Trust-Aware Motion Planning for Human-Robot Collaboration under Distribution Temporal Logic Specifications

Pian Yu[1], Shuyang Dong[2], Shili Sheng[2], Lu Feng[2], and Marta Kwiatkowska[1]

Abstract— Recent work has considered trust-aware decision making for human-robot collaboration (HRC) with a focus on model learning. In this paper, we are interested in enabling the HRC system to complete complex tasks specified using temporal logic formulas that involve human trust. Since accurately observing human trust in robots is challenging, we adopt the widely used partially observable Markov decision process (POMDP) framework for modelling the interactions between humans and robots. To specify the desired behaviour, we propose to use syntactically co-safe linear distribution temporal logic (scLDTL), a logic that is defined over predicates of states as well as belief states of partially observable systems. The incorporation of belief predicates in scLDTL enhances its expressiveness while simultaneously introducing added complexity. This also presents a new challenge as the belief predicates must be evaluated over the continuous (infinite) belief space. To address this challenge, we present an algorithm for solving the optimal policy synthesis problem. First, we enhance the belief MDP (derived by reformulating the POMDP) with a probabilistic labelling function. Then a product belief MDP is constructed between the probabilistically labelled belief MDP and the automaton translation of the scLDTL formula. Finally, we show that the optimal policy can be obtained by leveraging existing point-based value iteration algorithms with essential modifications. Human subject experiments with 21 participants on a driving simulator demonstrate the effectiveness of the proposed approach.

I. INTRODUCTION

Autonomous robots are rapidly evolving into an essential component of our society, to mention home assistive robots [1] and automated vehicles (AV) [2]. Despite significant advances made in automation in recent years, attaining full autonomy, which would enable robots to successfully deal with complicated and unpredictable events or situations, remains stubbornly out of reach. For instance, the AV industry has had to reset expectations as it shifts its focus from level 5 to 4 autonomy [3]. In many applications where robots work with or alongside humans, it is customary to have robots that are operated or supervised by a human [4]. Collaborative human-robot partnerships often hinge on the foundation of trust. Therefore, recognizing human trust and incorporating it into the decision-making process is essential for achieving the full potential of human-robot interactive systems.

[1]Pian Yu and Marta Kwiatkowska are with the Department of Computer Science, University of Oxford, Oxford, United Kingdom {pian.yu, marta.kwiatkowska}@cs.ox.ac.uk

[2]Shuyang Dong, Shili Sheng, and Lu Feng are with School of Engineering, University of Virginia, United States {sd3mn, ss7dr, lu.feng}@virginia.edu

The subject of trust has been actively studied in multiple contexts such as psychology [5] and automation [6]. It is a multifaceted concept that can be influenced by a large number of factors. In the context of human-robot collaboration (HRC), studies have shown that the level of human trust in robots evolves over their interaction, affected by factors such as the automation's reliability, predictability, and transparency [7]. While earlier work has focused on studying the measurement [8], modelling [9], and calibration [10] of human trust in robots, recent work has gravitated towards devising strategies that enable robots to proactively infer and influence the human collaborator's trust [11], [12].

Various methods exist for modeling the interaction between humans and robots. Among these, the game-theoretic approaches [13], [14] and the partially observable Markov decision process (POMDP) framework [15], [16] have been extensively explored. Since trust is not fully observable, in this work we adopt the POMDP, where human trust can be modelled as a hidden variable. While the POMDP formulation allows the robot to act according to its beliefs about the human collaborator's trust based on observations, finding solutions to POMDPs of a realistic size is computationally challenging and existing work often relies on approximation algorithms [17]–[19]. Due to the inherent complexity of solving a POMDP, prior work devoted to trust-based decision making for HRC often focused on relatively simple specifications (e.g., accumulated reward maximisation) [20], [21]. Moreover, in all these studies, trust was treated as an implicit factor that impacts the performance of collaboration. None of these works has considered trust as part of the specification, where explicit requirements can be imposed. Real-world case studies have shown that an inappropriate level of trust may result in the misuse or disuse of automation [6]. Therefore, in practical scenarios, it might be advantageous to stipulate conditions such as "the trust level must not fall below a certain threshold" and "the trust level at a particular juncture must surpass a certain threshold".

Recently, there has been a growing interest in using Linear Temporal Logic (LTL) [22] or its finite variant, syntactically co-safe LTL (scLTL) [23], [24], for specifying complex behaviours of partially observable systems. In [25], it was demonstrated that, for Gaussian linear-time invariant (LTI) POMDPs, a finite-state abstraction can be constructed. This abstraction allows for policy synthesis, which can then be refined to the original Gaussian LTI POMDPs. However, we note that this approach is not applicable to general POMDPs. In [26], [27], policy synthesis for POMDPs under LTL specifications was investigated. It is worth noting that,

in these studies, LTL was employed to define specifications over the state space of the POMDPs, rather than beliefs.

In this work, we investigate trust-aware motion planning for HRC with complex temporal logic specifications applied to both the state of the robot and the trust (belief) of human. The trust-based human-robot interaction is modelled by a trust POMDP and syntactically co-safe linear distribution temporal logic (scLDTL) [28] is utilised to specify the desired behaviour of the system. In [28], scLDTL was introduced as an extension of scLTL to leverage the richness of information contained within belief states of partially observable systems. It was shown that scLDTL is capable of expressing properties involving uncertainty and likelihood that cannot be described by existing logic. Nevertheless, the increased complexity introduced by the inclusion of belief predicates in scLDTL, which must be evaluated over the continuous (infinite) belief space, renders verification and synthesis from scLDTL a more demanding task. In [28], a feasibility checking algorithm was proposed for POMDPs with scLDTL specifications. However, to the best of our knowledge, the more challenging synthesis problem remains unresolved. Our contributions are summarised as follows. (i) We demonstrate the suitability of scLDTL for specifying the desired behaviour of trust-aware HRC systems that involve requirements in the robot workspace as well as the trust (belief) space. (ii) We propose an efficient algorithm to solve the scLDTL optimal policy synthesis for trust POMDPs, which overcomes the aforementioned complexity of scLDTL specifications. (iii) We design and conduct human subject experiments with 21 participants on a driving simulator to evaluate the proposed approach, with encouraging results.

