Robotics: Science and Systems 2023
Daegu, Republic of Korea, July 10-July 14, 2023

Follow my Advice: Assume-Guarantee Approach to Task Planning with Human in the Loop

Georg Friedrich Schuppe*, Ilaria Torre*†, Iolanda Leite* and Jana Tumova*
* KTH Royal Institute of Technology, Sweden
† Chalmers University of Technology

***Abstract*—We focus on correct-by-design robot task planning from finite Linear Temporal Logic (LTL_f) specifications with a human in the loop. Since provable guarantees are difficult to obtain unconditionally, we take an assume-guarantee perspective. Along with guarantees on the robot's task satisfaction, we compute the weakest sufficient assumptions on the human's behavior. We approach the problem via a stochastic game and leverage algorithmic synthesis of the weakest sufficient assumptions. We turn the assumptions into runtime advice to be communicated to the human. We conducted an online user study and showed that the robot is perceived as safer, more intelligent and more compliant with our approach than a robot giving more frequent advice corresponding to stronger assumptions. In addition, we show that our approach leads to less violations of the specification than not communicating with the participant at all.**

I. Introduction

When deploying autonomous robots into the real world, we desire for them to be safe and to work as intended. One increasingly popular way to approach this challenging problem is via the formal methods framework, which combines two great advantages: (i) various involved tasks and constraints on the robot's behavior can be described rigorously in temporal logics, such as Linear Temporal Logic (LTL); and (ii) formal methods-based synthesis allows to algorithmically compute plans that are correct by design, i.e. provably satisfy the logic formula. Formal methods-based synthesis from temporal logic specification has been successfully deployed in a number of robot task planning scenarios [18] ranging from manipulation that is robust against a human operator's cooperative or adversarial interventions [20] to multi-robot task planning with pre-failure warnings that give users insight as to why a specification may be violated in the future [14].

A question that remains largely open is how to guarantee that a robot operates safely and as intended when there is a human in the loop. Attempting to provide guarantees regardless of the human's actions often leaves us with the answer: "There is no correct-by-design plan." For example, imagine an autonomous vehicle following a road when a pedestrian appears and wants to cross the street. If the pedestrian jumps right in front of the vehicle, a collision is unavoidable and the system safety will be violated. However, it might still be possible to synthesize a provably safe control strategy for the autonomous vehicle under a certain assumption, for instance that the pedestrian does not decide to cross the street if the vehicle is too close, that the pedestrian does not cross the street at all, or that the pedestrian commits to crossing the street only at designated space and time window as advised by traffic lights. In other words, it might still be possible to obtain conditional guarantees under the – not unreasonable – assumption that the pedestrian is willing to some extent to collaborate on maintaining safety.

In this work, we propose to take an *assume-guarantee* perspective to correct-by-design task planning with a human in the loop. Along with guarantees on the robot's task satisfaction, we synthesize assumptions on the human's behavior. As the example above shows, desired guarantees can be achieved under a variety of assumptions, so an additional challenge is to synthesize assumptions that will be perceived as *acceptable*. On the technical level, we tackle the problem through formulating the human-robot interaction as a stochastic game. We transform the task planning problem under finite LTL specifications into a stochastic game with a reachability objective, and find the weakest sufficient assumptions [10]. We turn these assumptions into advice communicated at runtime. Loosely speaking, these take form of actions that the human should not take (e.g., "Don't take a tomato now.") and actions that the human is encouraged to take in near future (e.g., "Help me with the ketchup."). In an online user study, we show that communicating the advice leads to less specification violations than not communicating any advice at all. We also show that communicating advice based on the weakest sufficient assumptions is perceived as more safe, more intelligent and more compliant compared to a more frequent advice based on stronger assumptions.

A. Related Work

Task planning for autonomous robots with a human in the loop has been approached from various angles. For instance, Buisan and Alami [7] leverage hierarchical task networks to plan for a task while also considering and emulating the human decision, action, and reaction processes. In formal methods-based task planning, many approaches rely on models of the systems they analyze, including the humans. For example, Feng et al. [13] model a human operator in a UAV control scenario via a Markov Decision Process based on key characteristics such as the operator's workload, proficiency, and fatigue. Junges et al. [17] use probabilistic human models obtained through reinforcement learning on previously recorded observations of human trajectories. However, explicitly modelling humans is usually not very robust to small changes in the context. Many assumptions, such as that the humans behave rationally also do not hold in real-world experiments [8].

