_{free} by randomly sampling configurations and checking feasibility with respect to X_{obs}, have several particularly desirable properties for complex, real-time planning problems [17, 38]. By construction, sample-based methods avoid issues of discretization granularity and are amenable to planning in high-dimensional state spaces. Furthermore, the performance of sample-based methods scale well with the available computational resources. Finally, the trajectory-wise constraint satisfaction afforded by sample-based methods leads to a significant reduction in computational complexity over that of standard optimization routines [17, 38].

Within the class of sample-based methods, the Probabilistic Roadmap (PRM) [14] and Rapidly-exploring Random Tree (RRT) [18, 19] algorithms have been used extensively in the motion planning literature. The latter is especially effective for planning on nonholonomic and/or differentially constrained vehicles. Because the dynamics of sensor platforms are typically nontrivial, we build on the RRT as a baseline algorithm that is amenable to general vehicle constraints in the informative motion planning problem.

The RRT retains a tree-structured graph $\mathcal{T}$ of nodes emanating from a root node n_{root} and attempting to connect the collision-free space X_{free}. Each node $n \in \mathcal{T}$ is a tuple $n = \langle T_n, \sigma_n, v_n \rangle$, where $T_n \in \mathbb{R}_{>0}$ is the node duration, $\sigma : [0, T_n] \to X$ is the node state trajectory mapping instants in time to points in the state space X, and $v_n = \sigma_n(T_n) \in X$ is the terminal state, or *waypoint*.

4.1 Tree Expansion

Let $\rho : X \times X \to \mathbb{R}_{\geq 0}$ be a distance metric comparing two states in X, $\texttt{Sample}(\cdot)$ be a function that generates random samples $\mathbf{x}_{\text{samp}} \sim \texttt{Sample}(\cdot)$ in the environment,[1] and $\texttt{Nearest} : X \times 2^{\mathcal{T}} \to 2^{\mathcal{T}}$ be a function that returns a subset of nodes $N_{\text{near}} \subset \mathcal{T}$ that are nearest to a specified state $\mathbf{x}_{\text{samp}}$ as measured by ρ. Expansion of the tree $\mathcal{T}$ proceeds by generating $\mathbf{x}_{\text{samp}} \sim \texttt{Sample}(\cdot)$, computing $N_{\text{near}} \coloneqq \texttt{Nearest}(\mathbf{x}_{\text{samp}}, \mathcal{T})$, and attempting to "connect" nodes in N_{near} to $\mathbf{x}_{\text{samp}}$ using an as yet unspecified function $\texttt{Steer}$. For ease of discussion, this paper focuses on the attempt to steer from one node $n_{\text{near}} \in N_{\text{near}}$ towards $\mathbf{x}_{\text{samp}}$, considering $v_{n_{\text{near}}}$ as the waypoint for this near node.

In open-loop RRT [18] (denoted here as OL-RRT), $\texttt{Steer}(v_{n_{\text{near}}}, \mathbf{x}_{\text{samp}}, T)$ samples input profiles $u : [0, T] \to \mathcal{U}$ from an input space $\mathcal{U}$ over some finite duration $T \in \mathbb{R}_{>0}$, simulates the resulting state sequences $\sigma_u : [0, T] \to X$, and selects from them the feasible sequence $\sigma : [0, T] \to X_{\text{free}}$ that terminates nearest to $\mathbf{x}_{\text{samp}}$ as measured by ρ. A new node $n = \langle T, \sigma, v \rangle$ is then instantiated, with $v \coloneqq \sigma(T)$, and added to the tree, i.e., $\mathcal{T} \coloneqq \mathcal{T} \cup \{n\}$.

For dynamical systems with closed-loop control, open-loop RRT generates paths that may not be executable. In closed-loop RRT (CL-RRT) [10,16], the state vector can be thought of as a concatenation of the dynamic and reference states of the vehicle. Instead of sampling open-loop input profiles, the reference states between $v_{n_{\text{near}}}$ and $\mathbf{x}_{\text{samp}}$ are connected (often by a simple guidance law), and the full closed-loop response σ of the vehicle and controller in response to the reference is generated. The termination criterion is that the projection of the *reference* state onto X is within some distance of its counterpart at the sampled location. As before, if $\sigma : [0, T] \to X_{\text{free}}$ for its duration $T \in \mathbb{R}_{>0}$, a node is instantiated and added to the tree. In addition to its executability properties, closed-loop RRT affords a notably accurate prediction of the state trajectory resulting from following a reference path [23], making it well suited as a baseline algorithm for informative motion planning, where the measurement poses along paths must be predicted accurately.

Though we extend CL-RRT rather than OL-RRT, the two differ most fundamentally in the tree expansion phase. In the remainder of this work, we will maintain an agnostic view of RRTs and use the generic function $\texttt{Steer} : (\mathbf{x}, \mathbf{x}'', \Delta t) \mapsto \mathbf{x}'$ to denote the forward simulation from state $\mathbf{x}$ towards $\mathbf{x}''$ over Δt seconds, resulting in state $\mathbf{x}'$; we further use $\texttt{Reached}(\mathbf{x}, \mathbf{x}'', t) \in \{0, 1\}$ as an indicator function denoting whether $\mathbf{x}$ and $\mathbf{x}''$ meet closeness criteria, or alternatively whether $t \in \mathbb{R}_{>0}$ exceeds some threshold.

4.2 Path Selection

In describing the path selection process used in RRTs, it is useful to define several operators on nodes in the tree. Let the root operator $\texttt{root} : \mathcal{T} \mapsto n_{\text{root}}$ return the root node. Due to the tree structure of the graph $\mathcal{T}$, the path connecting n_{root} to any other node $n \in \mathcal{T}$ is unique. Therefore, let the parent operator $\texttt{pa} : \mathcal{T} \to \mathcal{T}$ map a node to its parent, with $\texttt{pa}(n_{\text{root}}) \coloneqq n_{\text{root}}$. By telescoping, let the ancestor operator $\texttt{anc} : \mathcal{T} \to 2^{\mathcal{T}}$ map a node to the set of its ancestors (i.e., all parents of parents back to n_{root}). Likewise, one can define a children operator $\texttt{chi} : \mathcal{T} \to 2^{\mathcal{T}}, \{n\} \mapsto \{n' \in \mathcal{T} | \texttt{pa}(n') = n\}$ mapping a node to the set of nodes for which it is a parent. Finally, let the path operator $P : \mathcal{T} \to 2^{\mathcal{T}}, \{n\} \mapsto \texttt{anc}(n) \cup \{n\}$ return the union of any particular node with its ancestors. The set of all paths contained in the tree is defined as $\mathcal{P} \triangleq \{P(n) | n \in \mathcal{T}\}$.

The cost function $c : \mathcal{P} \to \mathbb{R}_{\geq 0}$ assigns nonnegative, real-valued cost to all paths in the tree. The best path is denoted by $P_* = P(n^*)$, with $n^* = \operatorname{argmin}_{n \in \mathcal{T}} c\,(P(n))$. Often, the cost of a path $P(n)$ is composed of individual costs for each node $n' \in P(n)$, as captured by the nodal path cost function $\psi : \mathcal{T} \to \mathbb{R}_{\geq 0}$, and a cost-to-go term of reaching X_{goal} from the terminal waypoint v_n.

[1] The proviso for an admissible function $\texttt{Sample}(\cdot)$ is that X is uniformly sampled with positive probability.

5 Information-rich RRTs

The problem as stated in Section 3 is that of adaptive sampling, a sequential decision problem in which a set of sensing configurations must be selected, given all previous information, to best reduce the uncertainty about an unknown quantity $\mathbf{y}_i$. Given that it makes informativeness explicit in the decision problem, adaptive sampling can significantly reduce the cost of obtaining an observation set that achieves a satisfactory estimate or classification.

Suppose that by traversing a path σ, a set of observations $\mathcal{Z}_\sigma$ are stochastically generated according to the likelihood function $p_{\mathbf{z}|\mathbf{y}_i}(\mathcal{Z}_\sigma|\cdot)$. One would wish to assess the informativeness of σ via metrics on the posterior belief $p_{\mathbf{y}_i|\mathbf{z}}(\cdot|\mathcal{Z}_\sigma)$. However, one cannot anticipate the exact realized observation sequence that will result from future traversal of a candidate path. The objective of informative motion planning then is to quantify the potential uncertainty reduction of the *measurement configuration* set M_σ and to embed such information metrics in the planning problem, either as constraints or cost components.

This section presents the Information-rich Rapidly-exploring Random Tree (IRRT) algorithm [20,21], an extension of CL-RRT that constructs dynamically feasible paths for sensing platforms and predicts uncertainty reduction via the Fisher information of measurement configuration sets along such paths.

One of the strengths of this approach is its natural extension to decentralized, multiagent settings. Suppose each agent q in a connected network $\mathcal{Q}$ is described by a tuple $q = \langle \mathcal{T}^{[q]}, P_*^{[q]}, S^{[q]} \rangle$ with a tree of collision-free nodes $\mathcal{T}^{[q]}$, a selected path $P_*^{[q]}$ that it executes, and a set of sensors $S^{[q]}$. It is assumed that each sensor has a specified measurement interarrival time,[2] from which the observation times within each node may be anticipated. There also exists for each sensor $s \in S^{[q]}$ an indicator function $o_s : X \times Y \to \{0,1\}$ that captures whether s can generate an observation given a measurement pose $\mathbf{x} \in X$ and target state $\mathbf{y} \in Y$. Note that this observability indicator function subsumes a variety of possible sensor limitations, e.g., narrow field of view (FOV), limited range, and occlusions due to the presence of obstacles in X_{obs}.

5.1 Measurement Pose Prediction

To quantify the informativeness of paths in tree $\mathcal{T}^{[q]}$, the measurement pose sequence for each node is first predicted. Consider a single node n, which is described by a state trajectory $\sigma_n : [0, T_n] \to X$ of duration $T_n \in \mathbb{R}_{>0}$. The measurement interarrival times and temporal offsets (due to measurement in $\mathtt{pa}(n)$) of sensors in $S^{[q]}$ are used to generate a set M_n of measurement configuration tuples. Each element $m \in M_n$ is a tuple $m = \langle t_m, \mathbf{x}_m, s_m \rangle$ composed of the intranode measurement time $t_m \in [0, T_n]$, the measurement pose $\mathbf{x}_m = \sigma_n(t_m) \in X$, and the utilized sensor $s_m \in S^{[q]}$. In subsequent discussion, the process of anticipating a measurement pose sequence will be referred to by the generic function $\mathtt{MeasurementPoses}(\sigma, S)$.