II. MOTIVATING EXAMPLE

We describe a route planning problem for AVs. A human is driving an AV in a town, whose map is shown in Fig. 1(a). Within the town, we consider that there are 3 types of typical incidents that may occur on the road: (1) a pedestrian crossing the road, (2) an obstacle (e.g., a broken bicycle) ahead of the lane, and (3) an oncoming truck in the neighbouring lane. For simplicity, here we assume that there is at most one incident at a time for each road segment.

Fig. 1(b) shows a schematic view of the AV traveling from one location to another. Imagine the AV is approaching an incident on a road segment while in autopilot mode. For safety considerations, the driver might choose to take over control of the AV and switch to manual driving. The level of trust that the driver has in the AV's ability to handle various types of incidents can influence their takeover decision; a driver with lower trust is more inclined to do so. Furthermore, the driver's level of trust changes over time and depends on the takeover decision and the vehicle's ability to handle an incident. In our previous work [21], human subject experiments have shown that, by proactively inferring human trust and taking it into account during decision making, the AV can achieve higher cumulative rewards.

The research focus of [21] was on optimal route planning (e.g., navigating from one location to the other) for AVs.
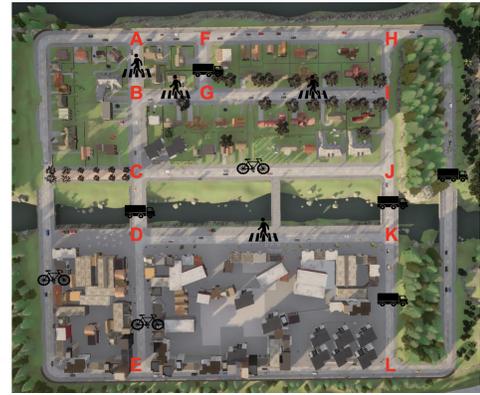

(a)
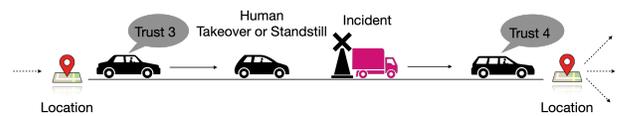
(b)

Fig. 1: Route planning for AVs. (a) A map with three types of road incidents (pedestrian, obstacle, and oncoming truck). (b) A schematic view of the decision making process.

In contrast, in this work we are interested in trust-aware HRC in a broader context. Our goal is to develop a trust-aware motion planning approach for HRC systems, which is capable of completing complex tasks specified in temporal logic that involve requirements on human trust levels.

III. PRELIMINARIES

Before formulating our problem, we provide preliminary background on POMDPs and scLDTL.

A. Partially observable Markov decision processes

This section introduces POMDPs, which are well suited to the modelling of HRC systems under investigation. The human internal states (e.g., trust), which are not fully observable to robots, can be modelled as hidden states in POMDPs. In order to accurately represent the interactions between humans and robots, modifications to the conventional definition of a POMDP [29] are incorporated.

Definition 1 (POMDPs). *A POMDP is defined as a tuple $\mathcal{M} = (S, A, O, \mathcal{Z}, \mathcal{T})$, where $S, A,$ and O are finite sets of states, actions, and observations, respectively, and*

- *$\mathcal{Z} : S \times A \times O \to [0, 1]$ is the probabilistic observation function, which gives the probability of observing o after taking action a in state s, i.e., $\mathcal{Z}(s, a, o) = p(o|s, a)$;*
- *$\mathcal{T} : S \times A \times O \times S \to [0, 1]$ is the probabilistic transition function, which gives the probability that the state has value s' after taking action a and receiving observations o in state s, i.e., $\mathcal{T}(s, a, o, s') = p(s'|s, a, o)$.*

Firstly, we note that, in Definition 1, the probability of receiving observation $o \in O$ is determined by the previous state s (instead of the resulting state s') and the action a

that was just taken. Secondly, the transition function $\mathcal{T}$ is dependent on the observations. For the purpose of this work, a reward function is redundant and has been omitted.

Since a POMDP state is partially observable, we rely on the concept of a *belief state*[1]. Let $\mathcal{B}$ be the belief space of S. A POMDP policy $\pi : \mathcal{B} \to A$ maps a belief state $b \in \mathcal{B}$, which is a probability distribution over S, to a prescribed action $a \in A$. Given a policy π, the control of the agent's actions is performed online. First, the agent takes an action $a = \pi(b)$ according to the given policy π and the current belief is b. Second, after taking an action a and receiving an observation o, the agent updates its belief:

$$b'(s') = \sum_{s \in S} b(s) \sum_{o \in \mathbb{F}_O(s,a)} \mathcal{Z}(s,a,o) \cdot \mathcal{T}(s,a,o,s'), \quad (1)$$

where $\mathbb{F}_O(s,a) = \{o \in O | \mathcal{Z}(s,a,o) > 0\}$. The process then repeats. An interesting property to note about the POMDP described in Definition 1 is that the belief update (1) is linear[2]. An *execution* $\boldsymbol{\rho}$ of a POMDP is a possibly infinite alternating sequence of belief states, actions, and observations, i.e., $\boldsymbol{\rho} = b_0 a_0 o_0 b_1 a_1 o_1 \cdots$.