In contrast, some other works that focus on formal methods-based task planning with a human in the loop do not model the human explicitly, but introduce reactive re-planning, such as in [20]. Migimatsu and Bohg [22] propose a object-cenric reactive task-planning method that produces sequential manipulation plans relative to object positions that remain valid even if the objects are moved, for example by a human. He et al. [15] synthesize strategies reacting to the human behaviour. They propose an efficient, compositional approach to perform synthesis for finite-horizon tasks.

Attempting to create plans that are robust to all possible human behavior in the environment is, however, often too conservative. In many situations, formal synthesis fails to find a correct-by-design task plan e.g., due to high uncertainty in the environment model, or infeasibility of the task. Solutions to this challenge include refining the model, i.e. adding assumptions on the robot or its environment, repairing the specification itself, or finding plans that violate the specification as little as possible. For instance, Raman and Kress-Gazit [23] automatically provide feedback through structured English expressions and allows a human user to add a refined description of the environment increasing the chances to find a correct-by-design plan. Sharma et al. [26] map natural language sentences to transformations of cost functions that enable users to correct goals and update motions to incorporate user preferences and recover from planning errors. Some other works propose quantitative evaluation of LTL specification satisfaction, leading to finding the least-violating plans in the given model [32, 33, 21]. To our best knowledge, none of the works on formal methods-based robot task planning with a human in the loop focuses on explicitly communicating advice, that – if followed – enables guarantees.

Several works study the effect of robots giving advice to human in a 1-on-1 human-robot interaction. Barlas [2] studied how robot-instructed actions alter the human's sense of agency. Strait et al. [28] investigate the effects of robot communication strategies in advice-giving situations based on robot appearance, interaction modality and distance. Torrey et al. [29] experimentally compared help-giving strategies. Tucker et al. [30] optimize agents to communicate according to utility, informativeness, and complexity. These works generally study how advice should be conveyed and what factors influence the acceptance of such advice, but they do not focus on what advice should be given in the first place.

Our approach to generating advice is driven by assume-guarantee formal synthesis that has been studied in the broad context of cyber-physical systems. Bloem et al. [5] survey different approaches to handling assumptions over the environment for $GR(1)$ specifications, an efficient fragment of LTL. Various notions of optimality were associated with the synthesized assumptions: Tumova and Dimarogonas [31] formalize least-limiting strategy advisers for safety objectives in game-based models of semi-autonomous systems. Chen et al. [11] use maximally permissive supervisors to generate flexible strategies for manufacturing systems. Bernet et al. [4] define permissive strategies for parity games as strategies that subsume the behavior of all memoryless strategies. Chatterjee et al. [10] generate the weakest sufficient assumptions on the environment. Using this definition of the weakest sufficient assumptions, Boteanu et al. [6] express environment assumptions based on [10] via natural language questions referencing objects in the environment in robot-initiated specification repair. Schuppe and Tumova [25] use the notion of the weakest sufficient assumptions [10] in order to propose a scalable, decentralized multi-agent task-planning method based on robots exchanging advice with each other.

II. Notation and Preliminaries

For a finite set X, a probability distribution on X is a function $\delta : X \to [0,1]$ such that $\sum_{x\in X} \delta(x) = 1$. We write $Supp(\delta) = \{x \in X \mid \delta(x) > 0\}$ for the support set of δ. We denote the set of probability distributions on X by $Dist(X)$.

A. Linear Temporal Logic and Automata

We use LTL interpreted over *finite traces*, called LTL$_f$ [12].

Definition 1 (Syntax of LTL$_f$)**.** *An LTL formula over a set of atomic propositions AP is defined as follows:*

$$\phi := true \mid a \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 U \phi_2 \mid X\phi, \quad a \in AP$$

A trace is a finite word over the alphabet 2^{AP}. $X\phi$ requires a proposition to hold in the *next* step. $\phi_1 U \phi_2$ requires ϕ_1 to hold *until* ϕ_2 holds. Using the operators $\neg$ and $\wedge$, the full power of propositional logic is obtained. Furthermore, $F\phi = true\, U\phi$ represents ϕ holding *eventually* in the future and $G\phi = \neg F \neg\phi$ *always* holding from now on. See [12] for a full introduction and definition of the semantics. An LTL$_f$ formula ϕ can be represented by a *deterministic finite automaton* (DFA) [12]. We use the tool MONA [16] to translate an LTL$_f$ formula ϕ into a DFA A_ϕ that precisely accepts $L(\phi)$. Translating LTL$_f$ formulae into DFA is proven to be EXP-complete.