5.2 Information Quantification

We now wish to quantify the informativeness of a node given its measurement pose sequence. Many information-theoretic measures exist for such a quantification [5]; we use the Fisher information metric [8]

$$J_{\mathbf{z}}(\mathbf{y}_i) \triangleq \mathbb{E}_{\mathbf{z}} \left\{ \nabla_{\mathbf{y}_i} \left[\nabla_{\mathbf{y}_i} \log p_{\mathbf{y}_i,\mathbf{z}}(\mathbf{y}_i, \mathbf{z}) \right] \right\}, \tag{4}$$

for its connection to the Cramér-Rao lower bound (CRLB) on the error covariance of unbiased estimators [31]. Indeed, the Cramér-Rao matrix is exactly the inverse of the Fisher information matrix (FIM). In general, the update for the *approximate* Fisher information $J_n(\hat{\mathbf{y}}_i)$ of target i across a node n is a function of the FIM at the parent node $J_{\mathtt{pa}(n)}$, the measurement pose sequence M_n, and the current belief $b_{\mathbf{y}_i}$, i.e.,

$$J_n(\hat{\mathbf{y}}_i) := \mathtt{FisherInformation}\left(J_{\mathtt{pa}(n)}(\hat{\mathbf{y}}_i),\ M_n,\ b_{\mathbf{y}_i} \right). \tag{5}$$

Hereafter, this paper specializes to the case where the targets are stationary, beliefs have a K_i-mode Gaussian mixture model distribution, and the observation model is linearizable-Gaussian for each target,

[2] More generally, one might consider that for each sensor, a known schedule dictates when that sensor may make observations.

i.e.,

$$\mathbf{z}_{m,i} = h(\mathbf{x}_m, \mathbf{y}_i) + \xi_{m,i}, \quad \xi_{m,i} \sim \mathcal{N}(\mathbf{0}, R_{s_m}), \tag{6}$$

where $h(\cdot,\cdot)$ is the vector-valued observation function with local linearization $H(\cdot,\cdot)$, which we refer to as the linearized measurement matrix. The strategy will be to maintain separate FIMs for each mode of target i and fuse them together in computing an uncertainty-based cost. Therefore, we consider $\mathbf{y}_{i,k} \sim \mathcal{N}(\mu_{i,k}, \Lambda_{i,k})$ for the ith target's kth mode and note that the update for the FIM across node n is [32]

$$J_n(\hat{\mathbf{y}}_{i,k}) \coloneqq J_{\mathtt{pa}(n)}(\hat{\mathbf{y}}_{i,k}) + \sum_{m \in M_n} o_{s_m}(\mathbf{x}_m, \hat{\mathbf{y}}_{i,k}) H^T_{s_m}(\mathbf{x}_m, \hat{\mathbf{y}}_{i,k}) R^{-1}_{s_m}(\mathbf{x}_m, \hat{\mathbf{y}}_{i,k}) H_{s_m}(\mathbf{x}_m, \hat{\mathbf{y}}_{i,k}), \tag{7}$$

where, for sensor s_m, $H_{s_m}(\mathbf{x}_m, \hat{\mathbf{y}}_{i,k})$ is the linearized measurement matrix about the measurement pose $\mathbf{x}_m$ and estimated kth modal mean $\hat{\mathbf{y}}_{i,k} = \mu_{i,k}$, $R_{s_m}(\mathbf{x}_m, \hat{\mathbf{y}}_{i,k})$ is the measurement noise covariance matrix, and $o_{s_m} : X \times Y \to \{0,1\}$ is the previously defined indicator function that captures whether s_m can generate an observation given $\mathbf{x}_m$ and $\hat{\mathbf{y}}_{i,k}$.

In the single-agent case, if one defines

$$J_0(\hat{\mathbf{y}}_{i,k}) \triangleq \Lambda^{-1}_{i,k} = \mathbb{E}\left[\left(\hat{\mathbf{y}}_{i,k} - \mathbf{y}_{i,k}\right)\left(\hat{\mathbf{y}}_{i,k} - \mathbf{y}_{i,k}\right)^T\right]^{-1}, \tag{8}$$

the recursion (7) initiates at the root node n_{root} with $J_{n_{\text{root}}}(\hat{\mathbf{y}}_{i,k}) \coloneqq J_0(\hat{\mathbf{y}}_{i,k})$, an exact relationship that arises from the Gaussianity of $\mathbf{y}_{i,k}$. Finally, let us denote the set of all FIMs (i.e., over all (i,k) pairs) on node n as $\mathbf{J}_n$.

Due to the additivity of (7), the extension of IRRT to the multiagent case, which accounts for the information content of paths selected by other agents in the network, is straightforward. For any agent $q \in \mathcal{Q}$, its *information contribution* along path $P(n)$ is denoted by

$$\Delta\mathbf{J}^{[q]}\left(P(n)\right) \triangleq \mathbf{J}^{[q]}_n - \mathbf{J}^{[q]}_{n_{\text{root}}}, \tag{9}$$

where operations on the FIM sets are elementwise (i.e., for each existing (i,k) pair). Therefore, factoring in the information contribution of other agents in the network is equivalent to initiating the FIM recursion at the root node as

$$\mathbf{J}^{[q]}_{n_{\text{root}}} = \mathbf{J}_0 + \sum_{q' \in \mathcal{Q}\setminus\{q\}} \Delta\mathbf{J}^{[q']}\left(P^{[q']}_*\right). \tag{10}$$

Fisher information is generally a matrix quantity, so one further requires a scalar cost function to operate on Fisher information matrices J; many such cost functions exist [39]. For computational and geometric reasons [29], we choose to use the A-optimality criterion $\mathtt{trace}(J^{-1})$ to assign cost to uncertain states, thereby rewarding uncertainty reduction. The *information error* with respect to $\mathbf{y}_i$ at the terminus of node n is defined as

$$\mathcal{I}_i(n) = \mathtt{trace}\left(\sum_{k=1}^{K_i} w_{i,k} J^{-1}_n(\hat{\mathbf{y}}_{i,k})\right), \qquad \sum_{k=1}^{K_i} w_{i,k} = 1, \qquad w_{i,k} > 0, \tag{11}$$

where the relative modal weights $w_{i,k}$ are exactly those maintained by the estimator running online. Note that the term within the $\mathtt{trace}$ operator is an approximation of the mixture covariance that does not consider the second set of terms in (3). The overall information error for node n is a convex combination of the errors for each target

$$\mathcal{I}(n) = \sum_i \gamma_i \mathcal{I}_i(n), \qquad \sum_i \gamma_i = 1, \quad \gamma_i > 0, \tag{12}$$

where the coefficients γ_i may denote the relative importance of each target i in the mission.

Algorithm 1 IRRT-Expand($\mathcal{T}^{[q]}$)

```
$\mathbf{x}_{\text{samp}} \sim \texttt{Sample}(\cdot)$
$n_{\text{near}} \leftarrow \texttt{Nearest}(\mathbf{x}_{\text{samp}}, \mathcal{T}^{[q]})$
$\mathbf{x} \leftarrow v_{n_{\text{near}}}$
$t \leftarrow 0$
while $\mathbf{x} \in X_{\text{free}} \wedge \neg\texttt{Reached}(\mathbf{x}, \mathbf{x}_{\text{samp}}, t)$ do
    $\bar{\sigma}(t) \leftarrow \mathbf{x}$
    $\mathbf{x} \leftarrow \texttt{Steer}(\mathbf{x}, \mathbf{x}_{\text{samp}}, \Delta t)$
    $t \leftarrow t + \Delta t$
end while
if $\mathbf{x} \in X_{\text{free}}$ then
    $\bar{\sigma}(t) \leftarrow \mathbf{x}$
    $T \leftarrow t;\quad \sigma \leftarrow \bar{\sigma};\quad v \leftarrow \sigma(T)$
    $M \leftarrow \texttt{MeasurementPoses}(\sigma, S^{[q]})$
    $\mathbf{J} \leftarrow \texttt{FisherInformation}(\mathbf{J}_{n_{\text{near}}}, M, b_{\mathbf{y}})$
    $n \leftarrow \langle T, \sigma, v, M, \mathbf{J} \rangle$
    $\mathcal{T}^{[q]} \leftarrow \mathcal{T}^{[q]} \cup \{n\}$
end if
```

Algorithm 2 IRRT-Execute(q)

```
$n_{\text{init}} \leftarrow \left\langle 0, \{\mathbf{x}_0^{[q]}\}, \mathbf{x}_0^{[q]}, \emptyset, \mathbf{J}_0^{[q]} \right\rangle$
$\mathcal{T}^{[q]} \leftarrow \{n_{\text{init}}\}$
$\mathbf{x}^{[q]} \leftarrow \mathbf{x}_0^{[q]}$
while $\mathbf{x}^{[q]} \notin X_{\text{goal}}^{[q]}$ do
    Update $\mathbf{x}^{[q]}$ and target beliefs $b_{\mathbf{y}}$
    while time remaining in selection cycle do
        IRRT-Expand($\mathcal{T}^{[q]}$)
    end while
    UpdateInformation($b_{\mathbf{y}}$, $\texttt{root}(\mathcal{T}^{[q]})$)
    $n^* \leftarrow \text{argmin}_{n \in \mathcal{T}^{[q]}}\, c(P(n))$
    Announce $P^{[q]}(n^*)$ to network, and execute it
end while
```

5.3 Revised Algorithms

Now that the construction of the measurement sequence M_n and FIM set $\mathbf{J}_n$ along node n has been elucidated, one may expand the notion of the node tuple to $n = \langle T_n, \sigma_n, v_n, M_n, \mathbf{J}_n \rangle$. Furthermore, we now revise the standard RRT algorithm descriptions to account for informativeness of paths in the tree.

The method IRRT-Expand($\mathcal{T}^{[q]}$) for expanding the tree is given in Algorithm 1. The function `Nearest` here subsumes several of the nearest node heuristics previously motivated in [21]. Given a tree $\mathcal{T}^{[q]}$ of candidate paths, the method IRRT-Execute given in Algorithm 2 will continually expand the tree, select the best path $P_*^{[q]}$, and execute a portion of it until the next selection cycle. The cost function $c : \mathcal{P}^{[q]} \rightarrow \mathbb{R}_{\geq 0}$ it uses is of the form

$$c(P^{[q]}(n)) = \tilde{\rho}(v_n, X_{\text{goal}}^{[q]}) + \alpha \sum_{n' \in P^{[q]}(n)} \psi(n') + \beta \mathcal{I}(n), \tag{13}$$

where $\tilde{\rho}(\cdot, A) \triangleq \min_{a \in A} \rho(\cdot, a)$ is the distance metric representing a cost-to-go, $\psi(\cdot)$ is some node trajectory cost, $\mathcal{I}(\cdot)$ is the information-based cost (12), and $\alpha \in [0, 1]$ and $\beta \in \mathbb{R}_{\geq 0}$ are weights capturing the relative

Algorithm 3 UpdateInformation($b_{\mathbf{y}}, n$)

```
$\mathbf{J}_n \leftarrow \texttt{FisherInformation}(\mathbf{J}_{\texttt{pa}(n)}, M_n, b_{\mathbf{y}})$
for all $n' \in \texttt{chi}(n)$ do
    UpdateInformation($b_{\mathbf{y}}, n'$)
end for
```

importance of timely goal arrival and uncertainty reduction. In our case, $\psi(n') = T_{n'}$, i.e., the node path cost is simply the node trajectory duration.

