

Safe Policy Synthesis in Multi-Agent POMDPs via Discrete-Time Barrier Functions

Mohamadreza Ahmadi, Andrew Singletary, Joel W. Burdick, and Aaron D. Ames

Abstract—A multi-agent partially observable Markov decision process (MPOMDP) is a modeling paradigm used for high-level planning of heterogeneous autonomous agents subject to uncertainty and partial observation. Despite their modeling efficiency, MPOMDPs have not received significant attention in safety-critical settings. In this paper, we use barrier functions to design policies for MPOMDPs that ensure safety. Notably, our method does not rely on discretizations of the belief space, or finite memory. To this end, we formulate sufficient and necessary conditions for the safety of a given set based on discrete-time barrier functions (DTBFs) and we demonstrate that our formulation also allows for Boolean compositions of DTBFs for representing more complicated safe sets. We show that the proposed method can be implemented online by a sequence of one-step greedy algorithms as a stand-alone safe controller or as a safety-filter given a nominal planning policy. We illustrate the efficiency of the proposed methodology based on DTBFs using a high-fidelity simulation of heterogeneous robots.

I. INTRODUCTION

Complex mission planning of multiple heterogeneous robots, e.g. flying and ground robots (see Figure 1), presents an inherent tension between the need for greater autonomy and the absolute necessity of strong safety [7] and performance guarantees [5]. Safety is crucial for the duration of a safety-critical mission, for example, those involving human-robot interactions [38]. The planning problem becomes even more involved in the presence of partial or uncertain information about the world, as well as stochastic actions and noisy sensors [33], [34].

Multi-agent partially observable Markov decision processes [25], [11] provide a sequential decision-making formalism for high-level planning of multiple autonomous agents under partial observation and uncertainty. In MPOMDPs, the agents share their local observations and make decisions based on a global information state (the joint belief). Despite this unique modeling paradigm, the computational complexity of MPOMDPs is PSPACE-complete [16], [22]. Therefore, several promising approximate methods for solving MPOMDPs have been proposed in the literature, e.g. sampling-based methods [11] and point-based methods [35]. However, it is difficult to provide safety assurances when one employs approximate methods for solving MPOMDPs, as such methods either use discretization techniques [21] or finite-state controllers [36].

M. Ahmadi, A. Singletary, J. W. Burdick, and A. D. Ames are with the California Institute of Technology, 1200 E. California Blvd., MC 104-44, Pasadena, CA 91125, e-mail: ({mrahmadi, asinglet, jwb, ames}@caltech.edu).

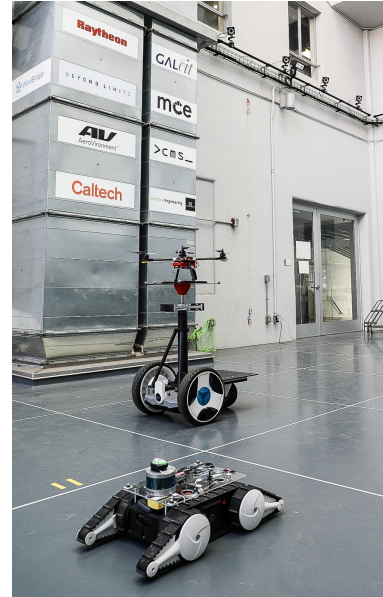


Fig. 1: A team of heterogeneous robots consisting of a quadrotor, a Segway, and a Flipper.

Safety verification can be encoded as checking whether the solutions of a system remain inside a pre-specified safe set or alternatively avoid a pre-defined unsafe set. Then, a natural method for checking safety is to compute the reachable set of a system subject to disturbances and controls [26], [1], [10]. However, for complex and high-dimensional systems such methods are either intractable, or overly conservative. Alternative approaches to reachability date back to the pioneering works of Nagumo [28] to study the set invariance of ordinary differential equations (ODEs). Nagumo's works were extended to ODEs with inputs by Aubin *et al.* [15] in the context of viability theory. The interest in hybrid systems in the 2000's led to the introduction of barrier certificates for safety verification [32]. However, the construction of these barrier certificates require solving a set of polynomial optimization problems that become intractable for high-dimensional systems (despite some promising recent directions [3]). The recently proposed notion of barrier functions [12] circumvent the computational bottleneck of barrier certificates inasmuch as the closed-form expression for a barrier function can be derived from the definition of the safe set. By taking advantage of this property, barrier functions have been used for designing safe controllers (in the absence of a nominal controller) and safety filters (in the presence of a nominal controller) for dynamical systems,

such as biped robots [29] and trucks [18], with guaranteed performance and robustness [39], [24].

In this paper, we extend the application of barrier functions from low-level safety constraints of dynamical systems to high-level safety objectives of MPOMDPs. Our results are based on the observation that the joint belief evolution of an MPOMDP is described by a discrete-time system [9], [4], [8]. We begin by formulating a, both necessary and sufficient, theorem for safety verification of a given set for discrete-time systems based on *discrete-time barrier functions* (DTBFs) and we demonstrate that our formulation allows for more complicated safe belief sets described by Boolean compositions of DTBFs. Then, we apply these DTBFs to study the safety of a given set of safe beliefs. We propose online methods based on one-step greedy algorithms to either synthesize a safe policy for an MPOMDP or synthesize a safety filter for an MPOMDP given a nominal planning policy. We illustrate the efficacy of the proposed approach by applying it to an exploration scenario of a team of heterogeneous robots in a high-fidelity simulation environment.