Definition 2 (Deterministic Finite Automaton)**.** *A DFA is a tuple $A = (Q, \Sigma, \Delta, q_i, F)$ where*

- *Q is the set of states,*
- *Σ the alphabet,*
- *$\Delta : Q \times \Sigma \to Q$ the transition function,*
- *q_i the initial state and*
- *F the set of accepting states.*

A finite run *of a DFA on a trace $\rho = \rho_0\rho_1 \dots \rho_n$, where $\rho_i \in \Sigma$, is the sequence of states $q_0 q_1 \dots q_{n+1}$ such that $q_{i+1} = \Delta(q_i, \rho_i)$ for all $0 \le i \le n$. This run is accepting if $q_{n+1} \in F$.*

B. Stochastic Games

Definition 3 (Labelled Stochastic Game)**.** *A two and a half player stochastic game (SG) is a tuple $G = ((S, E), Act, \delta, s_0, AP, L)$ where*

- *(S, E) is a finite, directed graph,*
- *$S = S_1 \uplus S_2 \uplus S_p$ is the states space partitioned into player 1, player 2, and probabilistic states,*

- *$E = E_1 \uplus E_2 \uplus E_p$ is the set of edges, partitioned analogously,*
- *Act is a set of actions,*
- *$\delta : S_p \to Dist(S_1)$ is a probabilistic transition function,*
- *$s_0 \in S_1$ is the initial state,*
- *AP is a set of atomic propositions and*
- *$L : S_1 \to 2^{AP}$ is a* state labelling function.

We define $E_1 \subseteq S_1 \times Act \times S_2, E_2 \subseteq S_2 \times Act \times S_p, E_p \subseteq S_p \times S_1$. For all $s_p \in S_p$ and $s_1 \in S_1$, we have $(s_p, s_1) \in E$ iff $\delta(s_p)(s_1) > 0$. The state labelling function assigns to every state the atomic propositions which hold true in there.

If each state in S_2 has a single outgoing edge, G is called a *Markov Decision Process* (MDP); in that case we omit S_2 from the definition, and use $E_1 \subseteq S_1 \times Act \times S_p$ in the expected way.

Definition 4 (Plays). *A* play *of game G is an infinite sequence $\pi = s_0 s_1 s_2 \ldots$ of states such that $(s_k, a, s_{k+1}) \in E_1 \cup E_2$ or $(s_k, s_{k+1}) \in E_p$ for all $k \geq 0$. We use Π for the set of all plays.*

Definition 5 (Strategy). *A deterministic finite-memory strategy for player i is a function $\lambda : S_i^* \to Act$, where $\lambda(s_1 \ldots s_i) \in Act(s_i)$. A memoryless strategy is such that $\lambda(s_1 \ldots s_i) = \lambda(s_i)$ for all $s_1 \ldots s_i \in S_i^*$ and we use $\lambda : S_i \to Act$.*

Strategies λ_1, λ_2 define plays that follow them $\pi_{\lambda_1,\lambda_2} = s_0 s_1 s_2 \ldots$, where $(s_k, \lambda_i(s_k), s_{k+1}) \in E_i$ if $s_k \in S_i, i = 1, 2$. For a set of plays $\mathcal{E} \subseteq \Pi$, state s, and strategies λ_1 and λ_2, we denote the probability that a play beginning in s and following λ_1, λ_2 belongs to $\mathcal{E}$ by $Pr_s^{\lambda_1,\lambda_2}(\mathcal{E})$.

