Analysis of the Effects of Perception Inconsistency Among Interacting Vehicles in Partially Observable Environments

Md Rayhanul Islam[1], Tichakorn Wongpiromsarn[1] and Samik Basu[1]

Abstract— **This paper analyzes the effect of inconsistencies in the overall system due to the differences in autonomous and human-driven vehicles' perception and decision-making capabilities. Owing to the partial observability of the environment, autonomous vehicles may construct different belief environment models depending on their perceptions to synthesize policies. The findings show that synthesizing policies without considering the differences in perception and decision-making of other agents may increase the probability of making the complete system unsafe. Finally, this approach is run on the autonomous vehicle simulator CARLA to support our findings.**

I. INTRODUCTION

In the real-world scenario, autonomous vehicles (also referred to as agents in this paper) coexist on the road with other (human-driven or other autonomous) vehicles and the environment (pedestrian), which is an external entity that interacts with vehicles. This complex and interactive world brings substantial challenges to autonomous vehicles. On top of that, not all autonomous vehicles possess the same perception and decision-making capabilities. Therefore, the differences in autonomy and capabilities in perception and decision-making among vehicles may introduce inconsistencies in the transportation network. Additionally, the presence of partial observability of environments may hinder autonomous vehicles from perceiving the *true* model of the environment. As a result of these inconsistencies and partial observability, different autonomous vehicles may exhibit different behaviors in the same situation and make the overall system unsafe by increasing the risk of accidents in the transportation network. For example, many rear-end collisions happen in autonomous vehicles due to inconsistencies in perception and decision-making [1], [2], [3].

When an agent/vehicle can fully observe its current state and make decisions based on complete environment information, its behavior is typically modeled by a Markov Decision Process (MDP) [4]. However, a partially observable MDP (POMDP) is introduced when vehicles cannot observe the underlying state of the system [5]. We consider autonomous vehicles, which employ sensors like LIDAR, Radars, GPS, and cameras to perceive the environment and gather information about their surroundings [6]. This information is used to detect, identify, and classify environments and objects, road signs, and traffic signals. However, a challenge arises when an autonomous vehicle fails to differentiate among environments or objects from its perception. For instance, a trash can might be misinterpreted as a pedestrian or vice versa. Therefore, in such a situation, even if autonomous vehicles know the current states of other environments, they may not be able to model their transition model representing their behaviors.

Thus, partial observability might be introduced in the overall system when autonomous vehicles fail to observe other environments' models. To address this partial observability, we consider the approach introduced in [7]. In this new perspective of partial observability, each autonomous vehicle maintains a set of transition models, where each transition model represents the possible behavior of other environments.

This paper proposed an approach to measure how inconsistencies in perception and decision-making among interactive agents affect the safety of the overall system, considering the partial observability of environments. In order to quantify safety, we construct the belief environment model from the set of potential transition models that each agent maintains. It is also needed to mention that different autonomous vehicles may construct different belief environment models due to their differences in perception and decision-making capabilities. Then, we extract policies from the complete system such that it maximizes the probability of satisfying the desired specification. After that, we apply policies on the true system and construct the induced Markov chain of the complete system to verify the desired specification. Our findings show that we can extend the original work [8] to quantify the probability of safety of the overall system by allowing agents to maintain a set of potential environment models in the presence of partial observability of environments. Finally, experimental result from a case study is presented along with simulation results on the autonomous driving simulator CARLA [9].

A. Contributions

In this paper, considering the partial observable environment, we seek answers to the following questions: 1. Given the partial observation of autonomous vehicles regarding the environment's model, how safe is the overall system when autonomous vehicles interact with other vehicles (autonomous or human-driven)? 2. How do the discretization of beliefs, discrepancies in perception, the different number of environment models, and the number of interacting agents impact computing the safety of a system considering the partially observable environment?

*This work was supported by NSF grant CNS-2211141.

[1]All authors are with the Department of Computer Science, Iowa State University, Ames, USA. rayhanul@iastate.edu

B. Motivating Example

For concertizing the problem, we consider this example from [8] where an autonomous vehicle A interacts with another autonomous/human-driven vehicle H and a pedestrian E, shown in Fig. 1. The pedestrian can move in and out of the pedestrian crossing with probability $p \in [0, 1]$. At any location, agents A and H have only two available actions: "go" or "stop." Fig. 2 provides the transition behavior of vehicles A and H and pedestrians E. The collective objective of both vehicles is to safely reach the goal state from their start state without colliding with pedestrians E or each other. The objective is expressed in linear temporal logic (LTL) formula $\varphi = (\neg crash \ \mathtt{U} \ goal)$, where a crash happens when vehicle A or H shares the same location with each other or pedestrian E, and the goal is state s_6.

II. Related Work

Existing literature concerning the interaction of heterogeneous agents focuses on policy synthesis, but they do not consider inconsistencies in decision-making and perception. Approaches [10], [11] discuss the uncertainty of the other agent's intent, such as pedestrians or human drivers, but these approaches suffer from the state explosion problem as the number of agents increases. Robust synthesis methods [12], [13] can handle many agents in an uncertain environment and do not face state explosion problems. However, these approaches do not consider the case when agents disagree on the uncertainty in their environments. Cai et al. [14] introduce a set of behavioral rules and constraints to guide interacting agents in achieving both safe and fair conflict resolution. They show that this research guarantees safety under all traffic conditions and liveness under sparse traffic conditions, considering the predefined assumptions about how all agents make decisions, but it does not consider differences in assumption.

This paper [13] explains the parameter synthesis problem for parametric MDP by formulating the problems as the product MDP and separating the parameter space into regions based on the satisfaction of the specification. Wongpiromsarn et al. [15] proposes an incremental approach to synthesize control policies for a non-independent heterogeneous multi-agents system to maximize the probability of satisfying the specification; additionally, to tackle the state explosion problem, it initially incorporates a small subset of agents in the synthesis procedure and then continues to add more agents until the limitation of computational resources is reached. Another paper [16] synthesizes policies for an agent by formulating a problem as a two-player stochastic game with multiple objectives. Approaches [17], [18], [19] use barrier function in control synthesis to remove unsafe transitions from the composed system. However, none of the works address the inconsistencies in potential assumptions about interacting agents' intent in synthesizing policies.