The rest of the paper is organized as follows. In the next section, we briefly review MPOMDPs and related notions. In Section III, we formulate a barrier function theorem for discrete-time systems. In Section IV, we use the tools developed in Section III to ensure safe planning in MPOMDPs. Section V elucidate our results by a high-fidelity multi-robot exploration simulation. Finally, in Section VI, we conclude the paper and give directions for future reserch.

Notation: \mathbb{R}^n denotes the n -dimensional Euclidean space. $\mathbb{R}_{\geq 0}$ denotes the set $[0, \infty)$. $\mathbb{N}_{\geq 0}$ denotes the set of non-negative integers. For a finite-set A , $|A|$ denotes the number of elements in A . A continuous function $f : [0, a) \rightarrow \mathbb{R}_{\geq 0}$ is a class \mathcal{K} function if $f(0) = 0$ and it is strictly increasing. Similarly, a continuous function $g : [0, a) \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is a class \mathcal{KL} function if $g(r, \cdot) \in \mathcal{K}$ and if $g(\cdot, s)$ is decreasing with respect to s and $\lim_{s \rightarrow \infty} g(\cdot, s) \rightarrow 0$. For two functions $f : \mathcal{G} \rightarrow \mathcal{F}$ and $g : \mathcal{X} \rightarrow \mathcal{G}$, $f \circ g : \mathcal{X} \rightarrow \mathcal{F}$ denotes the composition of f and g and $\text{Id} : \mathcal{F} \rightarrow \mathcal{F}$ denotes the identity function satisfying $\text{Id} \circ f = f$ for all functions $f : \mathcal{X} \rightarrow \mathcal{F}$.

II. MULTI-AGENT POMDPs

An MPOMDP [25], [11] provides a sequential decision-making formalism for high-level planning of multiple autonomous agents under partial observation and uncertainty. At every time step, the agents take actions and receive observations. These observations are shared via (noise and delay free) communication and the agents decide in a centralized framework.

Definition 1: An MPOMDP is a tuple

$$(I, Q, p^0, \{A_i\}_{i \in I}, T, R, \{Z_i\}_{i \in I}, O),$$

wherein

- I denotes a index set of agents;

- Q is a finite set of states with indices $\{1, 2, \dots, n\}$ (which can be described as the product space of the states of all agents);
- $p^0 : Q \rightarrow [0, 1]$ defines the distribution of the initial states, i.e., $p^0(q)$ denotes the probability of starting at $q \in Q$;
- A_i is a finite set of actions for agent i and $A = \times_{i \in I} A_i$ is the set of joint actions;
- $T : Q \times A \times Q \rightarrow [0, 1]$ is the transition probability, where $T(q, a, q') := P(q^t = q' | q^{t-1} = q, a^{t-1} = a)$, $\forall t \in \mathbb{Z}_{\geq 1}, q, q' \in Q, a \in A$, i.e., the probability of moving to state q' from q when the joint actions a are taken;
- $R : Q \times A \rightarrow \mathbb{R}$ is the immediate reward function for taking the joint action a at state q ;
- Z_i is the set of all possible observations for agent i and $Z = \times_{i \in I} Z_i$, representing outputs of discrete sensors. Often, $z \in Z$ are incomplete projections of the world states q , contaminated by sensor noise;
- $O : Q \times A \times Z \rightarrow [0, 1]$ is the observation probability (sensor model), where $O(q', a, z) := P(z^t = z | q^t = q', a^{t-1} = a)$, $\forall t \in \mathbb{Z}_{\geq 1}, q \in Q, a \in A, z \in Z$, i.e., the probability of seeing joint observations z given joint actions a were taken and resulting in state q' .

Since the states are not directly accessible in an MPOMDP, decision making requires the history of joint actions and joint observations. Therefore, we must define the notion of a joint *belief* or the posterior as sufficient statistics for the history [14]. Given an MPOMDP, the joint belief at $t = 0$ is defined as $b^0(q) = p^0(q)$ and $b^t(q)$ denotes the probability of the system being in state q at time t . At time $t + 1$, when joint action $a \in A$ is taken and joint observation $z \in Z$ is observed, the belief is updated via a Bayesian filter as

$$\begin{aligned} b^t(q') &= P(q' | z^t, a^{t-1}, b^{t-1}) \\ &= \frac{P(z^t | q', a^{t-1}, b^{t-1}) P(q' | a^{t-1}, b^{t-1})}{P(z^t | a^{t-1}, b^{t-1})} \\ &= \frac{P(z^t | q', a^{t-1}, b^{t-1})}{P(z^t | a^{t-1}, b^{t-1})} \\ &\quad \times \sum_{q \in Q} P(q' | a^{t-1}, b^{t-1}, q) P(q | a^{t-1}, b^{t-1}) \\ &= \frac{O(q', a^{t-1}, z^t) \sum_{q \in Q} T(q, a^{t-1}, q') b^{t-1}(q)}{\sum_{q' \in Q} O(q', a^{t-1}, z^t) \sum_{q \in Q} T(q, a^{t-1}, q') b^{t-1}(q)} \quad (1) \end{aligned}$$

where the beliefs belong to the belief unit simplex

$$\mathcal{B} = \left\{ b \in [0, 1]^{|Q|} \mid \sum_{q \in Q} b^t(q) = 1, \forall t \right\}.$$

A policy in an MPOMDP setting is then a mapping $\pi : \mathcal{B} \rightarrow A$, i.e., a mapping from the continuous joint beliefs space into the discrete and finite joint action space. The special case of I being a singleton (only one agent) is known as a partially observable Markov decision process (POMDP) [37].