B. Syntactically Co-Safe Linear Distribution Temporal Logic

This section introduces scLDTL for concisely specifying the desired behaviour of the HRC systems. It will become clear later that scLDTL is capable of expressing requirements in both the robot workspace and the trust (belief) space.

scLDTL consists of two types of predicates: (i) state predicates ν that are evaluated over the states and (ii) belief predicates μ, which are obtained after evaluation of a predicate function $g_\mu : \mathcal{B} \to \mathbb{R}$ on the belief space $\mathcal{B}$ as: $\mu = \top$ if $g_\mu(b) < 0$ and $\mu = \bot$ if $g_\mu(b) \geq 0$.

An scLDTL formula is defined inductively according to the following syntax [28]:

$$\varphi ::= \top | \nu | \mu | \neg\nu | \neg\mu | \varphi_1 \wedge \varphi_2 | \varphi_1 \vee \varphi_2 | \varphi_1 \mathsf{U} \varphi_2 | \bigcirc \varphi | \Diamond \varphi, \quad (2)$$

where ν is a state predicate, μ is a belief predicate, $\neg$ (negation), $\wedge$ (conjunction), and $\vee$ (disjunction) are logic connectives, and U (until), $\bigcirc$ (next) and $\Diamond$ (eventually) are temporal operators. We omit the scLDTL semantics due to page limit and refer the reader to [28].

Let AP be a set of state predicates and BP be a set of belief predicates. The satisfaction of an scLDTL formula φ over $\mathrm{AP} \cup \mathrm{BP}$ can be captured through a deterministic finite automaton (DFA) $\mathcal{A} = (Q, q_0, 2^{\mathrm{AP} \cup \mathrm{BP}}, \delta, \mathrm{Acc})$, where Q is a finite set of states, $q_0 \in Q$ is the initial state, $\delta : Q \times 2^{\mathrm{AP} \cup \mathrm{BP}} \to Q$ is the transition function, and $\mathrm{Acc} \subseteq Q$ is the set of accepting states. A finite *run* $q = q_0 q_1 \ldots q_k$ of $\mathcal{A}$ is called *accepting* if $q_k \in \mathrm{Acc}$. Next we define the notion of probabilistic satisfaction with respect to an execution $\boldsymbol{\rho}$ of a POMDP.

---

[1] A belief state is a probability distribution over all possible states in the POMDP. It represents the agent's subjective probability distribution of being in each state given its past observations and actions.

[2] The belief update of a conventional POMDP is often represented using the Bayes' filter.

Definition 2. *[scLDTL satisfaction with respect to a POMDP execution] Given an execution* $\boldsymbol{\rho} = b_0 a_0 o_0 b_1 a_1 o_1 \ldots$ *of a POMDP* $\mathcal{M}$*, the probability that the execution* $\boldsymbol{\rho}$ *satisfies the scLDTL formula* φ *is given by*

$$\mathrm{Pr}_\mathcal{M}(\{s_0 s_1 \cdots \text{ such that } (s_0, b_0)(s_1, b_1) \cdots \models \varphi\} \mid \boldsymbol{\rho}).$$

For simplicity, it is denoted in shorthand as $\mathrm{Pr}_\mathcal{M}(\varphi \mid \boldsymbol{\rho})$.

IV. POMDPs FOR HRC

In this work, we consider a human and a robot working collaboratively in the workspace X. The human (H) adopts a supervisory role and the robot (R) is charged with performing tasks. The human can intervene in the task execution due to, for instance, low trust.

A. HRC modelling

Within the workspace X, one can identify a set of incidents I, i.e., a set of events that can affect human trust and/or takeover decision. The likelihood of observing an incident $id \in I$ is determined by the current state and the action of the robot. Denote by Θ the state space of human trust in the robot, which is not fully observable by the robot.

The human-robot interaction can be modelled as a POMDP $\mathcal{M}$ as per Definition 1, where the state space S of $\mathcal{M}$ is factored as the observable state space X and the non-observable state space Θ, i.e., $S = X \times \Theta$. Accordingly, the probabilistic transition function $\mathcal{T}$ is factored as the world state and the human trust probabilistic transition functions $\mathcal{T}_X$ and $\mathcal{T}_\Theta$, respectively. It has been shown in [20] that the human trust affects human behaviour (e.g., takeover decision) and the human trust is affected by factors such as the robot performance (i.e., success/fail in handling an incident). Therefore, the state evolution of the trust POMDP $\mathcal{M}$ is determined not only by the robot action a^r, but also posterior observations, including the incident id encountered during execution, the human takeover decision a^h, and the robot performance e^r. Formally, the trust POMDP is specified as a tuple $\mathcal{M} = (X, \Theta, A^r, O, \mathcal{Z}, \mathcal{T}_X, \mathcal{T}_\Theta)$, where