This paper [8] measures the effect of inconsistencies in perception and decision-making among agents when they do not consider the difference in perception and decision-making with other agents in computing policy in an observable environment. To the best of our knowledge, no approach has yet been proposed to quantify the probability of safety of a system, considering the partial observability of the environment in synthesizing policies.

III. Preliminaries

We denote a finite or countably infinite discrete set $\mathcal{X} \subset [0, 1]^n$, where $n \in \mathbb{N}$. A probability distribution over the set $\mathcal{X}$ is a function $\mu : \mathcal{X} \to [0, 1]$ such that $\sum_{x \in \mathcal{X}} \mu(x) = 1$. The cardinality of the set $\mathcal{X}$ is defined by $|\mathcal{X}|$.

A. System modeling and control policy

The control policy synthesis and verification process requires modeling the system's behavior using mathematical frameworks such as the Finite Transition System, Markov chain, and Markov Decision Process. A brief description of these mathematical frameworks is provided below.

Definition 3.1: Markov Decision Process [4]. A Markov Decision Process (MDP) is a tuple (S, A, s_0, P, AP, L) where S is the finite set of states, and A is the set of actions. The initial state is s_0, and the probability transition function is $P : S \times A \times S \to [0, 1]$ such that $\forall s \in S, \alpha \in A$, $\sum_{s' \in S} P(s, \alpha, s') \in \{0, 1\}$. The set of the atomic proposition is AP and the labeling function $L : S \to 2^{AP}$, which assigns a subset of AP to state $s \in S$ and $L(s) \subseteq AP$. We also define $ActS : S \to A$, which provides the set of enabled actions in a state. For $s \in S$, $ActS(s)$ is defined as $ActS(s) = \{\alpha \in A \mid \exists s' \in S, P(s, \alpha, s') \neq 0\}$. In addition, for $s \in S$, the function $Post : S \to S$ is defined as $Post(s) = \{s' \in S \mid \exists \alpha \in A, P(s, \alpha, s') \neq 0\}$.

Definition 3.2: Markov Chain [4]. A Markov Chain (MC) is a tuple (S, s_0, P, AP, L) with transition function $P : S \times S \to [0, 1]$ such that for all states $s \in S$, $\sum_{s' \in S} P(s, s') = 1$. The definitions of S, s_0, AP, and L are the same as defined in MDP. We can say that MDP is an MC if $|ActS(s)| \leq 1$ for all $s \in S$.

Definition 3.3: Finite Transition System [4]. A Finite Transition System (FTS) is a tuple $(S, A, s_0, \to, AP, L)$ with the transition function $\to \subseteq S \times A \times S$. We say $P(s, \alpha, s') = 1$ for any two states, $s, s' \in S$, and action $\alpha \in A$ if there exists a transition from state s to s' with action α. The definitions of S, s_0, AP, and L are the same as defined in MDP. S, A, and AP are finite for the FTS.

Definition 3.4: Path [4]. A finite path π of an MDP/FTS M is a sequence of states $s_0 s_1, \ldots, s_n$ such that $s_i \in Post(s_{i-1})$ for all $0 < i \leq n$ and $n \geq 0$. $Last(\pi)$ is the last state of π, and all finite paths of M is a set $Paths^M_{fin}$.

Definition 3.5: Policy [4]. A policy σ for an MDP/FTS M is a function $\sigma : Paths^M_{fin} \to A$ where $\sigma(\pi) \in ActS(Last(\pi))$ for all $\pi \in Paths^M_{fin}$.

Definition 3.6: Induced Markov Chain [8]. Consider an MDP/FTS M and a policy σ, then the MC induced by M and σ is $M^\sigma = (Paths^M_{fin}, A, s_0, P^\sigma, AP, L^\sigma)$ where

$$P^\sigma(\pi, \pi') = \begin{cases} P(Last(\pi), \sigma(\pi), s'), & \text{if } \pi' = \pi s' \\ 0, & \text{otherwise} \end{cases}$$

$L^\sigma(\pi) = L(Last(\pi))$, and the initial state s_0 and atomic proposition AP of M^σ remain same with M.

Definition 3.7: Product MDP/FTS [8]. Let's consider two MDP/FTS $M_1 = (S_1, A_1, s_{1_0}, P_1, AP_1, L_1)$ and $M_2 = (S_2, A_2, s_{2_0}, P_2, AP_2, L_2)$. The parallel composition of M_1 and M_2 is defined as $M_1 \parallel M_2 = (S_1 \times S_2, A_1 \times A_2, s_{1_0} \times s_{2_0}, P, AP_1 \cup AP_2, L)$, where $L(s_1 \times s_2) = L(s_1) \cup L(s_2)$. For any two state (s_1, s_2) and (s_1', s_2'), and action $(\alpha_1, \alpha_2) \in A_1 \times A_2$, the transition function $P : S_1 \times S_2 \times A_1 \times A_2 \times S_1 \times S_2 \rightarrow [0, 1]$ is defined as $P((s_1, s_2), (\alpha_1, \alpha_2), (s_1', s_2')) = P_1(s_1, \alpha_1, s_1') \times P_2(s_2, \alpha_2, s_2')$ if $P_1(s_1, \alpha_1, s_1') > 0$ and $P_2(s_2, \alpha_2, s_2') > 0$. If M_2 is an MC $M_2 = (S_2, s_{2_0}, P_2, AP_2, L_2)$, the parallel composition of M_1 and M_2 is defined as $M_1 \parallel M_2 = (S_1 \times S_2, A_1, s_{1_0} \times s_{2_0}, P, AP_1 \cup AP_2, L)$.

B. System modeling under partial observability

Partial observability of the environment requires agents to maintain probability distribution as a belief to construct the system model. Here, we discuss some definitions needed to address the partial observability of the environment.