The execution of an MPOMDP is carried out in the following steps [31]. At every time step t , each agent i observes z_i^t and communicates its own observation z_i^t to all other agents. The agent then in return receives observations of others $z \setminus \{z_i\}$ and uses the joint observations z^t and the previous joint action a^{t-1} to update the new joint belief b^t from (1). Finally, the agent looks up the joint action from the joint policy $\pi(b^t) = a^t$ and executes its individual action a_i^t .

Noting that the joint belief evolution of an MPOMDP (1) is described by a discrete-time system [9], [4], [8], in the next section, we propose conditions based on DTBFs for safety analysis of discrete-time systems.

III. BARRIER FUNCTIONS FOR DISCRETE-TIME SYSTEMS

While there is a long history of studying the set invariance properties of dynamical systems [28], recently these concepts were extended to include conditions over a set. This was done through the concepts of barrier functions [13]. In the same vein, in [2], the barrier function method was extended to discrete-time dynamical systems. Unfortunately, with the latter formulation of the (reciprocal) barrier functions, we can not study the solutions of the discrete-time system outside of the invariant set, i.e., if the solution is on the boundary of the set or when it leaves the set. To overcome this difficulty, we next extend the notion of (zeroing) barrier functions [13] to discrete-time systems.

A. Discrete-Time Barrier Functions

We consider the following discrete-time system

$$x^{t+1} = f(x^t), \quad t \in \mathbb{N}_{\geq 0}, \quad (2)$$

with $f : \mathcal{X} \rightarrow \mathcal{X} \subset \mathbb{R}^n$ and a safe set defined as

$$\mathcal{S} := \{x \in \mathcal{D} \mid h(x) \geq 0\}, \quad (3)$$

$$\text{Int}(\mathcal{S}) := \{x \in \mathcal{D} \mid h(x) > 0\}, \quad (4)$$

$$\partial\mathcal{S} := \{x \in \mathcal{D} \mid h(x) = 0\}. \quad (5)$$

We then have the following definition of a DTBF.

Definition 2 (Discrete-Time Barrier Function): For the discrete-time system (2), the continuous function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ is a discrete-time barrier function for the set \mathcal{S} as defined in (3)-(5), if there exists $\alpha \in \mathcal{K}$ satisfying $\alpha(r) < r$ for all $r > 0$ and a set \mathcal{D} with $\mathcal{S} \subseteq \mathcal{D} \subset \mathbb{R}^n$ such that

$$h(x^{t+1}) - h(x^t) \geq -\alpha(h(x^t)), \quad \forall x \in \mathcal{D}. \quad (6)$$

In fact, the discrete-time barrier function would more correctly be called a discrete-time zeroing barrier function per the literature [12], but we drop the “zeroing” as it is the only form of barrier function that will be considered throughout the rest of this paper.

We can show that the existence of a DTBF is both necessary and sufficient for invariance.

Theorem 1: Consider the discrete-time system (2). Let $\mathcal{S} \subseteq \mathcal{D} \subset \mathbb{R}^n$ with \mathcal{S} as described in (3)-(5). Then, \mathcal{S} is

invariant if and only if there exists a DTBF as defined in Definition 2.

Proof: Please refer to [6]. ■

B. Boolean Composition of Discrete-Time Barrier Functions

It is often desirable to consider sets defined by Boolean composition of multiple barrier functions. In this regard, in [19], the authors proposed non-smooth barrier functions as a means to analyze composition of barrier functions by Boolean logic, i.e., \vee (disjunction), \wedge (conjunction), and \neg (negation). Similarly, in this study, we use \max to represent \vee and \min to show \wedge . In other words, if $x \in \{x \in \mathbb{R}^n \mid \max_{i=1, \dots, k} h_i(x) \geq 0\}$, then there exists at least one $i_* \in \{1, \dots, k\}$ such that $h_{i_*}(x) \geq 0$ and if $x \in \{x \in \mathbb{R}^n \mid \min_{i=1, \dots, k} h_i(x) \geq 0\}$, then for all $i \in \{1, \dots, k\}$ we have $h_{i_*}(x) \geq 0$. The negation operator is trivial and can be shown by checking if $-h$ satisfies the invariance property.

In the following, we propose conditions for checking Boolean compositions of barrier functions. Fortunately, since we are concerned with discrete time systems, this does not require non-smooth analysis.

In the context of DTBFs, we have the following result.