- A^r is the finite action space of the robot;
- $O = I \times A^h \times \mathcal{E}^r$ is the observation set, where
  - I is the set of incidents;
  - $A^h = \{\mathrm{tk}, \mathrm{st}\}$ is the action space of the human, where tk and st stand for "takeover" and "standstill", respectively; similarly to [20], we assume that the human first observes the robot's action a^r and then decides his or her own action a^h;
  - $\mathcal{E}^r = \{\mathrm{succ}, \mathrm{fail}\}$ represents the performance of the robot, where succ and fail stand for "success" and "failure", respectively.
- $\mathcal{Z} : X \times \Theta \times A^r \times O \to [0,1]$ is the probabilistic observation function, which is given by $\mathcal{Z}(x, \theta, a^r, o) = p(id, a^h, e^r | x, \theta, a^r)$;
- $\mathcal{T}_X : X \times A^r \times O \times X \to [0,1]$ is the world state probabilistic transition function, i.e., $\mathcal{T}_X(x, a^r, o, x') = p(x'|x, a^r, o)$;
- $\mathcal{T}_\Theta : \Theta \times O \times \Theta \to [0,1]$ is the human trust probabilistic transition function, i.e., $\mathcal{T}_\Theta(\theta, o, \theta') = p(\theta'|\theta, o)$.
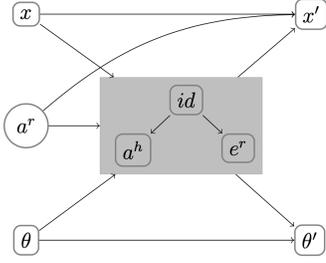
Fig. 2: The graphical model for the trust POMDP $\mathcal{M}$.

A graphical model of the trust POMDP is shown in Fig. 1. It contains two key components: (i) a trust dynamics model, which captures the evolution of human trust in the robot, and (ii) a human decision model, which connects trust with human actions.

Let $\mathcal{B}_\Theta$ be the belief space of Θ. We associate with each value $x \in X$ a belief space for θ: $\mathcal{B}_\Theta(x) := \{(x, b_\Theta) | b_\Theta \in \mathcal{B}_\Theta\}$. Then the belief space of the trust POMDP $\mathcal{M}$ can be defined as $\mathcal{B} = \cup_{x \in X} \mathcal{B}_\Theta(x)$. The POMDP policy π : $\mathcal{B} \to A^r$ maps a belief $b \in \mathcal{B}$ to a prescribed robot action $a^r \in A^r(b)$, where $A^r(b)$ is the set of actions available at b.

We illustrate our results with the motivating example.

Example (continued). *The town map shown in Fig. 1(a) has 12 road intersections $\{A, \cdots, L\}$. Depending on the driving direction, each intersection can be factored into 3 different states (for instance, intersection A contains states EA, BA, and FA). We use Muir's questionnaire [30] with a 7-point Likert scale as a human trust metric (i.e., trust ranges from 1 to 7). Therefore, one has that $X = \{EA, BA, FA \cdots, EL, KL, HL\}$, $\Theta = \{1, \cdots, 7\}$, and $I = \{\text{'pedestrian'}, \text{'obstacle'}, \text{'truck'}\}$. The robot action is route choices and one can define an indicator function $\mathcal{I}$ for incidents. For instance, $\mathcal{I}(EA, AB, \text{'pedestrian'}) = 1$.*

B. Problem formulation

In this work, we consider that the HRC system is required to complete complex tasks in the robot workspace X. Moreover, there are requirements on the human trust Θ and/or trust belief b_Θ. We formulate the tasks in the workspace as well as the requirements on human trust and trust belief in the form of an scLDTL formula φ.

Example (continued). *Consider now that the AV is required to complete the following tasks: (i) visits the target intersections G, J and L (in this order) from the initial location EA (see Fig. 1(a)), (ii) the human trust level cannot be too low (lower than 2) at all times, and (iii) when the vehicle reaches the final intersection L, the likelihood that the human trust level is high (higher than or equal to 6) is no less than 0.5. In scLDTL, this specification can be written as*

$$\varphi = (\neg\text{LOWTRUST})\text{U}(\Diamond(\nu_1 \wedge \Diamond(\nu_2 \wedge \Diamond(\nu_3 \wedge \text{HIGHTRUST})))),$$

where there are 3 state predicates $\nu_1 = \{BG, FG, IG\}$, $\nu_2 = \{IJ, CJ, KJ\}$, $\nu_3 = \{KL, EL, HL\}$ and 2 belief predicates LOWTRUST, HIGHTRUST. The predicate functions are given by $g_{\text{LOWTRUST}} = 1 - A_1 b_\Theta$, $g_{\text{HIGHTRUST}} = 0.5 - A_2 b_\Theta$, where

$A_1 = [1, 0, 0, 0, 0, 0, 0]$ *encodes trust lower than 2 and* $A_2 = [0, 0, 0, 0, 0, 1, 1]$ *encodes trust higher than or equal to 6.*

Given the initial belief state b_0 and a policy π, denote by $\boldsymbol{\rho}^\pi(b_0)$ the set of all possible executions generated by π. We consider the optimal policy synthesis problem for HRC under scLDTL specifications, i.e., find an optimal policy π such that the probability of the set of all executions that satisfy an scLDTL formula φ under π is maximised. Mathematically, this problem can be formulated as follows.

Problem 1. *Given the trust POMDP $\mathcal{M}$ and the scLDTL specification φ, find a policy $\pi \in \Pi$ such that*

$$\max_{\pi \in \Pi} \quad \text{Pr}^\pi_\mathcal{M}(\varphi) \triangleq \sum_{\boldsymbol{\rho} \in \boldsymbol{\rho}^\pi(b_0)} \text{Pr}(\boldsymbol{\rho}|b_0, \pi)\text{Pr}_\mathcal{M}(\varphi \mid \boldsymbol{\rho}), (3)$$

where Π is the set of all policies for the trust POMDP $\mathcal{M}$ and $\text{Pr}_\mathcal{M}(\varphi \mid \boldsymbol{\rho})$ is given in Definition 2.