Definition 3.8: δ-discretization of a vector space [20]. A δ-discretization of a vector space $[0, 1]^n$ with $n \in \mathbb{N}$, and $\delta \in [0, 1]$ is defined as $\Xi(n, \delta) = \{\sum_{i=1}^{n} \beta_i v_i \mid \beta_i = k\delta \;\&\; k\delta \leq 1\}\}$. Here, v_i is the basis vector, where $v_{i_j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$, β_i is the coefficients and $k \in \{0, 1, 2, 3, 4, 5, \ldots\}$. For each vector $\vec{v}_i = (v_{i_1}, \ldots, v_{i_n}) \in \Xi(n, \delta)$, each vector component $v_{i_j} \in \vec{v}_i$ is divided by L1 norm $\|\vec{v}_i\|_1$ to ensure that $\sum_{1 \leq j \leq n} v_{i_j} = 1$.

Definition 3.9: Belief Function [21]. Consider agent A maintains the behavior of an Environment E as a set of MCs $\mathbb{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n\}$ considering the partial observability of the environment, where each MC $\mathcal{D}_i = (S, s_0, P_i, AP, L)$. Here, an agent can observe the states of the environment E. However, the specific model $\mathcal{D}_i \in \mathbb{D}$ representing the environment model is not directly observable. Therefore, agent maintains a belief function $\mu : \mathbb{D} \rightarrow [0, 1]$ such that $\sum_{\mathcal{D}_i \in \mathbb{D}} \mu(\mathcal{D}_i) = 1$, and all beliefs form belief space Γ. For representational simplicity, we allow agent A to maintain a belief vector $\mathcal{B} = [\mu(\mathcal{D}_1), \mu(\mathcal{D}_2), \ldots, \mu(\mathcal{D}_n)] \in [0, 1]^n$ in each state such that $\mu(\mathcal{D}_1) + \mu(\mathcal{D}_2) + \cdots + \mu(\mathcal{D}_n) = 1$. All such belief vectors form the belief vector space $\mathbb{B} \subseteq [0, 1]^n$, where $\mu(\mathcal{D}_i) \in \mathcal{B} \in \mathbb{B}$ indicates the probability of using the model $\mathcal{D}_i$ by the environment E. We use $\mu(\mathcal{D}_i)$ or $\mathcal{B}_i$ to represent the i^{th} component of vector $\mathcal{B}$. To facilitate belief updates, agent A employs a function $\gamma : S \times S \times \mathbb{B} \times \mathbb{D} \rightarrow \Gamma$ and updates belief when it observes E transitioning from state s to s'. If A's current belief vector is $\mathcal{B}$ at state s, then each of the component of the next belief vector $\mathcal{B}'$ at state s' is determined using the function γ, which applies Bayes' rule [22] as $\mathcal{B}_i' = \gamma(s, s', \mathcal{B}, \mathcal{D}_i) = \frac{1}{\eta} \times \mathcal{B}_i \times P_i(s, s')$. Here, $\mathcal{B}_i \in \mathcal{B}$ is the i^{th} component of vector $\mathcal{B}$, and $P_i(s, s')$ is the transition probability in MC $\mathcal{D}_i$. The normalization factor is $\eta = \sum_{\mathcal{D}_k \in \mathbb{D}} \mathcal{B}_k \times P_k(s, s')$.

The belief space $\mathbb{B}$ can be infinite, resulting in intractability due to its exponential growth. To overcome this, we employ discretization, which selects a finite number of representative points from the belief space $\mathbb{B}$ depending on δ. Let's assume the δ discretization over the belief space $\mathbb{B}$ is $\mathbb{B}_\delta = \Xi(n, \delta) \subset \mathbb{B}$ following *Definition 3.8*. For any belief vector $\mathcal{B}' \in \mathbb{B}$, the function $\tau : 2^{\mathbb{B}} \times \mathbb{B} \rightarrow \mathbb{B}$, which returns the associated belief representative point $\mathcal{B} \in \mathbb{B}_\delta$, is defined as $\tau(\mathbb{B}_\delta, \mathcal{B}') = \mathcal{B}$, if $\mathcal{B} = \arg\min_{\mathcal{B}_\delta \in \mathbb{B}_\delta} \|\mathcal{B}' - \mathcal{B}_\delta\|_2$.

Definition 3.10: Construction of the complete Environment Model [23]. Recall agent A that maintains a set of MCs $\mathbb{D}$, and the finite δ discretized belief space $\mathbb{B}_\delta$. Now, the agent builds the complete environmental model as an MC $M^{env} = (S \times \mathbb{B}_\delta, s_0 \times \mathcal{B}_{init}, P_{env}, AP, L)$ where the AP and L of the environmental model remain same as in the original MC. For any two belief states $(s, \mathcal{B})$, and $(s', \mathcal{B}')$, the expected transition probability $P_{env}((s, \mathcal{B}), (s', \mathcal{B}')) = \frac{1}{\eta} \sum_{i=1}^{n} \mathcal{B}_i' \times P_i(s, s')$ if $\mathcal{B}' = \tau(\mathbb{B}_\delta, \mathcal{B}_{s'}) \;\&\; \mathcal{B}_{s'} = [\gamma(s, s', \mathcal{B}, \mathcal{D}_1), \ldots, \gamma(s, s', \mathcal{B}, \mathcal{D}_n)]$, and otherwise 0. Here, $\mathcal{B}_i'$ is the i^{th} component of belief vector $\mathcal{B}'$. $P_i(s, s')$ is the transition probability in MC $\mathcal{D}_i$. The normalization factor $\eta = \sum_{(s'', \mathcal{B}'') \in S \times \mathbb{B}_\delta} \sum_{\mathcal{D}_k \in \mathbb{D}} \mathcal{B}_k'' \times P_k(s, s'')$. η ensures the outgoing probability of a state to be 1 and makes the complete environmental model an MC [23].

Definition 3.11: Perception Discrepancy. If T_j be the transition matrix, then the Frobenius norm [24] of T_j is defined as $\|T_j\|_F = \sqrt{\sum_{k=1}^{K} \sum_{l=1}^{L} |t_{kl}|^2}$, where t_{kl} represents the element in the k^{th} row and j^{th} column of matrix T_j. As agent A maintains m number of MCs, if their corresponding matrix presentation becomes $\{T_1, T_2, \ldots T_m\}$, then we define perception discrepancy between m number of transition matrices as $\varrho = \arg\max_{i, j \in \{1, \ldots, m\}} \|T_j - T_i\|_F$.