Proposition 1: Let $\mathcal{S}_i = \{x \in \mathbb{R}^n \mid h_i(x) \geq 0\}$, $i = 1, \dots, k$ denote a family of safe sets with the boundaries and interior defined analogous to \mathcal{S} in (3). Consider the discrete-time system (2). If there exist a $\alpha \in \mathcal{K}$ satisfying $\alpha(r) < r$ for $\forall r > 0$ such that

$$\min_{i=1, \dots, k} h_i(x^{t+1}) - \min_{i=1, \dots, k} h_i(x^t) \geq -\alpha \left(\min_{i=1, \dots, k} h_i(x^t) \right) \quad (7)$$

then the set $\{x \in \mathbb{R}^n \mid \wedge_{i=1, \dots, k} h_i(x) \geq 0\}$ is forward invariant. Similarly, if there exist a $\alpha \in \mathcal{K}$ satisfying $\alpha(r) < r$ for all $r > 0$ such that

$$\max_{i=1, \dots, k} h_i(x^{t+1}) - \max_{i=1, \dots, k} h_i(x^t) \geq -\alpha \left(\max_{i=1, \dots, k} h_i(x^t) \right) \quad (8)$$

then the set $\{x \in \mathbb{R}^n \mid \vee_{i=1, \dots, k} h_i(x) \geq 0\}$ is forward invariant.

Proof: We prove the case for conjunction and the proof for disjunction is similar. If (7) holds from the proof of Theorem 1, we can infer that

$$\min_{i=1, \dots, k} h_i(x^t) \geq (\text{Id} - \alpha)^t \circ \left(\min_{i=1, \dots, k} h_i(x^0) \right).$$

That is, if $x^0 \in \{x \in \mathbb{R}^n \mid \min_{i=1, \dots, k} h_i(x) \geq 0\}$, then $\min_{i=1, \dots, k} h_i(x^t) \geq 0$ for all $t \in \mathbb{N}_{\geq 0}$, which in turn implies that $h_i(x) \geq 0$ for all $i \in \{1, \dots, k\}$. ■

The next section shows how the results in this section can be used to provide safety assurances for MPOMDPs.

IV. SAFETY-CRITICAL CONTROL OF MPOMDPs

Since the states are not directly observable in MPOMDPs, we are interested in guaranteeing safety in a probabilistic

setting in the joint belief space. To this end, we define the set of safe joint beliefs as

$$\mathcal{B}_s := \{b \in \mathcal{B} \mid h(b) \geq 0\}, \quad (9)$$

$$\text{Int}(\mathcal{B}_s) := \{b \in \mathcal{B} \mid h(b) > 0\}, \quad (10)$$

$$\partial\mathcal{B}_s := \{b \in \mathcal{B} \mid h(b) = 0\}, \quad (11)$$

where $h : \mathcal{B} \rightarrow \mathbb{R}$ is a given function. We denote by $\pi_n : \mathcal{B} \rightarrow \mathcal{A}$ a nominal joint policy mapping each joint belief into a joint action. We use subscript n to denote variables corresponding to the nominal policy designed offline.

We are interested in solving the following problems for MPOMDPs.

Problem 1: Given an MPOMDP as defined in Definition 1, a corresponding belief update (1), and a safe joint belief set \mathcal{B}_s , design a sequence of actions a^t , $t \in \mathbb{N}_{\geq 0}$ such that $b^t \in \mathcal{B}_s$, $\forall t \in \mathbb{N}$ and the instantaneous rewards $r^t = \sum_{q^t \in \mathcal{Q}} b(q^t)R(q^t, a^t)$ are maximized for all $t \in \mathbb{N}_{\geq 0}$.

Problem 2: Given an MPOMDP as defined in Definition 1, a corresponding belief update equation (1), a safe joint belief set \mathcal{B}_s , and a nominal planning policy π_n , determine a sequence of actions a^t , $t \in \mathbb{N}_{\geq 0}$ such that $b^t \in \mathcal{B}_s$, $\forall t \in \mathbb{N}_{\geq 0}$ and the quantity $\|r^t - r_n^t\|^2$ is minimized for all $t \in \mathbb{N}_{\geq 0}$, where r_n^t denotes the nominal immediate reward at time step t .

As can be inferred from Problems 1 and 2, we seek to ensure safety in addition to motion planning at every time step. Such problems are prevalent in multi-agent robot applications, where safety is of significant importance, e.g., robots in performing tasks in the presence of human coworkers [27].

A. Barrier Functions for MPOMDPs

Next, we use the result in Theorem 1 to ensure safety of a team of heterogeneous autonomous agents described by an MPOMDP. To this end, we solve the following discrete optimization problem at each time step t :

$$a^* = \arg \max_{a \in A} \left(\sum_{q' \in \mathcal{Q}} b(q')R(q', a) \right) \quad \text{s.t.} \quad h(b(q')) - h(b(q)) \geq -\alpha(b(q)). \quad (12)$$

Algorithm 1 summarizes the steps involved in finding the safe action based on barrier functions at each time step. At every time step, the algorithm picks a joint action $a(i)$ from $|A|$ combinations of actions (recall that $\times_{i \in I} A_i = A$). For each joint action $a(i)$, it computes the next joint belief and checks whether if the next joint belief satisfies the safety constraint. If the safety constraint is satisfied, it computes the corresponding reward function $r(i)$ for the joint action $a(i)$. After checking all actions, the algorithm returns the joint action maximizing the reward function.