Remark 1. *Point-based value iteration (PBVI) algorithm has been proposed for POMDPs under LTL specifications. In [27], the atomic propositions of an LTL formula are evaluated on the state space of the POMDP, which is finite. Therefore, the construction of the product POMDP and the computation of maximal end components are similar to finite MDPs, for which the existing graph-based methods [22] can be utilised. In this work we consider scLDTL specifications, in which the belief predicates are evaluated over the belief space $\mathcal{B}$ of the trust POMDP $\mathcal{M}$, which is infinite. Therefore, the approach proposed in [27] is not applicable here.*

V. PROPOSED APPROACH

This section presents our approach to solve the optimal policy synthesis problem (Problem 1), which falls outside the purview of existing policy synthesis algorithms designed for POMDPs, e.g., [25]–[27]. It is divided into two parts: (1) the construction of the product POMDP with the DFA of the scLDTL formula φ and (2) an algorithm to approximately compute a policy that maximises the probability of satisfying the given scLDTL formula φ.

A. Product POMDP

To begin with, we define the deterministic state and belief predicate labelling functions L_s and L_b as

- $L_s : X \times \Theta \to 2^{\text{AP}}$, which contains the set of state predicates that can be true at state (x, θ);
- $L_b : X \times \mathcal{B}_\Theta \to 2^{\text{BP}}$, which contains the set of belief predicates that can be true at belief state (x, b_Θ).

Then the corresponding values are given by $p_{L_s} : X \times \Theta \times 2^{\text{AP}} \to \{0, 1\}$ and $p_{L_b} : X \times \mathcal{B}_\Theta \times 2^{\text{BP}} \to \{0, 1\}$, respectively.

Now we propose to reformulate the trust POMDP $\mathcal{M}$ (equivalently) as a belief MDP and further expand it by including probabilistic labels, which yields a *probabilistically labelled* belief MDP $\hat{\mathcal{M}} = (\mathcal{B}, A^r, O, \hat{\mathcal{Z}}, \hat{\mathcal{T}}_\mathcal{B}, \hat{L}, p_{\hat{L}})$, where $\mathcal{B}$ is the belief space, A^r, O are given in $\mathcal{M}$, and

- $\hat{\mathcal{Z}} : \mathcal{B} \times A^r \times O \to [0, 1]$ is the probabilistic observation function;

- $\hat{\mathcal{T}}_\mathcal{B} = (\mathcal{T}_X, \mathcal{T}_{\mathcal{B}_\Theta})$ is the probabilistic transition function, where $\mathcal{T}_X$ is defined in $\mathcal{M}$ and $\mathcal{T}_{\mathcal{B}_\Theta} : \mathcal{B}_\Theta \times O \times \mathcal{B}_\Theta \to [0,1]$;
- $\hat{L} : \mathcal{B} \to 2^{2^{\mathrm{AP} \cup \mathrm{BP}}}$ is the belief state labelling function, where $\hat{L}(x, b_\Theta)$ contains the set of state and belief predicate subsets that can be true at (x, b_Θ);
- $p_{\hat{L}} : \mathcal{B} \times 2^{\mathrm{AP} \cup \mathrm{BP}} \to [0,1]$ specifies the associated probability.

The probabilistic labelling function $p_{\hat{L}}$ provides a unified way of assigning belief states with both state and belief predicates in an scLDTL formula. For instance, given a belief state (x, b_Θ), a state predicate ν, and a belief predicate μ, one has that $p_{\hat{L}}(x, b_\Theta, \nu) = b_\Theta(\theta) p_{L_s}(x, \theta, \nu), \forall \theta \in \Theta$ and $p_{\hat{L}}(x, b_\Theta, \mu) = p_{L_b}(x, b_\Theta, \mu)$.

The product belief MDP $\mathcal{M}^\times$ is constructed between the probabilistically labelled belief MDP $\hat{\mathcal{M}}$ and the DFA $\mathcal{A} = (Q, q_0, 2^{\mathrm{AP} \cup \mathrm{BP}}, \delta, \mathrm{Acc})$ of the scLDTL formula φ.

Definition 3 (Product belief MDP). *Denote by $\mathcal{M}^\times$ the product $\hat{\mathcal{M}} \times \mathcal{A}$ as a tuple $\mathcal{M}^\times = (S^\times, A^r, O, \mathcal{Z}^\times, \mathcal{T}^\times, \mathrm{Acc}^\times)$, where*

$$S^\times = \mathcal{B} \times 2^{\mathrm{AP} \times \mathrm{BP}} \times Q$$

is so that $(x, b_\Theta, l, q) \in S^\times, \forall (x, b_\Theta) \in \mathcal{B}, \forall l \in \hat{L}(x, b_\Theta)$, and $\forall q \in Q$, $\mathcal{Z}^\times : S^\times \times A^r \times O \to [0,1]$ is the probabilistic observation function, and $\mathrm{Acc}^\times = \{(x, b_\Theta, l, q) \in S^\times | q \in \mathrm{Acc}\}$. The probabilistic transition function $\mathcal{T}^\times : S^\times \times A^r \times O \times S^\times \to [0,1]$ is defined as

$$\mathcal{T}^\times((x, b_\Theta, l, q), a^r, o, (x', b'_\Theta, l', q')) =$$
$$\begin{cases} \hat{\mathcal{T}}_\mathcal{B}((x, b_\Theta), a^r, o, (x', b'_\Theta)) \cdot p_{\hat{L}}(x', b'_\Theta, l'), & \text{if } q' \in \delta(q, l), \\ 0, & \text{otherwise;} \end{cases}$$

where $\hat{\mathcal{Z}}, \hat{\mathcal{T}}_\mathcal{B}$, and $p_{\hat{L}}$ are defined in $\hat{\mathcal{M}}$.