IV. Problem Statement

Consider a set of agents $\{A_1, A_2, \ldots, A_n\}$, interacting with an environment E. Each agent A_i is characterized by an MDP/FTS $M_i^a = (S^a, A, s_0^{a_i}, P^a, AP^a, L^a)$ for $i \in 1, \ldots, n$. Agents have partial observation regarding the model of E, and no agent knows the true model D_T of environment E. Consequently, agent A_i considers the environment model as a set of MCs $\mathbb{D}_i = \{\mathcal{D}_{i_1}, \mathcal{D}_{i_2}, \mathcal{D}_{i_3}, \ldots, \mathcal{D}_{i_m}\}$ where each MC $\mathcal{D}_{i_j} \in \mathbb{D}_i$ is modeled as $\mathcal{D}_{i_j} = (S^d, s_0^d, P_{i_j}^d, AP^d, L^d)$. Agent A_i lacks information regarding which $\mathcal{D}_{i_j} \in \mathbb{D}_i$ represents the true environment model D_T. The true complete system model considering the true environment model D_T is $S_T = M_1^a \parallel M_2^a \parallel \cdots \parallel M_n^a \parallel D_T$. We assume that agents possess knowledge regarding the model of other agents but are incapable of communicating with each other.

As agent A_i does not know the specific MC $\mathcal{D}_{i_j} \in \mathbb{D}_i$ that corresponds to the true environmental model D_T, it hinges upon the belief environment model. Let us assume each agent A_i constructs a complete belief environment model D_i^b following *Definition 3.10*, where each transition represents the expected probability. As a result, the complete

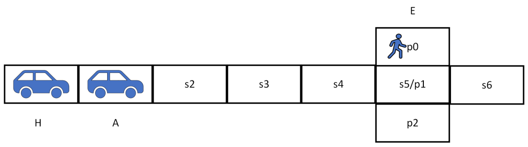


Fig. 1: Road and its discretized grids in the example scenario along with two vehicles on a straight street and a pedestrian in a crosswalk. The autonomous vehicle, A, originates from position s_1, while vehicle H starts from s_0. The stochastically moving pedestrian E begins at p_0 and moves into and out of the crosswalk with probability $p \in [0, 1]$. Grid cells s_5 and p_1 correspond to crosswalks on the road.

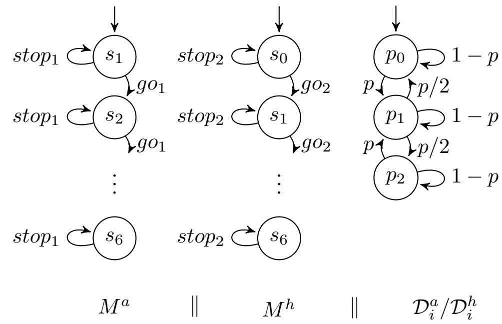


Fig. 2: The model description of both vehicles A, and H represented as M^a and M^h, respectively, and a single MC model $\mathcal{D}^a_i \in \mathbb{D}^a$ or $\mathcal{D}^h_i \in \mathbb{D}^h$ with probability $p \in [0, 1]$ of crossing the road.

system from the perspective of agent A_i is represented as $S_i = M^a_1 \parallel M^a_2 \parallel \cdots \parallel M^a_n \parallel D^b_i$, where $M^a_1, M^a_2, \ldots, M^a_n$ correspond to the models of other agents. All agents strive to maximize the probability that the complete system satisfies the provided LTL specification φ, believing that their observations contain the true model of the environment. Let $\sigma_i : Path^{S_i}_{fin} \to A$ denotes the policy of the complete system for agent A_i, such that σ_i maximizes $Pr_E(S_i \models \varphi)$.

Problem 1 Analysis of observation inconsistencies. Given the models $M_1, M_2, \ldots, M_n$ of the autonomous agents along with their observations of the environments as a set of MCs $\mathbb{D}_1, \mathbb{D}_2, \ldots, \mathbb{D}_n$, their belief discretization value δ, the true environment model D_T and specification φ, then what is the probability that $S_T \models \varphi$ provided that S_T be the true model of the complete system and σ_i is the policy of agent A_i.

Motivating Example (continued). To successfully transit the crosswalk and reach the goal state s_6, the autonomous vehicle A constructs its model M^a and model M^h for vehicle H and a set of MC models $\mathbb{D}^a$ for pedestrian E. Let us assume $\mathbb{D}^a = \{\mathcal{D}^a_1, \mathcal{D}^a_2\}$, and each $\mathcal{D}^a_i \in \mathbb{D}^a$ corresponds to the pedestrian's possible model with probability $p \in [0, 1]$ of moving forward, as shown in Fig. 2. The belief function of the autonomous vehicle A is $\mu : \mathbb{D}^a \to [0, 1]$ such that $\sum_{\mathcal{D}^a_i \in \mathbb{D}^a} \mu(\mathcal{D}^a_i) = 1$, and the belief vector is $\mathcal{B} = [\mu(\mathcal{D}^a_1), \mu(\mathcal{D}^a_2)]$ such that $\mu(\mathcal{D}^a_1) + \mu(\mathcal{D}^a_2) = 1$. The initial belief is $\mu(\mathcal{D}^a_1) = \mu(\mathcal{D}^a_2) = 0.5$, and belief vector $\mathcal{B}_{init} = [0.5, 0.5]$. Autonomous vehicle A updates its belief vector whenever it observes a transition in pedestrian E from one state to another.

The belief environment model $\mathcal{D}^b_a$ of agent A is constructed according to *Definition 3.10* considering the discretized belief space $\mathbb{B}^\delta_a$. In the next step, the composed system model for agent A is $S_a = M^a \parallel M^h \parallel \mathcal{D}^b_a$. Now, autonomous vehicle A synthesizes policy σ_a such that it maximizes the probability $Pr_E(S_a \models \varphi)$. Similarly, vehicle H forms the complete system S_h, and synthesizes policy σ_h to maximizes $Pr_E(S_h \models \varphi)$.

V. Policy Extraction and Analysis

In this section, we describe the analysis process of the composed system, where policies are extracted based on the assumptions of each agent. Consider a composed system of n decision-making agents $A_1, A_2, \ldots, A_n$, and their corresponding policies $\sigma_1, \sigma_2, \ldots, \sigma_n$ respectively.