Algorithm 1 designs a safe and myopic optimal action at each time step based on the current observation and the belief state at the step before. Therefore, it does not require a

Algorithm 1 The one-step greedy algorithm for finding the safe action at time t .

Require: System information $I, \mathcal{Q}, A, T, R, Z, O$, safe belief set \mathcal{B}_s , current observation z^t , the past belief b^{t-1}

```

i = 1
for i = 1, 2, ..., |A| do
  bt(q') =  $\frac{O(z^t|q', a(i)) \sum_{q \in \mathcal{Q}} T(q'|q, a(i)) b^{t-1}(q)}{\sum_{q' \in \mathcal{Q}} O(z^t|q', a(i)) \sum_{q \in \mathcal{Q}} T(q'|q, a(i)) b^{t-1}(q)}$ 
  if h(bt) - h(bt-1) ≥ -α(h(bt-1)) then
    r(i) =  $\left( \sum_{q' \in \mathcal{Q}} b(q')R(q', a(i)) \right)$ 
  end if
end for
i* = arg maxi=1,2,...,|A| r(i)
return a* = a(i*)

```

Algorithm 2 The one-step greedy algorithm for finding the safe action at time t when agents have different safety constraints.

Require: System information $I, \mathcal{Q}, A, T, R, Z, O$, safe belief set \mathcal{B}_s , current observation z^t , the past belief b^{t-1}

```

i = 1
for i = 1, 2, ..., |A| do
  bt(q') =  $\frac{O(z^t|q', a(i)) \sum_{q \in \mathcal{Q}} T(q'|q, a(i)) b^{t-1}(q)}{\sum_{q' \in \mathcal{Q}} O(z^t|q', a(i)) \sum_{q \in \mathcal{Q}} T(q'|q, a(i)) b^{t-1}(q)}$ 
  if hk(bkt(q')) - hk(bkt-1(q)) ≥ -αk(hk(bkt-1(q))) for all k ∈ I then
    r(i) =  $\left( \sum_{q' \in \mathcal{Q}} b(q')R(q', a(i)) \right)$ 
  end if
end for
i* = arg maxi=1,2,...,|A| r(i)
return a* = a(i*)

```

full memory of past actions and observations. This synthesis algorithm for POMDPs parallels those using control barrier functions for dynamical systems wherein safety for all time and optimality at each time instance is required [12].

Note that, if the safety requirement is defined by Boolean logic and we need to check either inequality (7) or (8), we can just replace the inequality in the “if” statement in Algorithm 1 with either inequality (7) or (8).

Furthermore, we remark that stability is not an issue in MPOMDP problems, since the beliefs evolve in the probabilistic belief simplex. However, we can encode instability in an MPOMDP problem as a set of bad states, that is, $\mathcal{B} \setminus \mathcal{B}_s$.

Each autonomous agent might have a different safety requirement, characterized by sets \mathcal{B}_i , $i \in I$, i.e., \mathcal{B}_i is the safe set for agent i . Then, we just need to check the safety of each agent separately. We denote by b_i , $i \in I$, the subset of joint beliefs concerning agent i , e.g. beliefs showing the location of the agent. Algorithm 2 demonstrates how we can check the safety requirement of each agent separately

Algorithm 3 The one-step greedy algorithm for filtering the nominal policy with a safe action at every time-step t .

Require: System information I, Q, A, T, R, Z, O , nominal policy π_n , safe belief set \mathcal{B}_s , current observation z^t , the past belief b^{t-1}

$$b^t(q') = \frac{O(z^t|q', a_n^t) \sum_{q \in Q} T(q'|q, a_n^t) b^{t-1}(q)}{\sum_{q' \in Q} O(z^t|q', a_n^t) \sum_{q \in Q} T(q'|q, a_n^t) b^{t-1}(q)}$$

if $h(b^t) - h(b^{t-1}) < -\alpha(h(b^{t-1}))$ **then**
 $i = 1$
for $i = 1, 2, \dots, |A|$ **do**

$$b^t(q') = \frac{O(z^t|q', a(i)) \sum_{q \in Q} T(q'|q, a(i)) b^{t-1}(q)}{\sum_{q' \in Q} O(z^t|q', a(i)) \sum_{q \in Q} T(q'|q, a(i)) b^{t-1}(q)}$$

if $h(b^t) - h(b^{t-1}) \geq -\alpha(h(b^{t-1}))$ **then**
 $r(i) = \left(\sum_{q' \in Q} b(q') R(q', a(i)) \right)$
end if
end for
 $i_* = \arg \min_{i=1,2,\dots,|A|} \|r(i) - r_n^t\|^2$
return $a^* = a(i_*)$
end if
return $a^* = a_n^t$.

at every time step.

In many real world multi-robot navigation scenarios, an offline policy for path planning exists (e.g. based on point-based methods [35]). However, such policy may not guarantee safety. We can use the barrier functions to design an online method for ensuring safety while remaining as much faithful as possible to the offline policy (see [17], [20] for analogous formulations for systems described by nonlinear differential equations).