As the beliefs about the target locations are updated online, the measurement pose sequences along nodes will not change. However, target belief updates can lead to a change in informativeness, due to factors such as changes in occlusion/observability and range to the target. The most current belief is cached over a planning cycle and is used to recompute the Fisher information at all nodes in the tree according to the recursive function UpdateInformation described in Algorithm 3.

5.4 IRRT Properties

IRRT has several useful properties that make it a particularly appealing solution to the constrained informative motion planning problem. As a sample-based algorithm, IRRT avoids issues of discretization granularity, and is amenable to planning in high-dimensional state spaces. Feasible solutions are empirically found on short timescales, and the number of discovered feasible solutions scales well with the available computational resources. Furthermore, as an algorithm that extends RRT, IRRT is suitable for planning on nonholonomic and/or differentially-constrained sensor platforms. When implemented to extend Closed-Loop RRT (CL-RRT) [10, 16], as is done in Section 6, IRRT is imbued with notably accurate prediction of the sensing platform's state trajectory [23], and thus accurate prediction of measurement pose sequences along candidate paths.

IRRT accommodates a network $\mathcal{Q}$ of heterogeneous sensing agents, where the heterogeneity can be in both mobility and sensing capabilities. Agent $q \in \mathcal{Q}$ is equipped with sensor bundle $S^{[q]}$, where each sensor $s \in S^{[q]}$ is characterized by its linearized measurement matrix function $H_s : X \times Y \to \mathbb{R}^{d_s \times d_y}$, linearized sensing noise matrix function $R_s : X \times Y \to \mathbb{R}^{d_s \times d_s}$, and binary observability indicator function $o_s : X \times Y \to \{0, 1\}$, where d_s is the dimension of the measurement vector for each observation taken by sensor s. The functions H_s, R_s, and o_s subsume all sensor parameters (e.g., mounting orientation on the sensing platform of interest). In particular, all observability loss is captured by o_s.

When IRRT is extended to incorporate CL-RRT for the multiagent setting, the communication bandwidth requirement depends only on the length of announced plans and the dimension d_x of X. If each agent $q \in \mathcal{Q}$ is assumed to possess a model (including dynamics and sensors) of the other agents in $\mathcal{Q} \backslash \{q\}$, then that agent need only transmit the waypoint set $\{v_n | n \in P_*^{[q]}\}$, which is of order $\mathcal{O}(d_x \cdot |P_*^{[q]}|)$, for each announcement. The full state trajectory that would result from this announced path, and consequently the measurement pose sequence and associated information contribution, may be reconstructed by other agents and incorporated into their planning. Additionally, by allowing the informativeness of teammate plans to be recomputed locally, IRRT is self-consistent with respect to differing beliefs between agents that could arise online during decentralized belief consensus [9].

Finally, we note that in the stationary target case, the total storage requirement for FIMs in tree $\mathcal{T}^{[q]}$ is $\mathcal{O}\left(\frac{d_y(d_y+1)}{2} \cdot |\mathcal{T}^{[q]}| \cdot \sum_{i=1}^{I} K_i\right)$. The complexity of computing these FIMs is similarly linear in the total number of measurement poses $\sum_{n \in \mathcal{T}^{[q]}} |M_n|$. We note that the complexity of storage and computation for the FIMs along the selected paths $P_*^{[q']}$ for all other agents $q' \in \mathcal{Q}$ is dominated whenever $|\mathcal{T}^{[q]}| \gg |\mathcal{Q}|$, which is typically the regime of interest.

5.5 Nonstationary Targets

As an aside, we now briefly consider the implications of extending the algorithm as hitherto presented to the case of nonstationary targets. Specifically, we contrast the tree information quantification exercise in the single-agent and multiagent regimes.

In the single-agent case, the extension to nonstationary targets requires revisiting (5). At some node $n \in \mathcal{T}$, the set of FIMs for each target i may be updated according to the FIMs of its parent node $\texttt{pa}(n)$, the node measurement configuration sequence M_n, and the cached belief $b_{\mathbf{y}_i}$ on the target location. Some model must be incorporated to account for the evolution of the belief at future times, due to factors such as the target dynamics, process noise, and estimated higher-order motion. In general, such models do not result in the FIM additivity seen in (7) for the case of stationary targets with linearizable-Gaussian observations. Instead, one must alternately propagate the belief and incorporate measurements from M_n, whose elements are ordered by timestamp. The covariance of the belief $b_{\mathbf{y}_i}$ will evolve approximately according to the FIMs and process models along node n; however, the predicted measurement configuration sequence M_n will not modify the prediction of the belief mean $\hat{\mathbf{y}}_i$ at future times, which depends on the actual realized observation sequence $\mathcal{Z}_n$. Note that one nonstationary model with an analytical expression for the approximate FIM update is a linearizable-Gaussian plant with a linearizable-Gaussian observation model [1].

In the case of multiple sensing agents localizing nonstationary targets, the informativeness of other agents' plans is not generally condensable to the form in (9) and (10). Instead, one must consider the *joint cumulative* measurement sequence due to all agents along their announced paths. Suppose each agent $q \in \mathcal{Q}$ announces its selected path as $P_*^{[q]}$. The *cumulative* measurement sequence $\tilde{M}_*^{[q]}$ is defined along such a path $P_*^{[q]}$ as

$$\tilde{M}_*^{[q]} \triangleq \bigcup_{n' \in P_*^{[q]}} M_{n'} \tag{14}$$

and leading up to an arbitrary node $n \in \mathcal{T}^{[q]}$ as

$$\tilde{M}_n^{[q]} \triangleq \bigcup_{n' \in P^{[q]}(n)} M_{n'}. \tag{15}$$

Then, for any node $n \in \mathcal{T}^{[q]}$, the *joint cumulative* measurement sequence $\tilde{M}_n^{\mathcal{Q}}$, which accounts for the measurement configurations in other agents' announced plans, is defined as

$$\tilde{M}_n^{\mathcal{Q}} \triangleq \tilde{M}_n^{[q]} \bigcup \left(\bigcup_{q' \in \mathcal{Q} \setminus \{q\}} \tilde{M}_*^{[q']} \right). \tag{16}$$

The joint cumulative measurement sequence must be used to compute Fisher information matrices set at every node, resulting in a marked increase in required computation for evaluating the information-based cost function at every node in the tree. Note, however, that one may still reuse computations from parent nodes in quantifying the information of child nodes. Let $\tilde{T}_n := \sum_{n' \in P(n)} T_{n'}$ be the cumulative path duration to reach the end of node $n \in \mathcal{T}^{[q]}$. Similarly, let $\tilde{t}_m$ be the predicted elapsed time after which measurement $m \in \tilde{M}_n^{\mathcal{Q}}$ will occur, for some $n \in \mathcal{T}$. Then the *truncated* set of measurement configurations $\bar{M}_n^{\mathcal{Q}} \triangleq \{m \in \tilde{M}_n^{\mathcal{Q}} : \tilde{t}_m \leq \tilde{T}_n\}$ need only be processed once for all children $n' \in \texttt{chi}(n)$. Therefore, by storing *two* sets of FIMs at every node $n \in \mathcal{T}$, one set corresponding to $\tilde{M}_n^{\mathcal{Q}}$ and one set corresponding to $\bar{M}_n^{\mathcal{Q}}$, a significant savings in computation can be achieved in the case of multiple sensing agents localizing a nonstationary target.

6 Results

The following simulation results demonstrate the effectiveness of the IRRT algorithm in managing the competing objectives of information-gathering and prompt goal arrival in real-time, while satisfying a complex

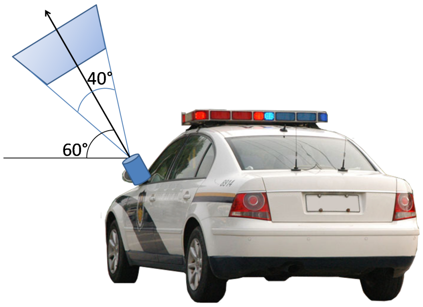


Figure 1: Dubins car diagram with a pointed monocular vision sensor.

constraint set. Increasingly complex problems (in line with Section 3) are considered, including nonholonomic vehicles, limited-FOV sensing, cluttered obstacle environments, multiple targets, three dimensions, and finally multiple sensing agents. Even subject to these constraints, which render many existing approaches in the literature intractable, the IRRT generates paths with emergent information-gathering characteristics. Some basic analysis is also provided to illuminate the trade-off between information-gathering and goal arrival that is taking place.

All simulations were performed using a real-time implementation of the IRRT algorithm in Java, run on an Intel 2.53 GHz quad-core laptop with 3.48 GB of RAM. The vehicle's current path is selected from the tree at a rate of 4 Hz; the tree capacity is specified to be 2000 nodes. In these simulations, all sensor measurements are assumed to be bearings-only; polar coordinate extended Kalman filters (EKFs) are used for online estimation of target positions, with measurements processed at a rate of 15 Hz.

6.1 Dubins Car Scenario

This scenario concerns a vehicle with nonholonomic dynamics and sensing constraints. Consider a small Dubins car agent traversing an obstacle-free environment while estimating the location of a stationary aerial target. The agent's monocular sensor is limited to a field of view of 40° in each of the horizontal and vertical axes. The sensor is yawed 90° (out the driver's left side) and pitched up by 60° from the horizontal plane (Figure 1); thus, the agent must achieve a proper combination of lateral distance and heading to see the target.

In this scenario, the car (radius 20 cm) begins at $\mathbf{x}^{[q]}(t_0) = [-2.5, -3.5, 1.0]^T$ m with an unobstructed path to the goal at $X_{\text{goal}} = \{[-2.5, 3.5, 1.0]^T \text{ m}\}$, but also uncertain knowledge of a target located at $\mathbf{y} = [0, 0, 2.0]^T$ m. For this scenario, we have selected the cost function parameters to be $\alpha = 0.5$ and $\beta = 8000$ s/m^2. The car is assumed to move at a fixed velocity of 0.4 m/s; a variation of the pure pursuit reference law (cf. [16]) is applied for steering control, assuming forward direction only. Note that this vehicle model could also be used to represent a fixed-wing vehicle (e.g., airplane) operating at a fixed velocity and altitude.

A typical trajectory generated by a trial of this scenario is given in Figure 2. The agent quickly identifies a winding path that both anticipates a visible observation sequence about the estimated target position and reaches the goal (Figure 2(a)). The uncertainty ellipsoid is markedly elongated in the line-of-sight direction, indicating large uncertainty in range. As the estimate improves (Figures 2(b)-2(d)), the planned path tightens around the estimated target position, in order to take an extended sequence of observations at close range. Given the relatively high value of β – the parameter governing the trade-off between information collection and resource expenditure – the path ultimately encircles itself (Figure 2(e)) in order to take additional measurements before finally turning toward the goal (Figure 2(f)). Though the set of vehicle states for which observations are unoccluded is never explicitly constructed or computed, the IRRT algorithm is able to sample from these regions and execute a path that spends significant time gathering useful information within those regions.