A. Policy Extraction

Once A_i constructs the composed system $S_i = M_1 \parallel M_2 \parallel \cdots \parallel M_n \parallel D^b_i$, it then extracts policy σ_i that maximizes the probability $Pr_E(S_i \models \varphi)$. As the system is represented as the composition of subsystems, agent A_i's policy σ_i becomes the composition of policies of n agents. We can express it as $\sigma_i :\parallel_{j=1:n} \sigma_{i|j}$, where $\sigma_{i|j}$ is the projection of policy σ_i under agent A_i's assumption of agent A_j's decisions. Each agent A_i assumes other agents in the system S_i implement policy σ_i and build policies accordingly. Consequently, agent A_i implements $\sigma_{i|i}$. Similarly, all agents extract their policies, and the implemented policy for n agents becomes $\sigma :\parallel_{i=1:n} \sigma_{i|i}$.

Motivating Example (continued). The policy σ_a for the autonomous vehicle A is extracted from the composed system S_a to maximize the probability $Pr_E(S_a \models \varphi)$. The σ_a is defined as $\sigma_a : \sigma_{a|a} \parallel \sigma_{a|h}$, where $\sigma_{a|h}$ is the projection of A's policy σ_a under the assumption of H's decision. As A believes H is also implementing σ_a, therefore, A also implements $\sigma_{a|a}$. Similarly, vehicle H's policy $\sigma_{h|h}$ is extracted from the composed system $S_h = M^h \parallel M^a \parallel \mathcal{D}^b_h$ for the same property, φ. Finally, the implemented policy for the composed system is $\sigma = \sigma_{a|a} \parallel \sigma_{h|h}$.

Consider a state (s_4, s_3, p_1) in the composed system S_a, where vehicle A is situated in state s_4, H is in s_3 and the pedestrian is at crosswalk state p_1. The optimal policy for vehicle A in this state is to execute action $stop_1$ in order to maximize the probability $Pr_E(S_a \models \varphi)$ because otherwise it will violate the specification φ and hit pedestrian E. Furthermore, vehicle A operates under the assumption that vehicle H is aware of the fact that the pedestrian is crossing the crosswalk. Therefore, vehicle H at state s_3 will follow policy $stop_2$, while in other states, vehicle H has the flexibility to select any action as long as $Pr_E(S_h \models \varphi) > 0$. However, we assume H selects a policy that also maximizes $Pr_E(S_h \models \varphi)$. So, the policy in state (s_4, s_3, p_1) for A is $\sigma_a((s_4, s_3, p_1)) = stop_1$. Similarly, vehicle A builds its complete policy σ_a for all states. Likewise, vehicle H also extracts its policy σ_h from the composed system S_h such it maximizes $Pr_E(S_h \models \varphi)$. So, the complete policy at state (s_4, s_3, p_1) is $\sigma(s_4, s_3, p_1) = \sigma_a(s_4, s_3, p_1) \parallel \sigma_h(s_4, s_3, p_1) = (stop_1, stop_2)$. Similarly, the complete policy for all states is computed.

B. System Analysis

To facilitate analysis, we introduce the true model D_T of the environment, which accurately represents its behavior. Considering the true environment model D_T, the complete system involving n agents is represented as $S_T = M_1 \parallel M_2 \parallel \cdots \parallel M_n \parallel D_T$. Now, the extracted policy σ is applied to S_T following *Definition 3.6*, and the resultant true system becomes an induced MC S_T^σ. We then compute the probability $Pr(S_T^\sigma \models \varphi)$, which represents the probability of satisfying the given specification φ in the true complete system S_T. This probability value depends on the policies of each agent because the induced complete system S_T^σ contains only transitions present in the σ.

Motivating Example (continued). Recall the true complete system S_T, and extracted policy $\sigma = \sigma_{a|a} \parallel \sigma_{h|h}$ from the composed system S_a, and S_h. Each state in S_T has all possible combinations of enabled actions depending on the next reachable state. However, when a policy is applied on S_T, the resultant true system becomes an MC S_T^σ, which has only actions that are present in σ. To illustrate, let us examine a specific state (s_4, s_3, p_1) within S_T. As previously discussed, the extracted policy for this state is $\sigma(s_4, s_3, p_1) = (stop_1, stop_2)$. Although there may exist four potential action choices, namely $(go_1, go_2), (go_1, stop_2), (stop_1, go_2), (stop_1, stop_2)$ in (s_4, s_3, p_1) of S_T. However, once the policy $\sigma = \sigma_{a|a} \parallel \sigma_{h|h}$ is applied, the resultant S_T^σ will have only action $(stop_1, stop_2)$ in (s_4, s_3, p_1) because this action maximizes $Pr_E(S_a \models \varphi)$ and $Pr_E(S_h \models \varphi)$.

VI. Experimental Results

This result section assesses the safety probability of the overall system by measuring the inconsistencies within it in a vehicle-pedestrian example. A higher probability value indicates enhanced safety in the overall system; a lower probability value means the opposite. Additionally, it shows the effect of belief discretization δ, perception discrepancy ϱ, the cardinality of environment model set m, and the number of agents on the composed system's construction, synthesis, and verification time.

We built a set of pedestrian models to conduct our experiment by generating the pedestrian moving probability p. For the analysis purpose, we considered the true pedestrian probability, $p = 0.65$, of crossing the road. We implemented a Python tool [1] utilizing probabilistic model checkers Stormpy [25] and PRISM [26]. The experiment was performed on a computer with an Intel core i7-4770, 3.40GHz processor. Finally, we validate our experimental result in an autonomous vehicle simulator CARLA.

TABLE I: The impact of belief discretization δ on various properties of the environment model and the complete system, including construction time, number of states, policy synthesis, and more. In these experiments, the set of MCs $|\mathbb{D}^a| = |\mathbb{D}^h| = 2$ for both agent A and H, and their corresponding p are 0.63 and 0.83. The true environment model with pedestrian moving probability 0.65. 1st column: δ- belief discretization value, 2nd column: Env. cons. time- the average construction time of the belief environment model, 3rd column: Env. States- the average number of states in the belief environment model, 4th column: Synthesis time- the average policy synthesis time for an agent, 5th column: Composed state- the average number of states in the composed system, and 6th: Verification time- the average verification time of the specification φ.