Algorithm 3 illustrates how barrier functions can filter the agent actions to ensure safety. At every time step t , the algorithm first computes the next joint belief b^t given the nominal action a_n designed based on the nominal policy π_n . It then checks whether that action leads to a safe joint belief update (this is allowed since the existence of a DTBF h satisfying (6) is both necessary and sufficient for safety). If yes, the algorithm returns a_n for implementation. If no, the algorithm finds a safe joint action that minimally changes the immediate reward from the nominal immediate reward r_n^t in a least squares sense.

V. CASE STUDY: MULTI-ROBOT EXPLORATION

To demonstrate our method, we consider a system of three heterogeneous robots exploring an unknown environment. The mission objective is to retrieve a sample located somewhere in the robots' vicinity. Each robot has different and limited capabilities to explore and observe the environment, so coordination and communication between the robots is required in order to complete the mission.

The robot team consists of a drone and two ground vehicles. The drone can rapidly explore the environment from above, but is unable to explore any covered or underground regions. The ground vehicles include a Rover Robotics

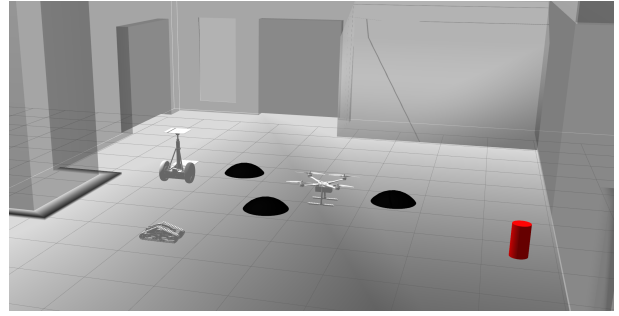


Fig. 2: The agents, the obstacles (black), and the sample (red) in ROS simulation environment.

Flipper, and a modified Segway. The Flipper is a small, tracked vehicle capable of traversing in tight spaces and rough terrain, while the Segway is a larger, wheeled robot without external sensing capabilities, whose purpose is to retrieve the sample.

The set of agents includes the UAV, A_U , the Flipper, A_F , and the Segway, A_S . These agents all inhabit a planar $n \times m$ grid, with the drone located two meters above the ground vehicles. The beliefs of the vehicle locations in this grid are updated after each action is completed, based on the previous belief and the observation made. The initial vehicle locations are known, and the remaining system states are given by the environmental model.

In order to capture the heterogeneity of the team, each grid in the environment has two states that measure the habitability of the grid for the Segway, as well as the probability that the grid in question contains the sample. These states are initialized to 0.5, and observations of these states can be made when the Flipper or the UAV are within a certain distance from the cell. The Flipper can make accurate observations about the traversability of the terrain, but cannot sense the location of the sample well. The drone is more suited to locating the sample, but less suited to gauging traversability.

The set of actions for each agent A_i is the same, and consists of five actions: remaining in the same grid, and moving either forwards, backwards, left, or right to an adjacent grid. Thus, the total number of actions is 125. The transitions between states are handled by controllers on the low-level dynamics, and the transition probability T when moving from one grid to another is modeled as a high chance to move to the desired grid, and equal, smaller chances of landing in one of the eight grids adjacent to the desired grid.

The components of the reward function for the Flipper and the UAV are measures of how much information will be gained from moving in that direction. The reward function also includes a heavy reward for the Segway moving towards a cell likely to contain the sample, and a heavy cost towards moving to a potentially dangerous cell. The observations for each agent update the environment states based on the observation made (binary detection) and the beliefs of the vehicles locations.

The exploratory mission is concluded when the sample has

been collected, resulting in a mission success. In terms of the system states, mission objective is satisfied when the Segway inhabits the same grid as the sample. If the Segway enters an uninhabitable region, this results in a mission failure. Thus, the safe set of beliefs is defined as all states in which the Segway does not coincide with an uninhabitable region. For this mission, given the partial observability constraint, we require that there is a 95% probability of the Segway entering a habitable grid with each action. It is important to note that safety for this problem does not depend on the entirety of the belief space. Thus, it is possible to verify safety without computing the beliefs of each of the states.

The simulation is carried out in a ROS environment as depicted in Figure 2. Occupancy grids are utilized to represent the states of the system, which are updated after each action is taken. When an action is initialized, the low-level dynamics of the vehicles are simulated, and a message is published when the action is complete. Utilizing the observations made by this action, the beliefs are updated, and a new joint action is generated. The simulation is ended when the true position of the robot inhabits the same grid as the true position of the sample, or when the robot coincides with an uninhabitable cell.

To demonstrate the efficacy of the policy filter, a near-optimal policy that violates the belief safety filter was passed through Algorithm 3. The trajectory of the Segway under this policy is shown in Figure 3. While the policy is successful in simulation, due to perfect control over the states, it does not meet the imposed requirements for probabilistic %95 safety.

The resulting trajectory of the Segway after the policy filter is shown in Figure 4. The first filtered action occurred at the time of the first image. While the desired trajectory of the Segway was to move towards the uninhabitable terrain, as shown in blue, the safety filter rejected this action. Instead, the action with the next highest reward, indicated by the orange arrow, was taken. This process continues, and the final trajectory is shown to move to the right wall of the building and then to the sample location. Thus, this filter was able to circumvent the unsafe policy, while still achieving the objective of the mission. While the resulting route is less optimal, it is a policy that could be implemented on a real system with realistic safety guarantees.