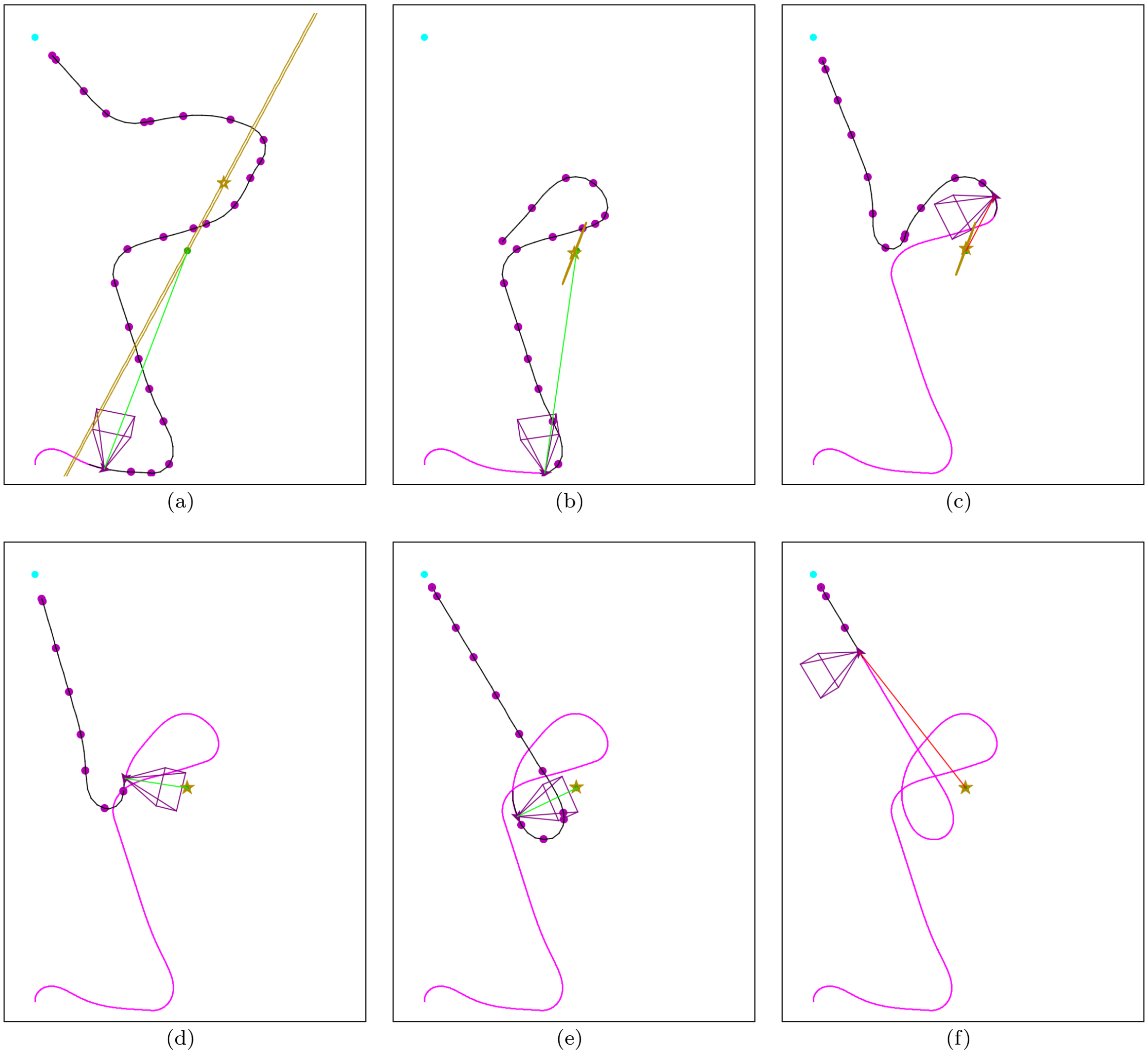

(a)
(b)
(c)
(d)
(e)
(f)

Figure 2: Snapshots of a simple Dubins car simulation with side mounted camera navigating toward a goal while tracking one aerial target. The magenta pyramid denotes a truncation of the camera field of view. The RRT tree has been suppressed for clarity. The vehicle (purple chevron, oriented with vehicle heading) starts at bottom-left and attempts to gather information about the target (green circle, center) and reach the goal waypoint (cyan, top-left). Relative uncertainty in the target location is represented with a 1-σ uncertainty ellipse (gold), with the current estimate denoted by a gold star. The vehicle's line-of-sight to the target is denoted with either a green or red line, the former denoting positive visibility, the latter a loss thereof. The vehicle's currently selected path is emphasized in black, where the magenta dots correspond to IRRT nodes/CL-RRT waypoints. A magenta "trail" traces the vehicle's actual position as it maneuvers through the environment.

6.2 Complex Dubins Scenario

Consider now the full problem statement as outlined in Section 3 for the Dubins car, extending the previous example. A Dubins car agent travels along a planar subspace of $X \subset \mathbb{R}^3$ with obstacles in $X_{\text{obs}} \subset X$ while estimating the location of *multiple* targets $\mathbf{y}_i \in Y \subset \mathbb{R}^3$, with the altitude of all $\mathbf{y}_i$ significantly above relevant features in X (hence, $X \cap Y = \emptyset$). Its monocular sensor is mounted on the driver's left side, pitched up by 60° as before, and has horizontal and vertical fields of view of 60° each. As in Section 6.1, the 20 cm car begins at $\mathbf{x}^{[q]}(t_0) = [2.5, -3.5, 1]^T$ m with an unobstructed path to the goal at $X_{\text{goal}} = \{[-2.5, 3.5, 1]^T \text{ m}\}$.

The presence of a cluttered obstacle environment presents several challenges over the previous example for the planning algorithm. First, the vehicle must be able to maintain feasibility by avoiding these obstacles; this is itself a challenging problem, since the vehicle moves at a fixed speed and thus cannot safely come to a stop, and may actually drive itself into a state where a collision is inevitable. Second, obstacles in the environment can occlude observations between sensors and targets, greatly complicating the representation of the sets of vehicle states for which the target(s) are observable. Whereas most heuristic approaches would have to adjust the path in an *ad hoc* manner to achieve both feasibility and visibility, these characteristics are embedded naturally in the IRRT algorithm.

A representative trial of the scenario is depicted in Figure 3; here the RRT trees have been left visible to demonstrate how the set of feasible paths evolves over time. Due to anticipation of occlusion between the sensor and targets, the planner selects paths that result in long periods of visibility. The agent initially plans to move toward the goal and then loiter in its vicinity, occasionally making distant measurements of the targets (Figure 3(a)). As the agent approaches the goal, the algorithm identifies a path in the tree that entails a more informative observation sequence while still avoiding obstacles (Figure 3(b)). As the target locations are made more precise, subsequent snapshots show the agent carefully moving through the obstacle field, attempting to take measurements at closer range while ensuring a safe return trajectory to the goal is available (Figures 3(c)-3(e)). When the vehicle has gathered enough information with respect to its cost function, it expeditiously plans a path to the goal through a tight corridor (Figures 3(e) and 3(f)).

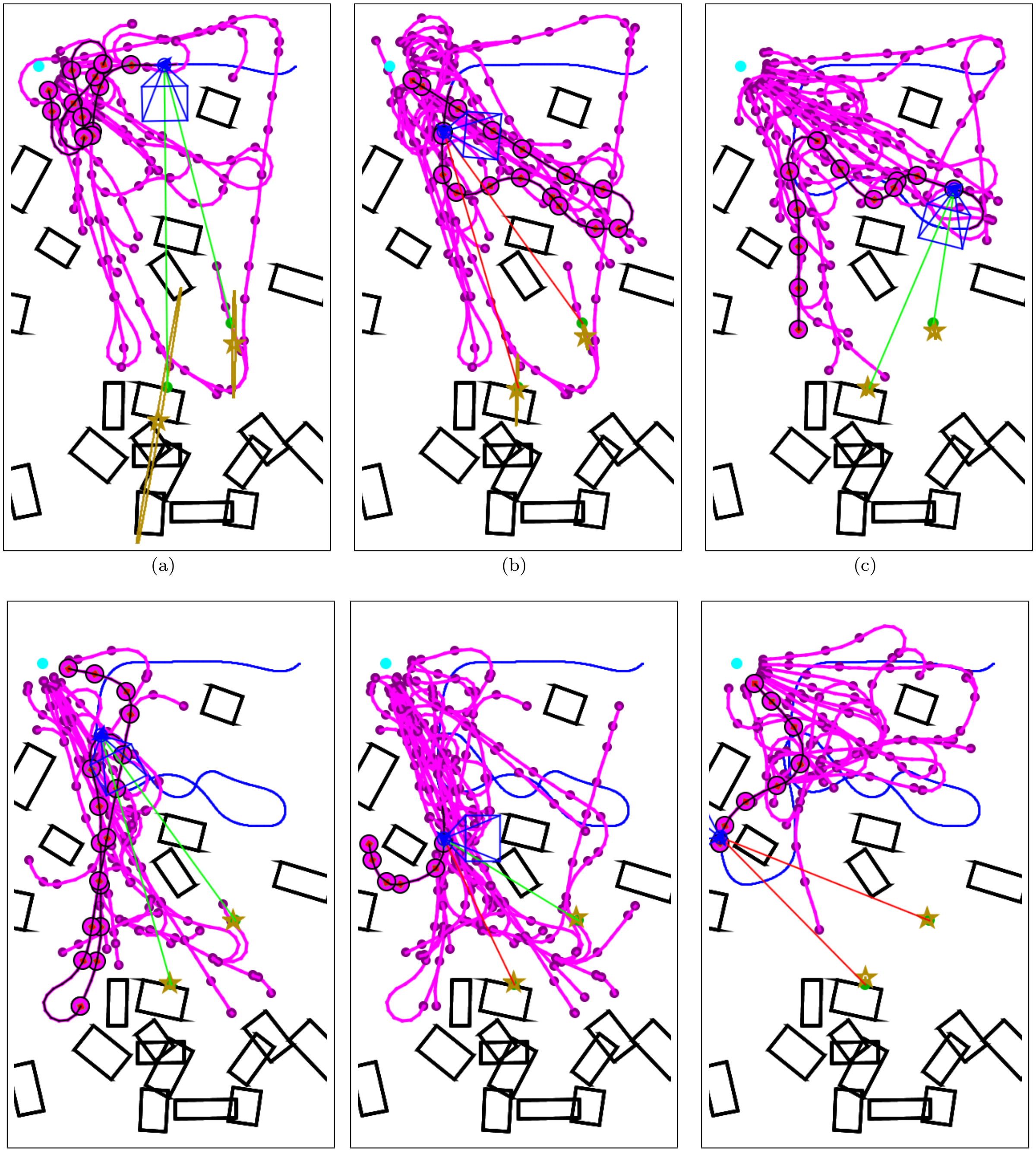
6.3 Analysis

Before proceeding to more complex examples, it is instructive to analyze how effective the IRRT algorithm is in gathering information along prospective trajectories, and how that capacity is weighed against the objective to reach the goal. In this section, we revisit the complex Dubins scenario considered in Section 6.2, with particular focus on reduction in target uncertainty over time.