δ	Env. Cons. time	Env. States	Synthesis time	Composed States	Verification time
0.01	-	-	-	-	-
0.05	0.564	60	17.789	22680	35.566
0.10	0.017	15	1.394	5670	2.805
0.15	0.005	9	0.575	3402	1.164
0.20	0.001	6	0.261	2268	0.546

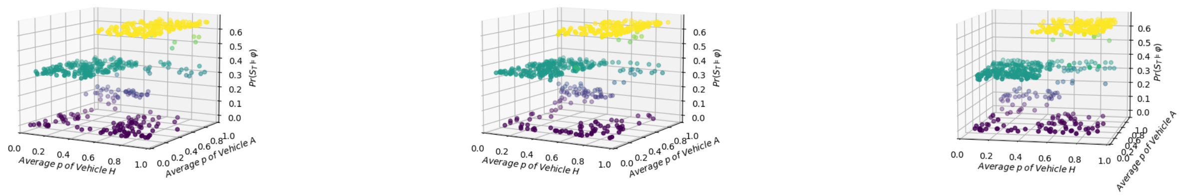
A. Impact of belief discretization

Due to exponential growth in the belief spaces, agents apply a belief discretization approach to work on the finite representative points from the belief space (*Definition 3.8*). Table I demonstrates how the belief discretization δ impacts the construction time of the belief environment model and the composed system. It also illustrates the impacts of δ on the average policy synthesis time and verification time. The outcome indicates that a smaller δ increases the construction time and the number of states in both the belief environment model and the composed system. This happens because smaller δ results in a higher number of belief points in the discretized belief space. For example, when δ was set to 0.01, it took longer execution time due to the large number of belief states in the discretized belief space, and we terminated after 2 hours. Since it is impractical to present experimental results for all possible values of δ, we showed the construction, policy synthesis, and verification time for some specific $\delta \in \{0.01, 0.05, 0.10, 0.15, 0.20\}$. In summary, the choice of δ influences the computational resources required to construct the belief environment model. In our study, we opted for a $\delta = 0.20$ value that ensures computational efficiency.

B. Impact of the increasing number of environment models

Table II demonstrates that an increase in the number of environment models results in higher construction times for both the belief environment model and the composed system, longer policy synthesis times, and increased verification time in the induced true system. However, the number

[1] https://tinyurl.com/4tmrsfdp



(a) $|\mathbb{D}^a| = |\mathbb{D}^h| = 2$ (b) $|\mathbb{D}^a| = |\mathbb{D}^h| = 4$ (c) $|\mathbb{D}^a| = |\mathbb{D}^h| = 8$

Fig. 3: The overall system safety, represented as the probability $Pr(S_T^\sigma \models \varphi)$ considering the partial observability of the environment where agent A and H maintain an MCs set $\mathbb{D}^a$ and $\mathbb{D}^h$ respectively. The experiment is conducted 500 times, and each time, agents are allowed to construct their belief pedestrian model based on their observations. Each point in this figure means the average sample assumptions of an agent and the probability of satisfying φ.

of states in both the belief environment and composed system model remains unchanged due to the fixed δ.

In Fig. 4, the box plot illustrates the changes in the mean, median, and standard deviation of the probability $Pr(S_T^\sigma \models \varphi)$ over the different number of environment models. It is evident that when agents maintain 2 MCs, the mean, median, and standard deviation of $Pr(S_T^\sigma \models \varphi)$ are 0.26, 0, and 0.30, respectively. This value continues to increase or remain the same for the mean and median $Pr(S_T^\sigma \models \varphi)$ as we increase the number of MCs. However, the standard deviation of $Pr(S_T^\sigma \models \varphi)$ decreases as we increase the number of MCs. So, it can be concluded that increasing the number of environment models increases the probability of $Pr(S_T^\sigma \models \varphi)$. This occurs because employing more environment models increases the probability of constructing a belief environment model that corresponds to the true behavior of the environment.

Another important consideration is that the impact of increasing the number of environment models on $Pr(S_T^\sigma \models \varphi)$ depends on the perception discrepancy of the maintained MCs with the true environment model D_T. For instance, when the pedestrian's moving probability in D_T is 0.65, and the agent maintains two MCs with pedestrian moving probabilities of 0.05 and 0.35, the increasing MCs with probabilities between 0.05 and 0.35 do not increase the probability $Pr(S_T^\sigma \models \varphi)$. This happens because the perception discrepancy ϱ of the true environment model D_T and these environment models with probabilities of 0.05 and 0.35 exceeds the perception discrepancy ϱ environment models with probabilities of 0.05 and 0.35. In contrast, when the perception discrepancy of the actual D_T with all observed environment models is less than or equal to the perception discrepancy ϱ, then increasing MCs increases the $Pr(S_T^\sigma \models \varphi)$. For instance, when the agent maintains two MCs with pedestrian moving probabilities of 0.5 and 0.8, increasing the number of MCs is more likely to result in a higher probability of satisfying φ.

TABLE II: The impact of increasing the number of environment models on $Pr(S_T^\sigma \models \varphi)$ and the construction, synthesis, and verification time. 1st column: # MC- the number of environment models.

# MC	Env. cons. time	Synthesis time	Verification time
2	0.00146	0.08321	0.20154
3	0.00208	0.10421	0.24275
4	0.00285	0.12720	0.29819
5	0.00619	0.19378	0.43219
6	0.01152	0.41674	0.86327

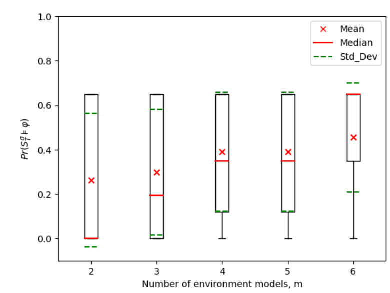


Fig. 4: The impact of different environment models m on the mean, median, and standard deviation of $Pr(S_T^\sigma \models \varphi)$.