VI. CONCLUSIONS AND FUTURE WORK

We proposed a method for safe planning under uncertainty and partial observation of teams of heterogeneous robots modelled by MPOMDPs based on barrier functions. We applied our method for safe planning of a team of three robots using high-fidelity simulations.

We considered agents with perfect communication. Prospective work will consider MPOMDPs with communication delays [31] or with no communication (decentralized POMDPs) [30].

For Markov decision processes, safety can be encoded as a set of probabilistic temporal logic (PTL) specifications. In particular, in reinforcement learning finding optimal policies can induce unsafe behavior and shielded decision

making [23] has been introduced as an online safety filter to ensure safety in terms of PTL specifications. Future research will seek to present shielded decision making techniques for systems subject to uncertainty and partial observation using the DTBF-based safety filter developed in this paper.

Finally, in addition to high-fidelity simulations, we are implementing the results discussed in this paper in the Center for Autonomous Systems and Technologies (CAST) at the California Institute of Technology. Our eventual goal is to implement this work in the multi-agent planning framework for the DARPA Subterranean Challenge. Our experimental observations will be disseminated in a follow up paper.

REFERENCES

- [1] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.
- [2] A. Agrawal and K. Sreenath. Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation. In *Robotics: Science and Systems*, 2017.
- [3] Amir Ali Ahmadi and Anirudha Majumdar. Dsos and sdsos optimization: Lp and socp-based alternatives to sum of squares optimization. In *2014 48th annual conference on information sciences and systems (CISS)*, pages 1–5. IEEE, 2014.
- [4] M. Ahmadi, M. Cubuktepe, N. Jansen, and U. Topcu. Verification of uncertain pomdps using barrier certificates. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 115–122. IEEE, 2018.
- [5] M. Ahmadi, H. Mojalessi, and R. Wisniewski. Guaranteed cost Hinf controller synthesis for switched systems defined on semi-algebraic sets. *Nonlinear Analysis: Hybrid Systems*, 11:37–56, 2014.
- [6] M. Ahmadi, A. Singletary, J. W. Burdick, and A. D. Ames. Safe Policy Synthesis in Multi-Agent POMDPs via Discrete-Time Barrier Functions. *arXiv preprint arXiv:1903.07823*, 2019.
- [7] M. Ahmadi, G. Valmorbida, and A. Papachristodoulou. Safety verification for distributed parameter systems using barrier functionals. *Systems & Control Letters*, 108:33–39, 2017.
- [8] M. Ahmadi, B. Wu, Y. Chen, Y. Yue, and U. Topcu. Barrier certificates for assured machine teaching. *2019 American Control Conference*, 2019.
- [9] M. Ahmadi, B. Wu, H. Lin, and U. Topcu. Privacy verification in pomdps via barrier certificates. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 5610–5615. IEEE, 2018.
- [10] Matthias Althoff, Olaf Stursberg, and Martin Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *2008 47th IEEE Conference on Decision and Control*, pages 4042–4048. IEEE, 2008.
- [11] C. Amato and F. A. Oliehoek. Scalable planning and learning for multiagent pomdps. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [12] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017.
- [13] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control Barrier Function Based Quadratic Programs for Safety Critical Systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017.
- [14] K. J. Astrom. Optimal control of markov decision processes with incomplete state estimation. *Journal of mathematical analysis and applications*, 10:174–205, 1965.
- [15] J.-P. Aubin, A. M. Bayen, and P. Saint-Pierre. *Viability theory: new directions*. Springer Science & Business Media, 2011.
- [16] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of markov decision processes. *Mathematics of operations research*, 27(4):819–840, 2002.
- [17] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt. Control barrier certificates for safe swarm behavior. *IFAC-PapersOnLine*, 48(27):68–73, 2015.
- [18] Y. Chen, A. Hereid, H. Peng, and J. Grizzle. Enhancing the performance of a safe controller via supervised learning for truck lateral control. *arXiv preprint arXiv:1712.05506*, 2017.

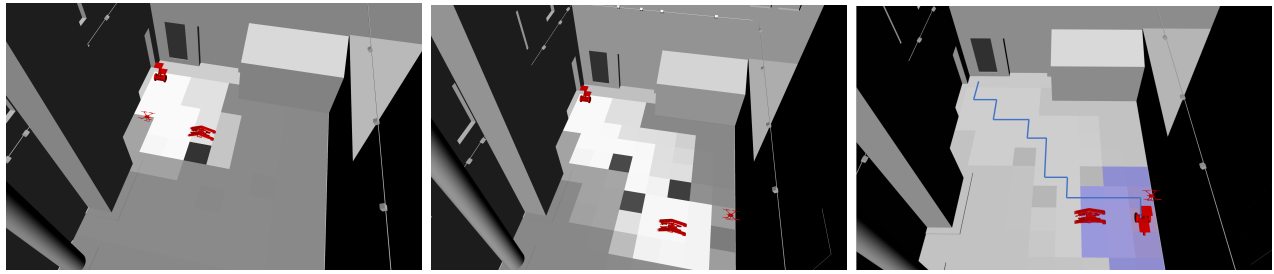


Fig. 3: Implementation of the nominal policy. Darker cells represent unsafe terrain, and the blue cells in the third image represents the belief of the Segway location.