Let $\Pi^\times$ be the set of all policies for the product belief MDP $\mathcal{M}^\times$. The set of *accepting states* of $\mathcal{M}^\times$ is given by $\mathrm{Acc}^\times$. We have the following result.

Theorem 1. *Given the trust POMDP $\mathcal{M}$ and the scLDTL formula φ, the maximal probability of satisfying φ is:*

$$\max_{\pi \in \Pi}\{\mathrm{Pr}_\mathcal{M}^\pi(\varphi)\} = \max_{\hat{\pi} \in \Pi^\times}\{\mathrm{Pr}_{\mathcal{M}^\times}^{\hat{\pi}}(\Diamond \mathrm{Acc}^\times)\}.$$

Theorem 1 shows that, with the set of accepting states $\mathrm{Acc}^\times$, the original optimal policy synthesis problem reduces to a reachability problem.

B. Optimal policy synthesis

PBVI algorithms have been widely used for solving POMDP synthesis problems [17]–[19]. They often offer convergence guarantees specified as upper and lower bounds on the value function. However, these PBVI algorithms are not directly applicable for solving our problem (Problem 1). This is because solving a reachability problem for POMDPs necessitates the presence of a clearly defined reward function, which assigns value 1 to states in the goal set and 0 otherwise. However, in our case, capturing the satisfaction of an scLDTL specification through a state-based reward function is not feasible due to the presence of belief predicates.

In the following, we show that, with essential modifications, the existing PBVI algorithms can be leveraged for solving Problem 1 with the set of accepting states $\mathrm{Acc}^\times$.

Given a state s of the product belief MDP $\mathcal{M}^\times$, we first define a value function $V : S^\times \to \mathbb{R}_{\geq 0}$ as

$$V(s) = \max_{\pi \in \Pi^\times}\{\mathrm{Pr}_{\mathcal{M}^\times}^\pi(\Diamond \mathrm{Acc}^\times)\},$$

which represents the maximal probability of reaching $\mathrm{Acc}^\times$ from initial state s. Then one can get that $V(s) = 1, \forall s \in \mathrm{Acc}^\times$. For $s \notin \mathrm{Acc}^\times$, we further define the dynamic programming operator T as

$$T(V)(s) = \max_{a^r \in A^r} \Big\{ \sum_{o \in \mathbb{F}_O(s, a^r)} \mathcal{Z}^\times(s, a^r, o) \sum_{s' \in S^\times} \mathcal{T}^\times(s, a^r, o, s') V(s') \Big\}.$$

Before running a PBVI algorithm, first we initialize the upper- and lower-bounds of the value function V as follows:

$$\overline{V}^0(s) = 1, \forall s \in S^\times, \quad \underline{V}^0(s) = \begin{cases} 1, & \text{if } s \in \mathrm{Acc}^\times, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

Then a precision parameter τ is provided that controls the tightness of the convergence (for example, by controlling the depth of the tree in SARSOP [19]), which yields $|\overline{V}(s_0) - \underline{V}(s_0)| \leq \tau$, where $s_0 = (b_0, q_0)$ is the initial state of the product belief MDP $\mathcal{M}^\times$.

Denote by $\mathrm{Pr}_\mathcal{M}^{\max}(\varphi) := \max_{\pi \in \Pi}\{\mathrm{Pr}_\mathcal{M}^\pi(\varphi)\}$ the maximal probability of satisfying the scLDTL formula φ. We have the following result.

Theorem 2. *Let $\overline{V}(s_0)$ and $\underline{V}(s_0)$ be the upper- and lower-bounds of $V(s_0)$ obtained using PBVI with the initialization function (4). One has that*

$$\underline{V}(s_0) \leq \mathrm{Pr}_\mathcal{M}^{\max}(\varphi) \leq \overline{V}(s_0).$$

Finally, the optimal policy $\hat{\pi}^*$ for state $s \in S^\times$ can be derived using the value function.

VI. IMPLEMENTATION

We have implemented the proposed approach to obtain the optimal policy for each scLDTL specification under consideration for the motivating example. To construct the trust POMDP, we utilise the trust dynamics model and the human takeover decision model, which were derived through an online user study involving 100 anonymous participants on Amazon Mechanical Turk (AMT) platform [21]. Then two scLDTL specifications $\varphi_1 = (\neg\mathrm{LOWTRUST})\mathsf{U}(\Diamond(\nu_1 \wedge \Diamond(\nu_2 \wedge \Diamond(\nu_3 \wedge \mathrm{HIGHTRUST}))))$ and $\varphi_2 = (\neg\mathrm{LOWTRUST})\mathsf{U}(\Diamond(\nu_4 \wedge \Diamond(\nu_5 \wedge \Diamond(\nu_6 \wedge \mathrm{HIGHTRUST}))))$ are considered for the AV, where state predicates ν_1, ν_2, ν_3 are defined in the Example, $\nu_4 = \{\mathrm{AB}, \mathrm{GB}, \mathrm{CB}\}, \nu_5 = \{\mathrm{DK}, \mathrm{JK}, \mathrm{LK}\}, \nu_6 = \{\mathrm{AE}, \mathrm{DE}, \mathrm{LE}\}$, and belief predicates LOWTRUST and HIGHTRUST are evaluated over the predicate function $g_{\mathrm{LOWTRUST}} = 1 - A_1 b_\Theta$ and $g_{\mathrm{HIGHTRUST}} = 0.5 - A_2 b_\Theta$. The DFA for each scLDTL specification $\varphi_i, i \in \{1, 2\}$ is derived using [31]. Then the corresponding product belief

MDP $\mathcal{M}_i^{\times}$ is constructed with the DFA of φ_i. Finally, the upper- and lower-bounds of the value function V_i are computed using the POMDP toolkit "pomdp_py" [32] (with essential modifications described in Section V.B). The precision parameter is set as $\tau = 0.01$. All simulations are carried out on a Macbook Pro (2.6 GHz 6-Core Intel Core i7 and 16 GB of RAM).