Definition 6 (Objective). *An* objective *for a player is a set $\psi \subseteq \Pi$ of plays that are winning for that player.*

Definition 7 (Almost-Sure Winning). *Given an objective ψ, strategy λ_1 of player 1 is* almost-sure winning *from state s, if for every strategy λ_2 of player 2, $Pr_s^{\lambda_1,\lambda_2}(\psi) = 1$.*

Given an objective ψ, we denote the set of states from which player 1 has an almost-sure winning strategy as $\langle\langle 1 \rangle\rangle(\psi)$. Given an objective ψ, the *cooperative winning set* $\langle\langle 1, 2 \rangle\rangle(\psi)$ is the set of states s where there exists a strategies λ_1, λ_2 for players 1 and 2, such that $Pr_s^{\lambda_1,\lambda_2}(\psi) = 1$. Stochastic games with a reachability goal can be solved by state-of-the-art tools, such as PRISM-games [19].

III. Weakest Sufficient Assumptions for Human-Robot Interaction

A. Modelling Approach

We model the robot's capabilities through a finite labelled MDP $\mathcal{M} = (S = S_1 \uplus S_p, Act, \delta, s_0, AP_r, L)$, where AP_r is a set of atomic propositions that describe the status of the robot with respect to its tasks, and $L : S \to AP_r$ is a function that labels state with atomic propositions, which hold there. For simplicity, we assume that each transition (action execution) takes one time unit and the robot's environment is observable at all times. Probabilistic transitions are used to model probabilistic outcomes of chosen actions. These probabilities could originate from modelling success rates of the actions or unmodelled disturbances of the environment.

The robot is given a high-level task formulated as an LTL$_f$ specification ϕ over AP_ϕ. AP_ϕ is a set of atomic propositions that include AP_r as well as other propositions regarding the status of the environment, or the human in the loop. This means the robot's task may involve subtasks that depend on actions of the human. The robot's aim is to satisfy the formula almost-surely; our aim is to find a strategy under which it does so along with assumptions on the behavior of the human in the loop.

Example III.1. *A simple example of a robot in an environment is given in Figure 1. The robot can move in the bottom six grey cells, interact with the five ingredients on the table in the center and it can also deliver at the designated delivery tray at the bottom. This is modeled as an MDP illustrated in Figure 2. The states S of $\mathcal{M}$ describe the position of the robot, expressed through the cell it currently occupies. In each state, the robot can move to an adjacent cell in either of the four cardinal directions or interact with the ingredient or delivery trays. The self-loops indicate not taking an action and idling for one time unit. In this particular case, the probability of each depicted transition is 1 and the action names are omitted for readability.*

The robot's task is to interact with all the ingredients of the table in the center once. The specification formula

$$\phi = \phi_g \wedge \phi_c \tag{1}$$

is split into goals and constraints. The goal formula ϕ_g describes that the robot should interact with all ingredient trays, regardless of their order and that it requires help from the human when interacting with the ketchup. The constraint formula ϕ_c forbids the collision when both actors reach into the same tray at the same time at all other ingredient trays. Clearly, to accomplish this task, the robot will need to assume some cooperation from the human.

$$\begin{aligned} \phi_g = {} & F\ buns_r \wedge F\ patty_r \wedge F\ lettuce_r && (2)\\ & \wedge\ F(ketchup_r \wedge ketchup_h) \wedge F tomato_r \\ \phi_c = {} & G\ \neg(buns_r \wedge buns_h) \wedge G\ \neg(patty_r \wedge patty_h) && (3)\\ & \wedge\ G\ \neg(lettuce_r \wedge lettuce_h) \wedge G\ \neg(tomato_r \wedge tomato_h) \end{aligned}$$

In this work, we do not aim to explicitly represent the human's states, goals or intention through a model. For task planning purposes, we abstract the influence of the human's actions on the environment relevant to the robot's task through a set of atomic propositions and their changes. We denote these atomic propositions $AP_h = AP_\phi \setminus AP_r$. The robot considers the human to be able to freely change an arbitrary subset of atomic proposition from AP_h in every step and the assumptions that we aim to synthesize take the form of restrictions on changes to this set.