C. Impact of perception discrepancy

Table III demonstrates the impacts of perception discrepancy ϱ on various aspects of the composed system. The result shows the number of states and construction time of the belief environment model and complete system increases with the increase of ϱ. Similarly, the policy synthesis time of the complete system and the verification time of the true system also increase with the increase in ϱ.

Fig. 5 shows the impacts of perception discrepancy ϱ on the mean, median, and standard deviation of $Pr(S_T^\sigma \models \varphi)$. Here, the mean and median value of $Pr(S_T^\sigma \models \varphi)$ decreases with the increase in ϱ. For example, the mean, median, and standard deviation of $Pr(S_T^\sigma \models \varphi)$ are 0.39, 0.43, and 0.26, respectively, when ϱ remains less than or equal to 0.234 among environment models, and, the mean, median, and standard deviation of $Pr(S_T^\sigma \models \varphi)$ continues

TABLE III: The effect of agent perception discrepancy ϱ on $Pr(S_T^\sigma \models \varphi)$ and other characteristics of the belief environment model and complete system. 1st column: ϱ- the perception discrepancy and the rest of the columns' names are the same as in the previous table.

ϱ	Env. cons. time	Env. States	Synthesis time	Composed states	Verification time
0.117	0.003	3	0.165	1134	0.353
0.234	0.045	6	0.342	2268	0.709
0.351	0.039	12	0.985	4536	1.990
0.469	0.066	15	1.394	5670	2.806
0.586	0.109	21	2.528	7938	5.050

TABLE IV: The average number of states, policy synthesis time, and verification time for different numbers of agents. 1st column: Road grid size- the number of discretized states in the road, 2nd column: # agents- number of agents.

Road grid size	# agents	Composed states	Synthesis time	Verification time
6	1	54	0.003	0.008
6	2	2268	0.165	0.353
7	3	81648	145.851	291.724
8	4	1469664	2965.958	5986.399

to decrease as we increase the perception discrepancy ϱ, and finally these values become 0.26 and 0, and 0.30 when $\varrho = 0.586$. Overall, $Pr(S_T^\sigma \models \varphi)$ decreases as we increase the value of ϱ. This happens because when the environment models have high ϱ with the true environment model, the constructed belief environment model does not represent the true environment model, which results in the lower $Pr(S_T^\sigma \models \varphi)$. On the other hand, when ϱ is small with the true environment model, the belief environment model accurately represents the true behavior of the environment, which results in higher $Pr(S_T^\sigma \models \varphi)$.

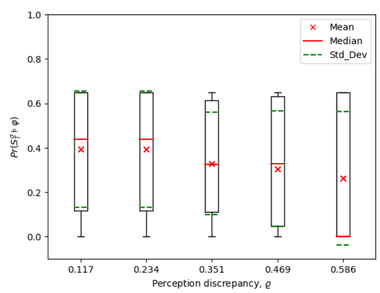


Fig. 5: The impact of perception discrepancy ϱ on the mean, median and standard deviation of $Pr(S_T^\sigma \models \varphi)$.

D. Comparison of different numbers of agents

Table IV summarizes the results for different numbers of agents. The average number of states in the composed system grows exponentially as we increase the number of agents. Therefore, it is computationally challenging to deal with a higher number of agents.

E. Verification of specification as the measure of safety

Fig. 3 showcases $Pr(S_T^\sigma \models \varphi)$ as the measure of the safety of the overall system considering the partial

TABLE V: The number of sample assumptions by both agents satisfying $Pr(S_T^\sigma \models \varphi)$ with different probabilities. 1st column: # MC- the number of MCs each agent maintains, 2nd column: Pr>0.40 - The number of samples satisfying $Pr(S_T^\sigma \models \varphi) > 0.40$, 3rd column: 0.2 < Pr $\leq$ 0.40: the number of samples with $Pr(S_T^\sigma \models \varphi)$ in between 0.2 and 0.40, 4th column: $Pr \leq 0.2$- the number of samples with $Pr(S_T^\sigma \models \varphi) \leq 0.2$, and 5th column: Average Pr- the average values of $Pr(S_T^\sigma \models \varphi)$.

# MC	Pr>0.40	0.2 < Pr $\leq$ 0.40	Pr $\leq$ 0.20	Average Pr
2	134	218	148	0.32
4	157	198	145	0.35
6	163	221	116	0.36
8	163	221	116	0.36

observability of environments. The higher $Pr(S_T^\sigma \models \varphi)$ implies a high probability of safety and low inconsistencies in the overall system. On the other hand, the lower $Pr(S_T^\sigma \models \varphi)$ indicates the opposite scenario. In this figure, we can separate the sample assumptions of agents into three regions: purple, yellow, and green. The purple region highlights the sample assumptions made by both vehicles, leading to $Pr(S_T^\sigma \models \varphi) = 0$. This happens when both agent's assumptions are inconsistent with each other. In contrast, the yellow region indicates assumptions of the pedestrian's probability of moving forward by vehicles A and H, resulting in the highest probability $Pr(S_T^\sigma \models \varphi)$. The highest $Pr(S_T^\sigma \models \varphi)$ is achieved because both vehicles are consistent with assumptions and observe the pedestrian similarly. Additionally, the green region shows the assumptions of both agents where the probability of satisfying specification is around 0.35. This occurs because both agents agree with some actions in the policy and disagree with the remaining others.

The number of samples in each region from Fig. 3 is listed in Table V. We see here that when the number of MCs is equal to 2, the number of samples satisfying $Pr(S_T^\sigma \models \varphi) \leq 0.2$ is 160, and this number continues to reduce as we increase the number of environment models. For example, the number of samples satisfying $Pr(S_T^\sigma \models \varphi) \leq 0.2$ is 116 when the number of MCs is 8. Similarly, the number of samples satisfying $Pr(S_T^\sigma \models \varphi) > 0.4$ increases with the number of MC models. It can be decided from this table that increasing the number of environment models increases the probability of satisfying the specification $Pr(S_T^\sigma \models \varphi)$.