VII. DRIVING SIMULATOR STUDY

We evaluate the effectiveness of obtained policies via a driving simulator study.[3]

Study design. The study was conducted in a fixed-based driving simulator from SimXperience, consisting of a 55-inch display, a racing car seat, a Logitech G29 steering wheel, and sport pedals, see Fig. 3. In our study, four buttons on the steering wheel were programmed to let the drivers increase/decrease trust, switch driving mode between manual and autopilot, switch gear between drive and reverse.

The experiments were run on a machine with 3.5GHz CPU, NVIDIA GeForce RTX 3080 Ti, 62GB memory, and Ubuntu 20.04.6 LTS operating system. The virtual driving environment was created using CARLA 0.9.13. An autopilot controller was programmed for driving tasks such as lane keeping, taking turns at intersections, and handling incidents.

We recruited 21 undergraduate students from the university community to participant. All participants had a valid driver's license and regular or corrected-to-normal vision. Each participant was compensated with a \$10 gift card. We adopted a within-subject study design: each participant took 4 unique routes, i.e., trust-aware and trust-free routes for both scLDTL specifications. The start and destination is either from A to L (φ_1) or from H to E (φ_2). The route is either the trust-aware route (obtained using the computed optimal policy) or the trust-free route (which is the shortest-distance route from the start to the destination). However, if the vehicle has not reached the destination after travelling 20 intersections, it reschedules a shortest-distance route. The order of trials is randomized and counter-balanced.

Study Procedure. Upon arrival, a participant was instructed to read and sign a consent form approved by the Institutional Review Board. We conducted a five-minute training session to familiarize the participant with the driving simulator setup.

The vehicle started driving in autopilot mode. When the vehicle approached an incident, the participants can decide whether to take over the vehicle to handle the incidents on the road. Should the participant choose not to take over, the vehicle will remain in autopilot mode to handle the incident. At any point during the experiment, the participant has the option to assume control of the vehicle and switch to manual driving. Should the participant choose to take over, he/she was required to switch back to autopilot mode before arriving at the next intersection so that the vehicle can choose the next direction to go. We asked the participants to periodically record their trust in the AV using the buttons

Fig. 3: Driving simulator setup.

on the steering wheel. It took about 40 minutes for each participant to complete the entire experiment.

Results. The computed maximal probabilities for satisfying the scLDTL specifications φ_1 and φ_2 are $\mathrm{Pr}_{\mathcal{M}}^{\max}(\varphi_1) = 0.8357$ and $\mathrm{Pr}_{\mathcal{M}}^{\max}(\varphi_2) = 0.7288$, respectively. For each participant, we evaluate the satisfaction of each scLDTL specification φ_i using the robot states x and human trust levels Θ recorded in the experiment, and the induced trust beliefs b_Θ. In total, 18 out of 21 ($18/21 = 0.8571 > 0.8357$) and 17 out of 21 ($17/21 = 0.8095 > 0.7288$) participants successfully complete the specifications φ_1 and φ_2 respectively, which validates the effectiveness of the proposed approach. We further compare the trust-aware and trust-free routes for all drivers and both trails. The percentage of scLDTL satisfaction is 0.85 (trust-aware) vs 0.775 (trust-free). The average human trust is 4.6018 (trust-aware) vs 4.3933 (trust-free). One can see that the trust-aware policy outperforms the trust-free one. A video demonstration of the human experiment can be found at: https://www.youtube.com/watch?v=pY0PkxYbQXo.

We summarise three key observations gained from the experiments. First, observing the AV successfully handle the same incidents multiple times does not necessarily guarantee an increase in human trust. Second, if a human's trust remains consistently low for an extended period, it becomes challenging for them to re-establish trust in the AV. Third, having the capability to effectively address an incident beforehand can contribute to boosting human trust. Based on the feedback received after the experiment, participants have indicated that, had they noticed the car braking earlier in situations involving pedestrians, they might have considered the AV more trustworthy.

VIII. CONCLUSIONS

In this work, we presented a trust-aware motion planning approach for HRC. We demonstrated the suitability of scLDTL for describing the desired behaviours of HRC systems and an algorithm was proposed for solving the optimal policy synthesis problem. Human subject experiments were conducted on a driving simulator, validating the effectiveness of the proposed approach and providing valuable new insights. Additionally, we observed variations in trust dynamics among individuals, which will be further investigated in future research.

---

[3]This study was approved by the University of Virginia Institutional Review Boards under IRB-SBS protocol #6045.

REFERENCES

[1] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun, "Towards robotic assistants in nursing homes: Challenges and results," *Robotics and autonomous systems*, vol. 42, no. 3-4, pp. 271–281, 2003.

[2] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 187–210, 2018.

[3] M. Anderson, "Surprise! 2020 is not the year for self-driving cars," *IEEE Spectrum*, 2020.