Figure 4 plots the value of the information A-optimality cost, (12), for the complex Dubins scenario trial shown in Figure 3 as a function of time. The colored bars at the bottom of the figure correspond to the time intervals during which each target is visible for measurement by the agent. It is apparent that reduction in the A-optimality occurs when the targets are visible, with the slope of the curve depending on which targets are visible. As target 2 is slightly more visible in the opening phase of the mission, there is a diminishing return associated with taking data on this target later in the mission, as compared with that of target 1.

Another important consideration is the effect of varying β, a user-specified parameter governing the tradeoff between uncertainty reduction and path traversal cost. To evaluate its impact, we performed multiple simulations of the complex Dubins scenario for different values of β, recording the final A-optimality and path duration at the conclusion of each simulation. Seven values of $\beta \in \{10^{-1}, 10^0, 10^1, \ldots, 10^5\}$ were considered. Note that as $\beta \to 0$, the standard, information-naïve RRT algorithm is recovered. For each value of β, 25 trials were performed, consisting of 5 trials each on the same 5 instances of the complex Dubins scenario, with each instance having a distinct, randomized (feasible) obstacle arrangement and initial target estimate.

Figure 5 shows the resulting relationship between average mission duration and average terminal A-optimality as a function of β, which increases from $\log \beta = -1$ at bottom-right to $\log \beta = 5$ at top-left. As expected, as β increases the final A-optimality decreases, at the expense of a longer final path. For the lowest values of β, the algorithm essentially behaves as standard RRT, ignoring the target in pursuit of the goal. As β increases, the A-optimality value becomes relatively more important when selecting paths, and the algorithm will opt to select longer paths that entail longer observation sequences about the target.



(a) (b) (c)

(d) (e) (f)

Figure 3: Snapshots of a complex simulation of a Dubins car with side mounted camera navigating toward a goal while tracking two aerial targets. See Figure 2 for a full legend. Three-dimensional obstacles (black) in the environment are drawn as projected into two dimensions. The RRT tree (magenta edges, with darker nodes) is left visible in all figures to demonstrate how the tree evolves over time; for clarity, the vehicle, the sensor FOV pyramid, and the trail are represented here in blue.

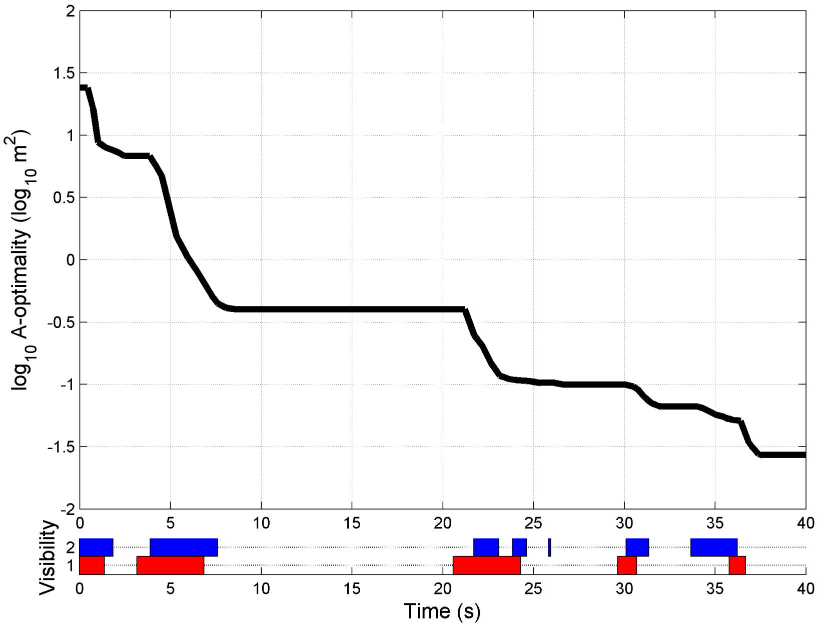


Figure 4: Comparison of the information A-optimality cost versus time for the complex Dubins scenario as shown in Figure 3. The colored bars at the bottom of the figure correspond to the time intervals during which each target is visible for measurement by the agent.

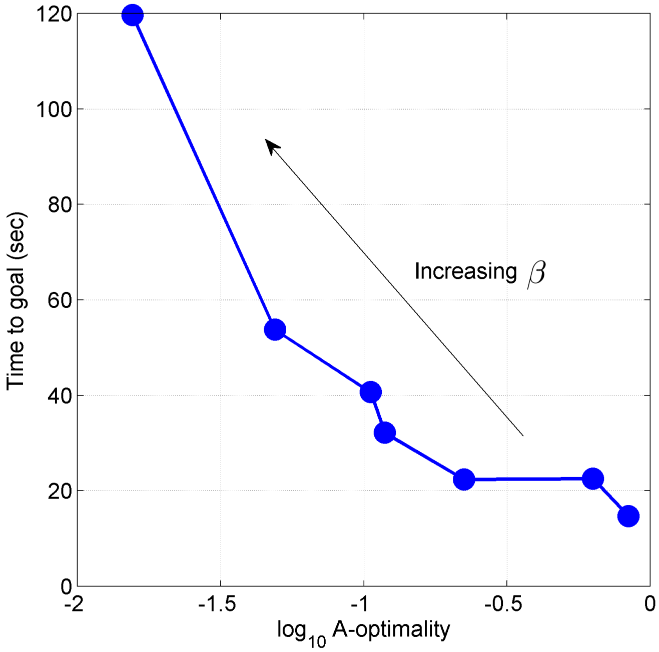


Figure 5: Comparison of average mission duration versus average terminal information A-optimality cost. Data points are parameterized by the relative information cost weighting term from (13) with values $\beta \in \{10^{-1}, 10^0, 10^1, \ldots, 10^5\}$. Each data point corresponds to one value of β, with $\log \beta = -1$ at bottom-right and $\log \beta = 5$ at top-left.

6.4 Three-Dimensional Scenario

The IRRT formulation can be applied in any number of dimensions; the following scenario demonstrates the capability of IRRT to design information-rich paths for a vehicle operating in a realistic, fully three-dimensional environment. Consider a quadrotor UAV agent navigating through an obstacle environment to track a stationary aerial target; the agent is free to change both its altitude and heading. In this sense, the RRT is actually sampling in four dimensions, three for position and one for heading. The agent's monocular sensor is mounted on the front of the vehicle, aimed forward and parallel to the ground, so it may be advantageous for the agent to change its heading to gain a vantage point for the target.

In this scenario, the agent begins on one end of a hallway at $\mathbf{x}^{[q]}(t_0) = [0.75, 5.25, 3.00]^T$ m, with an unobstructed path to the goal at $X_{\text{goal}} = \{[5.25, 5.25, 1.00]^T \text{ m}\}$. However, the agent also seeks to gather information on a target located at $\mathbf{y} = [2, 1, 2]^T$ m, which is located in a room off the hallway and behind a cluttered region of obstacles.

An example trial of the scenario is depicted in Figure 6. The agent begins with a path directly to the goal (Figure 6(a)), but the planner then identifies a path which gives the agent sufficient time to rotate and peer into the doorway (Figure 6(b)); upon doing so, the agent views the target. Now possessing more accurate knowledge of the target, the planner decides to send the agent into the room and underneath the obstacles (Figure 6(c)) to get a much closer view of the target behind them (Figure 6(d)). The planner then seeks to return the agent to the goal, and after some wandering succeeds in doing so (Figures 6(e) and 6(f)).

6.5 Multiagent Scenarios

6.5.1 Single Target Scenario

Consider a team of two Dubins agents collectively tasked with localizing an aerial target. Each agent generates and announces its own motion plan using IRRT, while broadcasting it to the other agent (Algorithm 2, line 11). It is assumed that the agents perform belief consensus on the target estimate after measurements have been realized and processed locally. Each agent is equipped with a monocular sensor, mounted on the vehicle's front and pitched up 25° from the plane, with 50° horizontal and vertical fields of view. A target is placed at $\mathbf{y} = [0, 1, 2]^T$ m. The mission consists of planning paths for the two agents with starting positions

$$\mathbf{x}_0^{[1]} = \begin{bmatrix}0, & -3.8, & 1.0\end{bmatrix}^T \text{ m} \qquad \mathbf{x}_0^{[2]} = \begin{bmatrix}0, & -3.0, & 1.0\end{bmatrix}^T \text{ m},$$