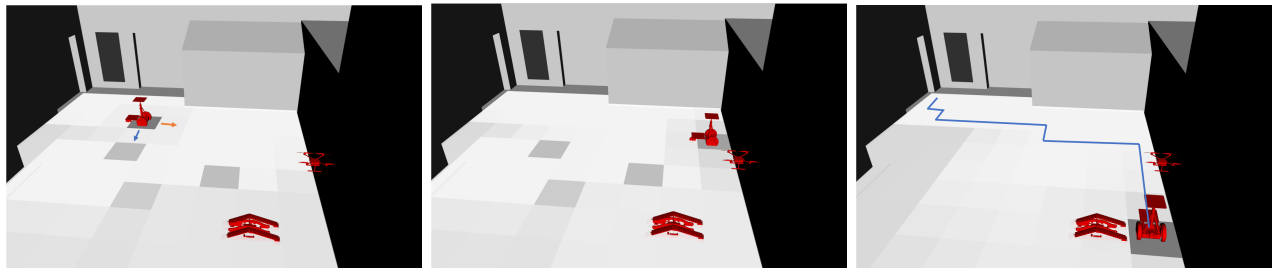


Fig. 4: Implementation of the safety filter. The blue arrow in the first image represents the desired action, and the orange arrow is the filtered action.

- [19] P. Glotfelter, J. Cortés, and M. Egerstedt. Nonsmooth barrier functions with applications to multi-robot systems. *IEEE control systems letters*, 1(2):310–315, 2017.
- [20] T. Gurriet, M. Mote, A. D. Ames, and E. Feron. An online approach to active set invariance. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 3592–3599. IEEE, 2018.
- [21] S. Haesaert, P. Nilsson, C. Vasile, R. Thakker, A. Agha-mohammadi, A. D. Ames, and R. M. Murray. Temporal logic control of pomdps via label-based stochastic simulation relations. *IFAC-PapersOnLine*, 51(16):271–276, 2018.
- [22] E. A. Hansen, D. S. Bernstein, and S. Zilberstein. Dynamic programming for partially observable stochastic games. In *AAAI*, volume 4, pages 709–715, 2004.
- [23] N. Jansen, B. Könighofer, S. Junges, and R. Bloem. Shielded decision-making in MDPs. *arXiv preprint arXiv:1807.06096*, 2018.
- [24] S. Kolathaya and A. D. Ames. Input-to-state safety with control barrier functions. *IEEE control systems letters*, 3(1):108–113, 2019.
- [25] J. V. Messias, M. Spaan, and P. U. Lima. Efficient offline communication policies for factored multiagent pomdps. In *Advances in Neural Information Processing Systems*, pages 1917–1925, 2011.
- [26] Ian M Mitchell, Alexandre M Bayen, and Claire J Tomlin. A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on automatic control*, 50(7):947–957, 2005.
- [27] V. Murashov, F. Hearl, and J. Howard. Working safely with robot workers: Recommendations for the new workplace. *Journal of occupational and environmental hygiene*, 13(3):D61–D71, 2016.
- [28] Mitio Nagumo. Über die lage der integalkurven gewöhnlicher differentialgleichungen. *Proceedings of the Physico-Mathematical Society of Japan. 3rd Series*, 24:551–559, 1942.
- [29] Q. Nguyen, A. Hereid, J. W. Grizzle, A. D. Ames, and K. Sreenath. 3d dynamic walking on stepping stones with control barrier functions. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 827–834. IEEE, 2016.
- [30] F. A. Oliehoek, C. Amato, et al. *A concise introduction to decentralized POMDPs*, volume 1. Springer, 2016.
- [31] F. A. Oliehoek and M. T. J. Spaan. Tree-based solution methods for multiagent pomdps with delayed communication. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.
- [32] S. Prajna, A. Jadabaie, and G. J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428, 2007.
- [33] D. V. Pynadath and M. Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of artificial intelligence research*, 16:389–423, 2002.
- [34] S. Seuken and S. Zilberstein. Formal models and algorithms for decentralized decision making under uncertainty. *Autonomous Agents and Multi-Agent Systems*, 17(2):190–250, 2008.
- [35] G. Shani, J. Pineau, and R. Kaplow. A survey of point-based POMDP solvers. *Autonomous Agents and Multi-Agent Systems*, 27(1):1–51, 2013.
- [36] R. Sharan and J. Burdick. Finite state control of POMDPs with LTL specifications. In *2014 American Control Conference*, pages 501–508. IEEE, 2014.
- [37] R. D. Smallwood and E. J. Sondik. The optimal control of partially observable markov processes over a finite horizon. *Operations research*, 21(5):1071–1088, 1973.
- [38] M. Vasic and A. Billard. Safety issues in human-robot interactions. In *2013 IEEE International Conference on Robotics and Automation*, pages 197–204. IEEE, 2013.
- [39] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*, 48(27):54–61, 2015.