[4] X. Zhou, A. Schmedding, H. Ren, L. Yang, P. Schowitz, E. Smirni, and H. Alemzadeh, "Strategic safety-critical attacks against an advanced driver assistance system," in *2022 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2022, pp. 79–87.

[5] D. Dunning and D. Fetchenhauer, "Understanding the psychology of trust," in *Social motivation*. Psychology Press, 2011, pp. 147–169.

[6] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," *Human factors*, vol. 46, no. 1, pp. 50–80, 2004.

[7] K. E. Schaefer, J. Y. Chen, J. L. Szalma, and P. A. Hancock, "A meta-analysis of factors influencing the development of trust in automation: Implications for understanding autonomy in future systems," *Human factors*, vol. 58, no. 3, pp. 377–400, 2016.

[8] A. Freedy, E. DeVisser, G. Weltman, and N. Coeyman, "Measurement of trust in human-robot collaboration," in *2007 International symposium on collaborative technologies and systems*. Ieee, 2007, pp. 106–114.

[9] A. Xu and G. Dudek, "Optimo: Online probabilistic trust inference model for asymmetric human-robot collaborations," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, 2015, pp. 221–228.

[10] N. Wang, D. V. Pynadath, and S. G. Hill, "Trust calibration within a human-robot team: Comparing automatically generated explanations," in *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2016, pp. 109–116.

[11] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Planning with trust for human-robot collaboration," in *Proceedings of the 2018 ACM/IEEE international conference on human-robot interaction*, 2018, pp. 307–315.

[12] S. Sheng, E. Pakdamanian, K. Han, Z. Wang, J. Lenneman, and L. Feng, "Trust-based route planning for automated vehicles," in *Proceedings of the ACM/IEEE 12th International Conference on Cyber-Physical Systems*, 2021, pp. 1–10.

[13] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions." in *Robotics: Science and systems*, vol. 2. Ann Arbor, MI, USA, 2016, pp. 1–9.

[14] X. Huang, M. Kwiatkowska, and M. Olejnik, "Reasoning about cognitive trust in stochastic multiagent systems," *ACM Transactions on Computational Logic (TOCL)*, vol. 20, no. 4, pp. 1–64, 2019.

[15] F. Broz, I. R. Nourbakhsh, and R. G. Simmons, "Planning for human-robot interaction using time-state aggregated pomdps." in *AAAI*, vol. 8, 2008, pp. 1339–1344.

[16] M. Lauri, D. Hsu, and J. Pajarinen, "Partially observable markov decision processes in robotics: A survey," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 21–40, 2022.

[17] D. Silver and J. Veness, "Monte-carlo planning in large pomdps," *Advances in neural information processing systems*, vol. 23, 2010.

[18] T. Smith and R. Simmons, "Point-based pomdp algorithms: Improved analysis and implementation," *arXiv preprint arXiv:1207.1412*, 2012.

[19] H. Kurniawati, D. Hsu, and W. S. Lee, "Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces." in *Robotics: Science and systems*, vol. 2008. Citeseer, 2008.

[20] M. Chen, S. Nikolaidis, H. Soh, D. Hsu, and S. Srinivasa, "Trust-aware decision making for human-robot collaboration: Model learning and planning," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 9, no. 2, pp. 1–23, 2020.

[21] S. Sheng, E. Pakdamanian, K. Han, Z. Wang, J. Lenneman, D. Parker, and L. Feng, "Planning for automated vehicles with human trust," *ACM Transactions on Cyber-Physical Systems*, 2022.

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

[23] A. M. Ayala, S. B. Andersson, and C. Belta, "Temporal logic motion planning in unknown environments," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 5279–5284.

[24] K. Cho, J. Suh, C. J. Tomlin, and S. Oh, "Cost-aware path planning under co-safe temporal logic specifications," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2308–2315, 2017.

[25] S. Haesaert, P. Nilsson, C. I. Vasile, R. Thakker, A.-a. Agha-mohammadi, A. D. Ames, and R. M. Murray, "Temporal logic control of pomdps via label-based stochastic simulation relations," *IFAC-PapersOnLine*, vol. 51, no. 16, pp. 271–276, 2018.

[26] M. Svoreňová, M. Chmelík, K. Leahy, H. F. Eniser, K. Chatterjee, I. Černá, and C. Belta, "Temporal logic motion planning using pomdps with parity objectives: Case study paper," in *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*, 2015, pp. 233–238.

[27] M. Bouton, J. Tumova, and M. J. Kochenderfer, "Point-based methods for model checking in partially observable markov decision processes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 10 061–10 068.

[28] A. Jones, M. Schwager, and C. Belta, "Distribution temporal logic: Combining correctness with quality of estimation," in *52nd IEEE Conference on Decision and Control*. IEEE, 2013, pp. 4719–4724.

[29] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Partially observable markov decision processes for artificial intelligence," in *KI-95: Advances in Artificial Intelligence: 19th Annual German Conference on Artificial Intelligence Bielefeld, Germany, September 11–13, 1995 Proceedings 19*. Springer, 1995, pp. 1–17.

[30] B. M. Muir, *Operators' Trust in and Use of Automatic Controllers in a Supervisory Process Control Task*. University of Toronto, 1990.

[31] J. Klein, "ltl2dstar-ltl to deterministic streett and rabin automata," 2007.

[32] K. Zheng and S. Tellex, "pomdp_py: A framework to build and solve pomdp problems," in *ICAPS 2020 Workshop on Planning and Robotics (PlanRob)*, 2020.