and goal positions

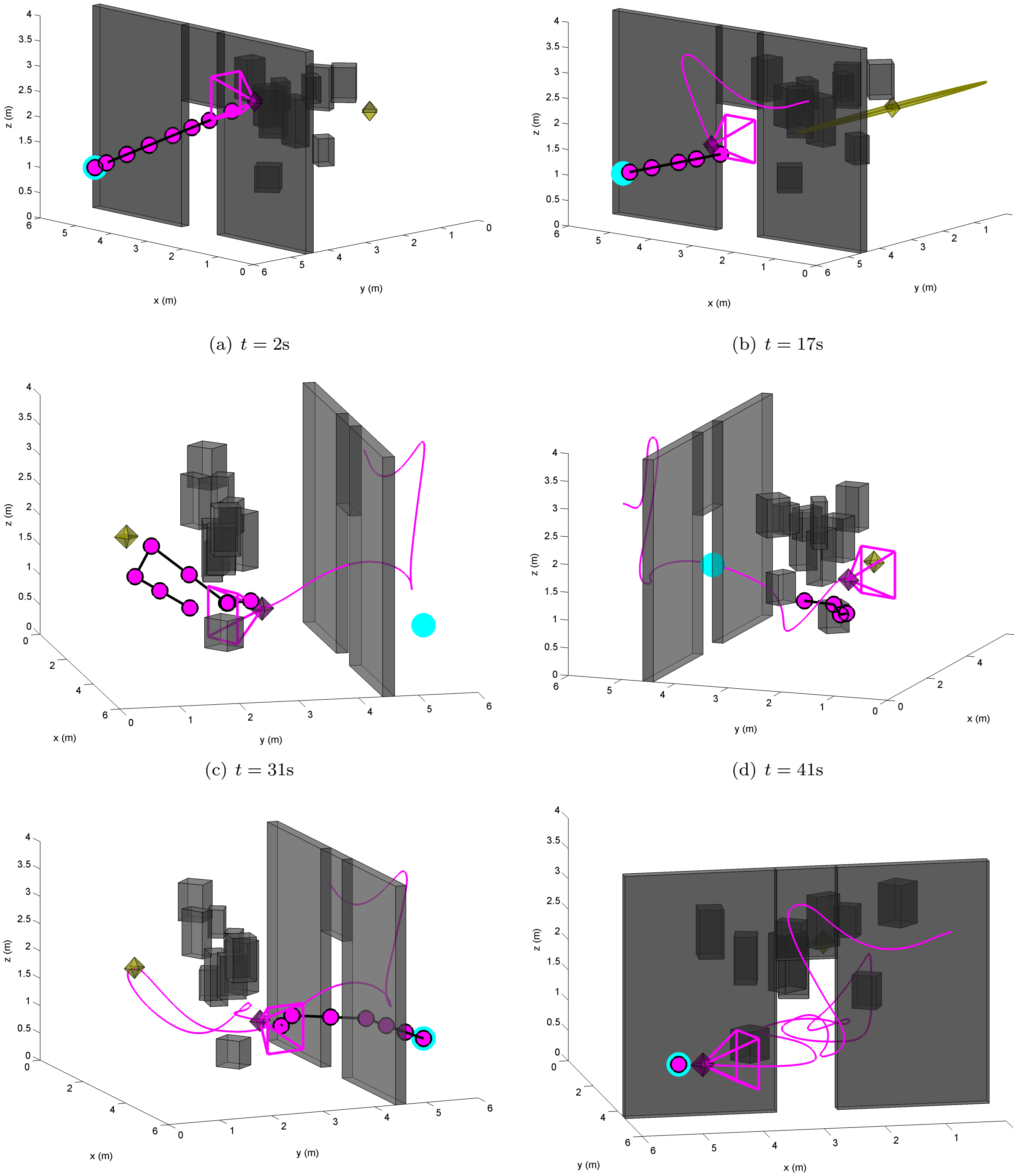
$$\mathbf{x}_g^{[1]} = \begin{bmatrix}0, & 3.0, & 1.0\end{bmatrix}^T \text{ m} \qquad \mathbf{x}_g^{[2]} = \begin{bmatrix}0, & 3.8, & 1.0\end{bmatrix}^T \text{ m},$$

that minimize the individual agent cost functions, with $\beta^{[1]} = \beta^{[2]} = 1900 \text{ s/m}^2$.

A representative trial for a such a scenario is depicted in Figure 7. Initially, the target position estimate is close to the true value, but the highly eccentric uncertainty ellipse is directed along the line-of-sight from both vehicles (Figure 7(a)). Based on the evolving target estimate, the vehicles individually plan paths that increase the difference in bearing between the two measurement sets, subject to the other agent's announced plan. Specifically, the path selected (Figures 7(b)-7(d)) balances deviation from the center-line (which forms the minimal-time path for each agent) with time spent triangulating the target. As the joint maneuver is sufficiently information-rich, when the target leaves the line of sight of both vehicles (Figure 7(e)), the remaining path segments connecting each agent to the goal are followed (Figure 7(f)).

6.5.2 Multitarget Scenario

A three-agent, eight-target scenario is now considered. Specifically, the performance of multiagent IRRT is compared for two planning modes. In both modes, belief consensus may be performed after measurements have been acquired and processed. The modal distinction arises from the treatment of planned information contributions of agents in the network. In the *noncooperative* mode, when an individual agent plans its path, the plan information contribution of all other agents in the network is ignored, effectively removing



(a) $t = 2$s

(b) $t = 17$s

(c) $t = 31$s

(d) $t = 41$s

(e) $t = 57$s

(f) $t = 70$s

Figure 6: Snapshots of a typical trajectory for a simulated quadrotor navigating in three dimensions toward a goal while localizing an aerial target. The vehicle (magenta diamond) attempts to gather information about the target (estimate and uncertainty in gold) and reach the goal waypoint (cyan) while avoiding obstacles. The agent's field of view is denoted by the magenta pyramid, while the vehicle's current reference path is denoted by magenta dots. The figure axes correspond to the environment boundaries.

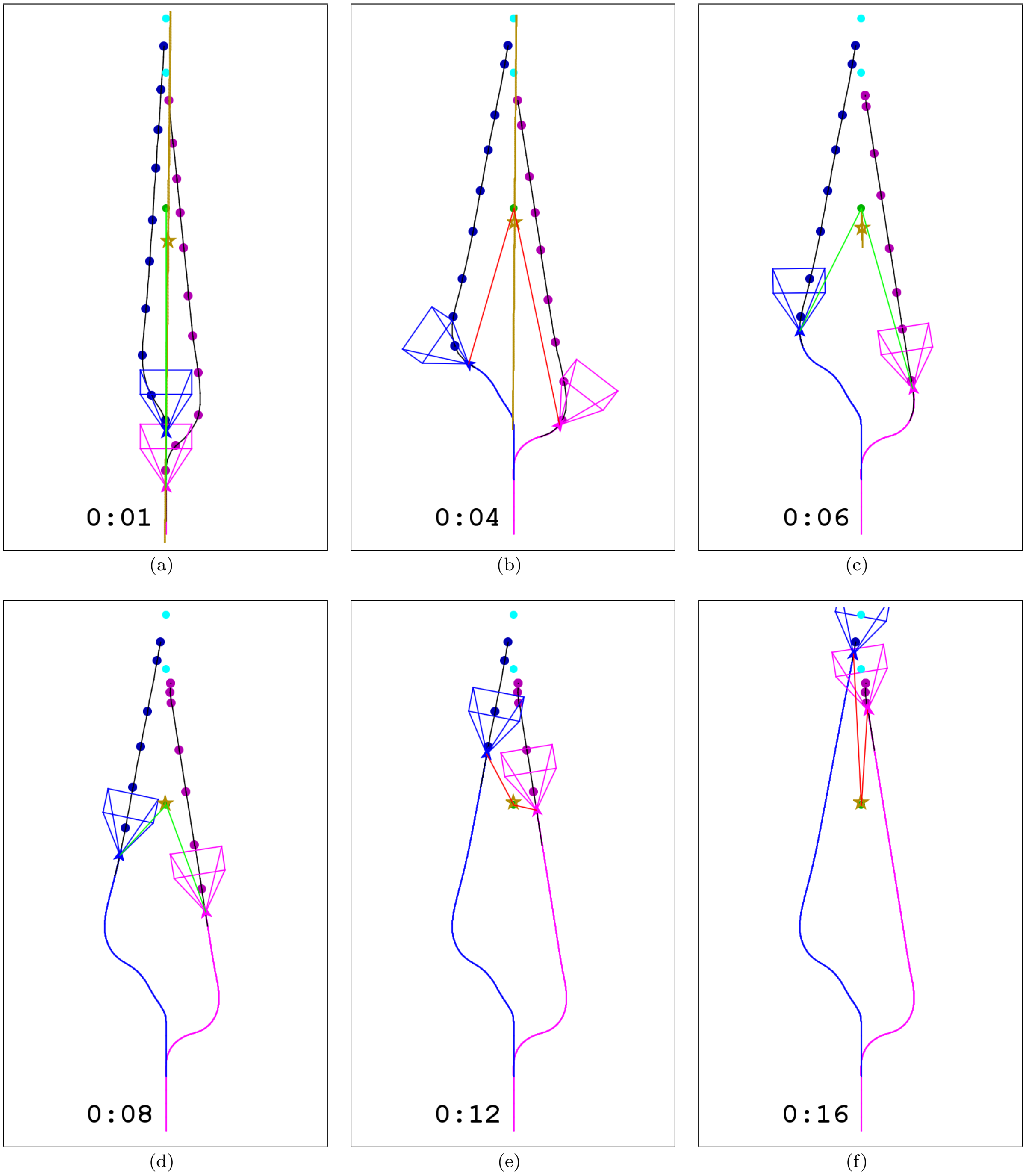


Figure 7: A multiagent Dubins scenario with sensor constraints. Each agent plans paths that minimize the goal arrival time and maximize the shared information. The ability for each agent to simultaneously take measurements from disparate measurement poses, as in (c) and (d), is considered favorable by the cost function of each agent's path planning module. (Timestamps are marked on each image, in seconds.)

Table 1: Multiagent scenario: true target positions.

	components in $\mathbb{R}^3$ [m]			
Target	x_i	y_i	z_i	
$\mathbf{y}_1$	[10.0,	5.0,	3.0	$]^T$
$\mathbf{y}_2$	[5.0,	15.0,	3.0	$]^T$
$\mathbf{y}_3$	[17.0,	9.0,	3.0	$]^T$
$\mathbf{y}_4$	[10.0,	10.0,	3.0	$]^T$
$\mathbf{y}_5$	[13.0,	18.1,	3.0	$]^T$
$\mathbf{y}_6$	[5.0,	8.0,	3.0	$]^T$
$\mathbf{y}_7$	[13.6,	13.6,	3.0	$]^T$
$\mathbf{y}_8$	[2.0,	18.0,	3.0	$]^T$

Table 2: Multiagent scenario: agent initial and goal states.

Agent	$\theta_0^{[q]}$ [rad]	$x_0^{[q]}$ [m]	$y_0^{[q]}$ [m]	$z_0^{[q]}$ [m]	$x_g^{[q]}$ [m]	$y_g^{[q]}$ [m]	$z_g^{[q]}$ [m]
1	0.0	1.5	1.0	1.0	15.0	10.0	1.0
2	3.1	19.0	17.5	1.0	5.0	10.0	1.0
3	0.0	1.5	12.0	1.0	11.7	19.0	1.0

the second term of (10). Alternatively, in the *cooperative* mode, an individual agent fully utilizes, as in the algorithm developed in Section 5, the plan information contribution of the other agents in the network.

The scenario environment, which is identical for all trials, consists of an axis-aligned box in the first octant of $\mathbb{R}^3$ with dimensions $[20,\ 20,\ 6]^T$ m. The cube is populated by 20 randomly generated box-shaped obstacles whose centroid placements are uniformly sampled within the environment, and whose lengths and widths are uniformly selected from the intervals $[0.5, 1.0]$ m and $[0.25, 0.5]$ m, respectively.