F. Simulated Environment

We simulate the case study of two vehicles and a pedestrian example on CARLA 0.9.14 [9]. We run the simulation 500 times, and in each simulation, we govern the behaviors of each agent based on the extracted policy from our previous experiment. To extract policy, we considered the experiment where each agent maintains two MC models as the behavior of the pedestrian (environment). In order to implement our policy on CARLA, we discretized the road similar to our experiment setup. We then configure the behavior of each agent based on the derived policy. In this simulation, the crash happens when autonomous and human-driven vehicles are inconsistent with each other's policy. The result from the simulation supports our experimental results of measuring safety considering the partial observability of environments. Results show that the average value of $Pr(S_T^\sigma \models \varphi)$ in the simulation is around 0.33, which is around 0.35 in the actual experiment.

VII. CONCLUSIONS

This paper quantifies the probability of safety of the overall system among interacting heterogeneous agents (autonomous and human-driven) considering partial observability of the environment. In order to work with a partially observable environment, agents consider the behavior of the environment as a set of Markov chains and construct a belief environment model to form the complete system. Agents extract policies from the complete system without accounting for the potential inconsistencies among other agents. The findings show that constructing policies without considering the inconsistencies among other agents may increase the probability of making the complete system unsafe. As a result, the probability that the complete system fails to satisfy the desired specification is higher. Additionally, this approach is run on the CARLA simulator, and the result from CARLA also supports our findings.

REFERENCES

[1] W. Zhan, C. Liu, C.-Y. Chan, and M. Tomizuka, "A non-conservatively defensive strategy for urban autonomous driving," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 459–464.

[2] "California dmv," https://www.dmv.ca.gov/portal/vehicle-industry-services/autonomous-vehicles/autonomous-vehicle-collision-reports/, 2022, accessed: 2023-08-25.

[3] J. Stewart, "Why people keep rear-ending self-driving cars," Oct 2018. [Online]. Available: https://www.wired.com/story/self-driving-car-crashes-rear-endings-why-charts-statistics/

[4] C. Baier and J.-P. Katoen, *Principles of model checking*. MIT press, 2008.

[5] R. D. Smallwood and E. J. Sondik, "The optimal control of partially observable markov processes over a finite horizon," *Operations research*, vol. 21, no. 5, pp. 1071–1088, 1973.

[6] J. Vargas, S. Alsweiss, O. Toker, R. Razdan, and J. Santos, "An overview of autonomous vehicles sensors and their vulnerability to weather conditions," *Sensors*, vol. 21, no. 16, p. 5397, 2021.

[7] T. Wongpiromsarn and E. Frazzoli, "Control of probabilistic systems under dynamic, partially known environments with temporal logic specifications," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 7644–7651.

[8] S. Carr, T. Wongpiromsarn, and U. Topcu, "Quantifying faulty assumptions in heterogeneous multi-agent systems," in *2023 7th IEEE Conference on Control Technology and Applications*, 2023, to appear.

[9] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[10] T. Bandyopadhyay, K. S. Won, E. Frazzoli, D. Hsu, W. S. Lee, and D. Rus, "Intention-aware motion planning," in *Algorithmic Foundations of Robotics X: Proceedings of the Tenth Workshop on the Algorithmic Foundations of Robotics*. Springer, 2013, pp. 475–491.

[11] A. Jain, H. S. Koppula, B. Raghavan, S. Soh, and A. Saxena, "Car that knows before you do: Anticipating maneuvers via learning temporal driving models," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 3182–3190.

[12] M. Cubuktepe, Z. Xu, and U. Topcu, "Policy synthesis for factored mdps with graph temporal logic specifications," *arXiv preprint arXiv:2001.09066*, 2020.

[13] S. Junges, E. Abraham, C. Hensel, N. Jansen, J.-P. Katoen, T. Quatmann, and M. Volk, "Parameter synthesis for markov models," 2019.

[14] K. X. Cai, T. Phan-Minh, S.-J. Chung, and R. M. Murray, "Rules of the road: Formal guarantees for autonomous vehicles with behavioral contract design," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1853–1872, 2023.

[15] T. Wongpiromsarn, A. Ulusoy, C. Belta, E. Frazzoli, and D. Rus, "Incremental synthesis of control policies for heterogeneous multi-agent systems with linear temporal logic specifications," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 5011–5018.

[16] T. Chen, M. Kwiatkowska, A. Simaitis, and C. Wiltsche, "Synthesis for multi-objective stochastic games: An application to autonomous urban driving," in *Quantitative Evaluation of Systems: 10th International Conference, QEST 2013, Buenos Aires, Argentina, August 27-30, 2013. Proceedings 10*. Springer, 2013, pp. 322–337.

[17] D. S. Sundarsingh, J. Bhagiya, J. Chatrola, A. Saoud, P. Jagtap, *et al.*, "Controller synthesis for local and global specifications in multi-agent systems," *arXiv preprint arXiv:2304.01652*, 2023.

[18] M. Mizoguchi and T. Ushio, "Abstraction-based symbolic control barrier functions for safety-critical embedded systems," *IEEE Control Systems Letters*, vol. 6, pp. 1436–1441, 2021.

[19] M. Srinivasan, S. Coogan, and M. Egerstedt, "Control of multi-agent systems with finite time control barrier certificates and temporal logic," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 1991–1996.

[20] D. Grover and C. Dimitrakakis, "Adaptive belief discretization for pomdp planning," *arXiv preprint arXiv:2104.07276*, 2021.

[21] G. Lee, B. Hou, A. Mandalika, J. Lee, S. Choudhury, and S. S. Srinivasa, "Bayesian policy optimization for model uncertainty," *arXiv preprint arXiv:1810.01014*, 2018.

[22] S. Särkkä and L. Svensson, *Bayesian filtering and smoothing*. Cambridge university press, 2023, vol. 17.

[23] T. Wongpiromsarn and E. Frazzoli, "Control of probabilistic systems under dynamic, partially known environments with temporal logic specifications," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, 2012, pp. 7644–7651.

[24] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[25] C. Hensel, S. Junges, J.-P. Katoen, T. Quatmann, and M. Volk, "The probabilistic model checker storm," *International Journal on Software Tools for Technology Transfer*, pp. 1–22, 2021.

[26] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: Verification of probabilistic real-time systems," in *Proc. 23rd International Conference on Computer Aided Verification (CAV'11)*, ser. LNCS, G. Gopalakrishnan and S. Qadeer, Eds., vol. 6806. Springer, 2011, pp. 585–591.