The true positions of the eight targets are given in Table 1. The initial estimate for each target i is random for each trial and is generated by perturbing the true positions according to

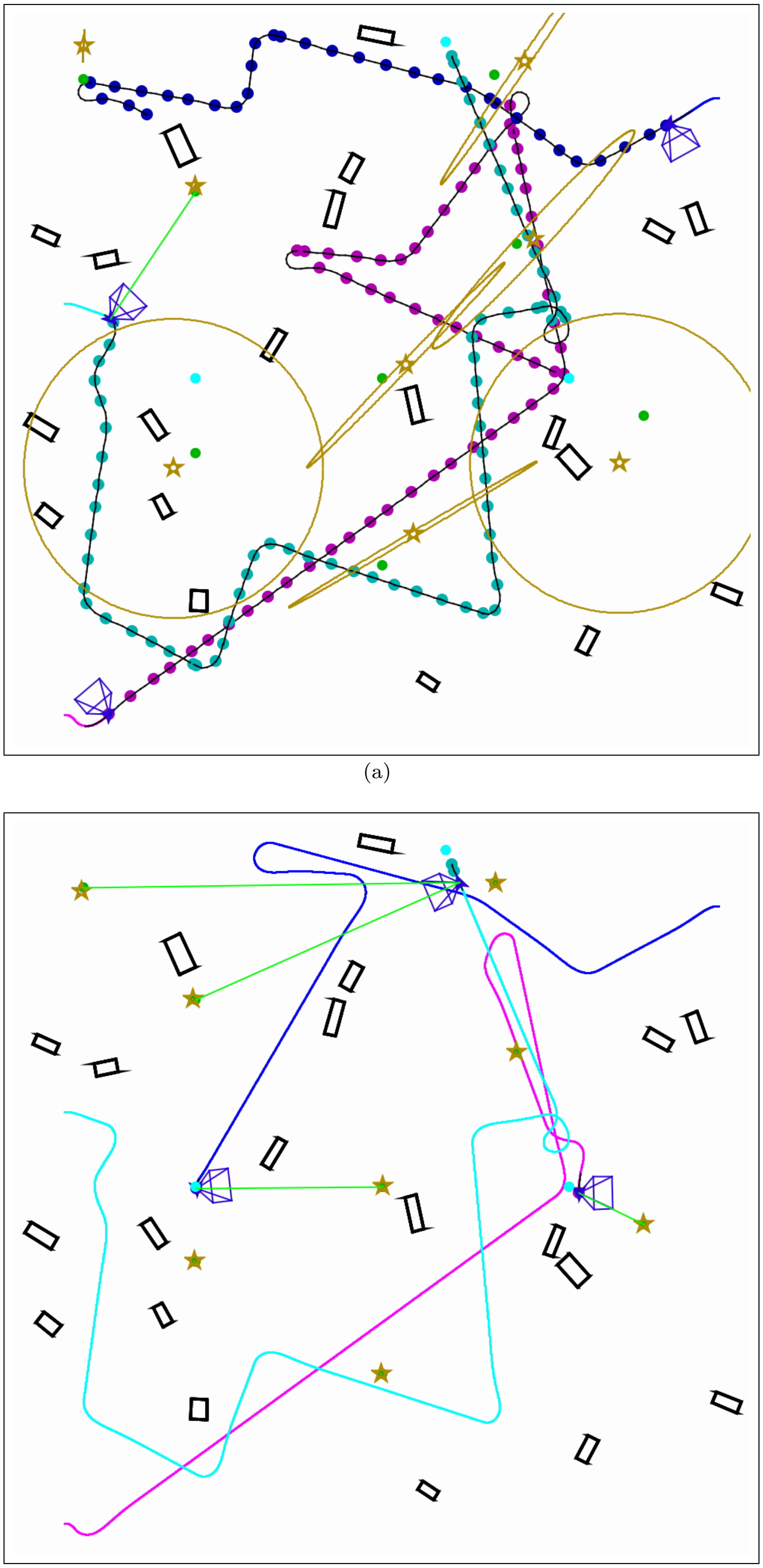
$$\hat{\mathbf{y}}_i(t_0) = \mathbf{y}_i + \mathbf{d}_i, \quad \mathbf{d}_i \sim \mathcal{N}\left(\mathbf{0}, \zeta_d^2 I_3\right), \quad \zeta_d = 0.5 \text{ m}. \tag{17}$$

All target covariances are initialized as $\Lambda_i(t_0) = \zeta^2 I_3, \zeta = 2.0$ m.

Each agent is a Dubins car with a diameter of 0.8 m and a monocular (bearings-only) sensor. The components of the initial and goal states for each agent are specified in Table 6.5.2. The sensor, which operates at 15 Hz and has 60° vertical and horizontal fields of view, is yawed 90° counterclockwise from the front of each vehicle and pitched up 30° from the plane. Simulated bearing measurements are corrupted by a zero-mean, additive white Gaussian noise sequence with a standard deviation of 5°. The parameters of the cost function (13) are set to $\alpha = 0.5$ and $\beta = 3000$ s/m^2 for all agents.

The qualitative behavior of the noncooperative and cooperative modes is illustrated in Figures 8 and 9, respectively, for example trials. Typically, agents in the noncooperative mode commit to path plans that are significantly longer than those selected by agents in the cooperative mode. While measurements taken by all agents reduce the uncertainty (hence, information cost) apparent to a particular agent, the inability to anticipate the plan information contribution of other agents leads each agent to selecting a (possibly circuitous) path for the purposes of collecting (possibly redundant) information from all targets. As seen in Figure 8, many of the agents' paths end up overlapping over the same regions, a behavior not observed in Figure 9.

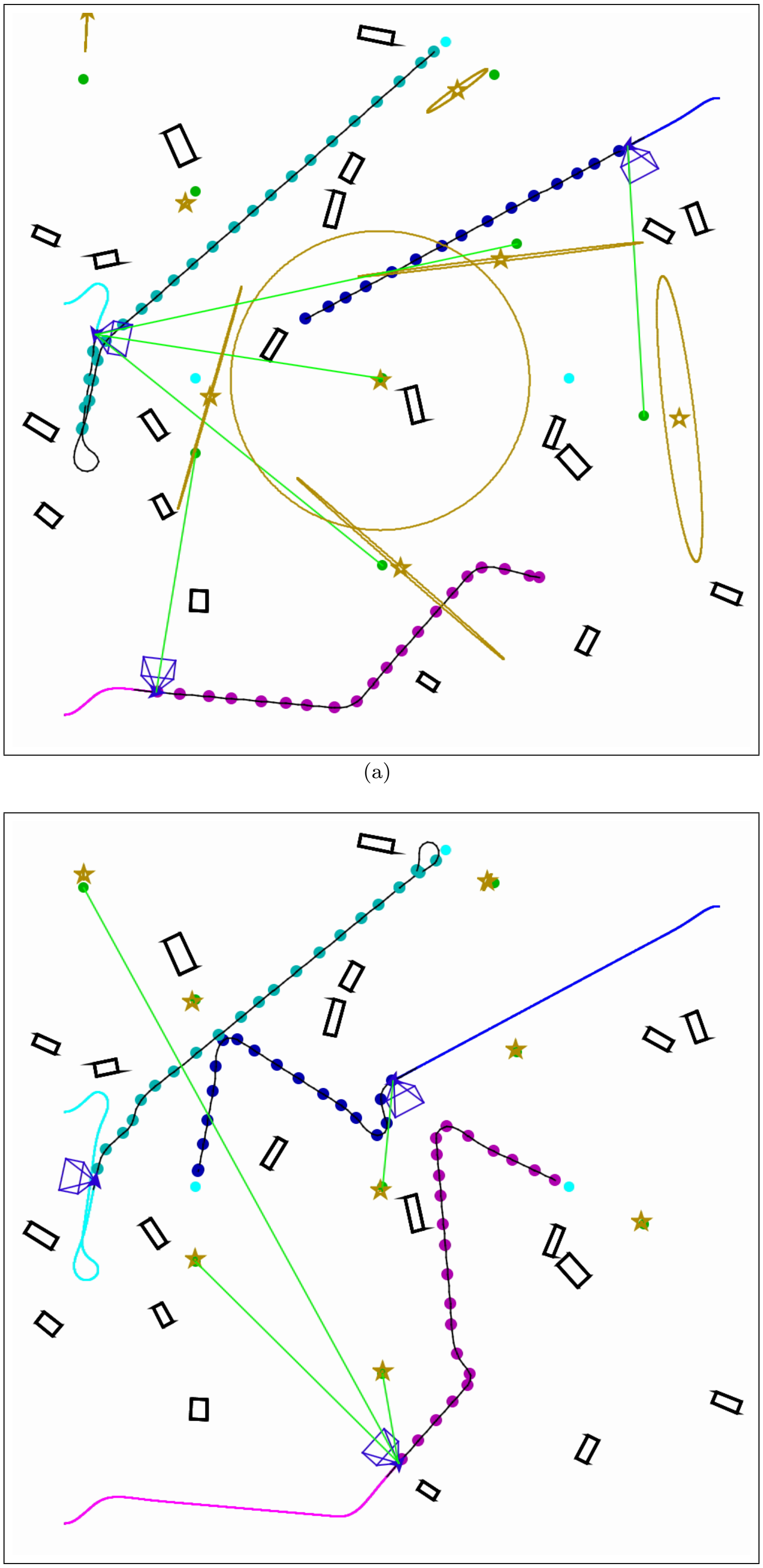
Recall that in the IRRT algorithm, the relative weighting between the information collection and path duration is β. Thus, in order to assess the mission performance, a mission-level cost $C = \Delta t + \beta \mathcal{I}_{term}$ is specified, where Δt is the mission duration and $\mathcal{I}$ is the terminal A-optimality cost. A network of agents that plan in the noncooperative mode typically gathers more information over the course of a mission, but does so



(a)

(b)

Figure 8: Sample simulation of the noncooperative multiagent IRRT scenario. (a) Each agent, ignoring the plan information content of the other agents, plans a circuitous path through the environment to collect information from all targets. (b) The resulting state history of the three agents at mission termination.



(a)

(b)

Figure 9: Sample simulation of the cooperative multiagent IRRT scenario. (a) Each agent, accounting for the plan information content of the other agents, plans an efficient, information-rich path through the environment to cooperatively collect information from all targets. (b) The resulting state history of the three agents at mission termination.

Table 3: Tabulated results of the batch multiagent IRRT comparison of the cooperative and noncooperative planning modes. The total cost is computed as $c = \Delta t + \beta \mathcal{I}_{term}$, where Δt is the mission duration, $\mathcal{I}_{term}$ is the terminal A-optimality cost, and $\beta = 3000$ s/m^2.

Metric		Cooperative	noncooperative
Mission Duration [s]	median	74.29	97.11
	mean	75.63	98.00
	std	14.35	15.93
	IQR	[63.96, 85.63]	[87.6, 106.8]
Terminal Information Cost [10^{-3} m^2]	median	5.02	2.99
	mean	5.20	3.15
	std	1.50	0.79
	IQR	[4.11, 5.92]	[2.59, 3.62]
Total Mission Cost [s]	median	88.36	105.07
	mean	91.23	107.46
	std	12.92	14.56
	IQR	[80.77, 99.86]	[97.69, 115.90]

at the expense of significantly longer mission durations. One would, therefore, expect the resultant mission-level cost to be higher in the noncooperative mode. To better quantify this statement for multiagent IRRT, a randomized algorithm, the performance of the noncooperative and cooperative modes are compared over a set of 100 trials of each. The statistical results can be found in Table 3 and in Figure 10. As expected, the cooperative mode generally outperforms the noncooperative mode, where the severity of underperformance in the latter is a function the scenario and mission parameters, particularly the information cost weight β in (13).

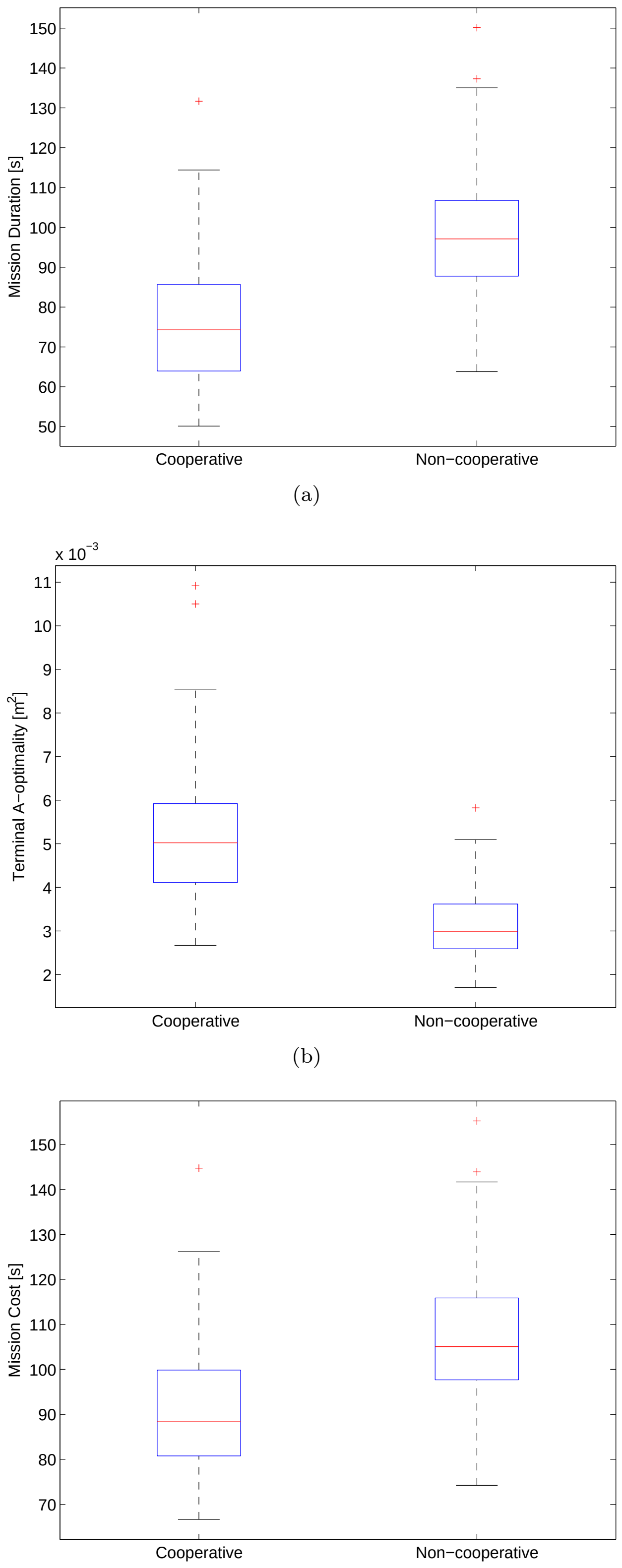
7 Conclusion

This paper has presented the Information-rich Rapidly-exploring Random Tree (IRRT) algorithm as a solution to the constrained informative motion planning problem that embeds metrics on uncertainty reduction at both the tree growth and path selection levels. IRRT possesses a number of beneficial properties, chief among them being the ability to find dynamically feasible, informative paths on short timescales, even subject to the aforementioned constraints. The utility of IRRT in efficiently localizing stationary targets was demonstrated in a progression of increasingly complex simulations, involving restrictive environments, multiple vehicles, and/or multiple targets. These results show that IRRT can be used in real-time to adaptively generate and execute information-rich paths in otherwise prohibitively constrained settings.

Two areas of possible future work could be pursued. In the multiagent configuration, there is the potential for "information loss" due to an agent publishing a plan and then reneging. Anticipating information loss is a potentially difficult but necessary step towards achieving robustness in decentralized informative motion planning. This work focused on parametric inference problems wherein the likelihood function is parametrized by some underlying vector. We note the inherent difficulty in quantifying the uncertainty reduction associated with observations in more general graph structures, for example, with hyperparameters over the parameters of the likelihood function.

Acknowledgments

This work was sponsored (in part) by the AFOSR and USAF under grant (FA9550-08-1-0086). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily



(a)

(b)

(c)

Figure 10: Box plot of the batch multiagent IRRT scenario. (a) The mission duration for the cooperative multiagent IRRT mode is significantly shorter, on average, than that of the noncooperative mode. (b) The extended duration of noncooperative missions generally results in more information collected, (c) though the overall cost may yet be adversely affected.

representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.

References

[1] S. Arulampalam, N. Gordon, and B. Ristic. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston, MA, 2004.

[2] D. B. Barber, J. Redding, T. Mclain, R. Beard, and C. Taylor. Vision-based target geo-location using a fixed-wing miniature air vehicle. *Journal of Intelligent and Robotics Systems*, 47(4):361–382, 2006.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, first edition, 2007.

[4] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte. Information based adaptive robotic exploration. In *IEEE International Conference on Intelligent Robots and Systems*, 2002.

[5] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New York, 1991.

[6] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: Part I. *Robotics & Automation Magazine, IEEE*, 13(2):99–110, 2006.

[7] H. J. S. Feder, J. J. Leonard, and C. M. Smith. Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research*, 18:650–668, 1999.

[8] R. A. Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical and Physical Sciences*, 222:309–368, 1922.

[9] C. S. R. Fraser, L. F. Bertuccelli, H.-L. Choi, and J. P. How. Hyperparameter consensus methods for agreement under non-Gaussian uncertainties. In *American Control Conference (ACC)*, pages 3192–3197, Baltimore, MD, July 2010.

[10] E. Frazzoli, M. A. Dahleh, and E. Feron. Real-time motion planning for agile autonomous vehicles. *AIAA Journal of Guidance, Control, and Dynamics*, 25(1):116–129, January-February 2002.

[11] E. W. Frew. *Trajectory Design for Target Motion Estimation Using Monocular Vision*. PhD thesis, Stanford University, 2003.

[12] R. He, S. Prentice, and N. Roy. Planning in information space for a quadrotor helicopter in a GPS-denied environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1814–1820, Los Angeles, CA, 2008.

[13] S. Huang, N. M. Kwok, G. Dissanayake, Q. Ha, and G. Fang. Multi-step look-ahead trajectory planning in SLAM: Possibility and necessity. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[14] L. Kavraki, P. Svestka, J. Latombe, and M. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996.

[15] C. Kreucher, K. Kastella, and A. Hero. Sensor management using an active sensing approach. *Signal Processing*, 85:607–624, March 2005.

[16] Y. Kuwata, J. Teo, S. Karaman, G. Fiore, E. Frazzoli, and J. P. How. Motion planning in complex environments using closed-loop prediction. In *AIAA Guidance, Navigation, and Control Conference (GNC)*, Honolulu, HI, Aug 2008. (AIAA-2008-7166).

[17] S. LaValle. *Planning Algorithms.* Cambridge University Press, 2006.

[18] S. M. LaValle. Rapidly-exploring random trees: A new tool for path planning. Technical Report 98-11, Iowa State University, October 1998.

[19] S. M. LaValle and J. J. Kuffner. Randomized kinodynamic planning. *International Journal of Robotics Research*, 20(5):378–400, May 2001.

[20] D. Levine, B. Luders, and J. P. How. Information-rich path planning with general constraints using rapidly-exploring random trees. In *AIAA Infotech@Aerospace Conference*, Atlanta, GA, April 2010. (AIAA-2010-3360).

[21] D. S. Levine. Information-rich path planning under general constraints using rapidly-exploring random trees. Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, Cambridge MA, June 2010.

[22] P. T. Liu. An optimum approach in target tracking with bearing measurements. *Journal of Optimization Theory and Applications*, 56(2):205–214, February 1988.

[23] B. Luders, S. Karaman, E. Frazzoli, and J. P. How. Bounds on track error using closed-loop rapidly-exploring random trees. In *American Control Conference (ACC)*, pages 5406–5412, Baltimore, MD, June/July 2010.

[24] A. Meliou, A. Krause, C. Guestrin, and J. Hellerstein. Nonmyopic informative path planning in spatio-temporal models. In *Proceedings of the National Conference on Artificial Intelligence (NCAI)*, 2007.

[25] J. Ousingsawat and M. E. Campbell. Optimal cooperative reconnaissance using multiple vehicles. *AIAA Journal of Guidance, Control, and Dynamics*, 30(1), 2007.

[26] J. M. Passerieux and D. Van Cappel. Optimal observer maneuver for bearings-only tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):777–788, Jul 1998.

[27] J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for POMDPs. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2003.

[28] S. S. Ponda. Trajectory optimization for target localization using small unmanned aerial vehicles. Master's thesis, Massachusetts Institute of Technology, 2008.

[29] S. S. Ponda, R. M. Kolacinski, and E. Frazzoli. Trajectory Optimization for Target Localization Using Small Unmanned Aerial Vehicles. In *AIAA Guidance, Navigation, and Control Conference (GNC)*, Chicago, IL, August 2009.

[30] F. Rafi, S. Khan, K. Shafiq, and M. Shah. Autonomous target following by unmanned aerial vehicles. In *Proceedings of SPIE*, Orlando, FL, May 2006.

[31] C. R. Rao. Information and the accuracy attainable in the estimation of statistical parameters. *Bulletin of the Calcutta Mathematical Society*, 37:81–89, 1945.

[32] B. Ristic, M. Morelande, and A. Gunatilaka. Information driven search for point sources of gamma radiation. *Signal Processing*, 90:1225–1239, April 2010.

[33] N. Roy and R. He. Efficient POMDP forward search by predicting the posterior belief distribution. Technical Report MIT-CSAIL-TR-2009-044, Massachusetts Institute of Technology, September 2009.

[34] R. Sim and N. Roy. Global a-optimality robot exploration in SLAM. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[35] A. Singh, A. Krause, C. Guestrin, W. Kaiser, and M. Batalin. Efficient planning of informative paths for multiple robots. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

[36] C. Stachniss and W. Burgard. Exploring unknown environments with mobile robots using coverage maps. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, 2003.

[37] S. Thrun. *Robotic mapping: A survey.* Morgan Kaufmann, 2002.

[38] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics.* MIT Press, Cambridge, MA, 2005.

[39] D. Uciński. *Optimal Measurement Methods for Distributed Parameter System Identification.* CRC Press, 2005.