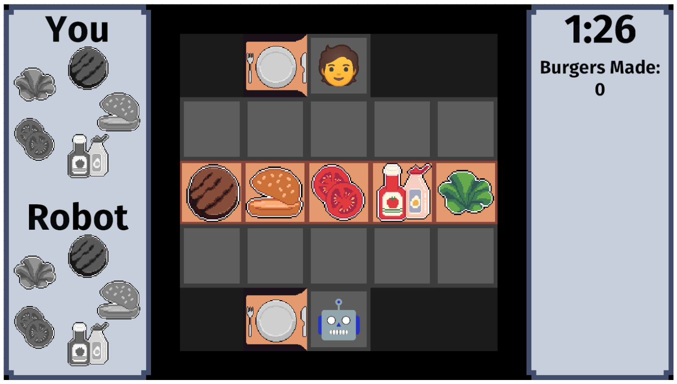

Fig. 1. The study game screen with the playing field in the center, ingredient inventory on the left, timer and burger counter on the right.

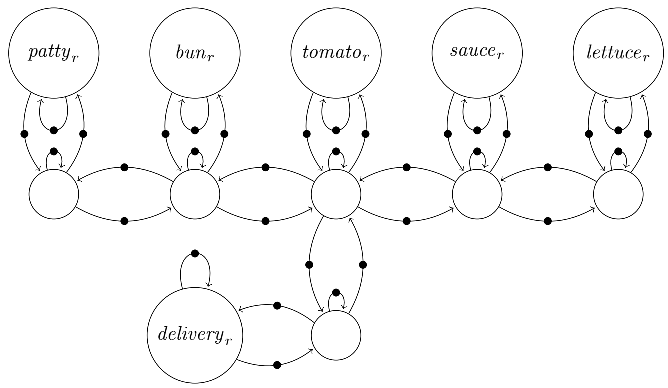

Fig. 2. Robot MDP of the study game. States in S_p are depicted as small dots and the outgoing edges from them are associated with probability 1. Propositions that hold in a state are denoted inside of the state.

We model the interaction through a stochastic game. Starting from the initial state, the robot represented through player 1 makes a decision on its move, takes an action from Act and induces a change to propositions from AP_r. Meanwhile, the human represented through player 2 may make a change to propositions from AP_h. Note that this change can be triggered by one or more actions of the human. The remaining half player corresponds to the resolution of probabilistic outcomes of the robot MDP's actions. While a concurrent game seems to be the most suitable choice, we opt for turn-based games as they are simpler to handle algorithmically with the objectives at hand. At the same time, in a turn-based game, player 2 is a stronger adversary, so winning resulting strategies for player 1 in a turn-based game are conservative, but also winning strategies for player 1 in a concurrent game, where player 2 has to choose without knowledge of the action of player 1.

Definition 8 (Interaction Model)**.** *Given a robot represented as a labelled MDP $\mathcal{M} = ((S = S_1 \uplus S_p, E), Act, \delta, s_0, AP_r, L)$ and a set of propositions controlled by the human actor $\Sigma_h = 2^{AP_h}$, we create a labelled stochastic game*

$$\widetilde{G} = ((\widetilde{S}, \widetilde{E}), Act \cup \{\epsilon\}, \tilde{\delta}, s_0, AP_r, L)$$

as follows:

1) State space. *$\widetilde{S} = S_1 \uplus S_p \uplus \widetilde{S}_p$, where $\widetilde{S}_p = S_p \times \Sigma_h$ $\tilde{s}_p = (s_p, \sigma)$ is the state of the game after the robot took an action and the human chose atomic propositions that hold true now. Without loss of generality, in practice $\widetilde{S}$ will include only the states reachable from $\tilde{s}_0$.*
2) Actions. *The actions of the game are the actions of the robot; the actions of humans as well as the half-player are not labeled.*
3) Edges. *$\widetilde{E} = E \uplus E_p \uplus \widetilde{E}_p$.*
   - *$E_p = \{(s_p, \tilde{s}_p) \mid s_p \in S_p, \tilde{s}_p = (s_p, \sigma) \in \widetilde{S}_p\}$ represent a change in the propositions triggered by the human.*
   - *$\widetilde{E}_p = \{(\tilde{s}_p, s_1) \mid \tilde{s}_p \in \widetilde{S}_p, s_1 \in S_1, s_1 \in Supp(\delta(s_p))\}$ resolve the probabilistic outcome of the actions of the robot MDP.*
4) Probabilistic Transition Function. *$\tilde{\delta} : \widetilde{S}_p \to Dist(S_1)$. Given $\tilde{s}_p = (s_p, \sigma) \in \widetilde{S}_p$, we define:*

$$\tilde{\delta}(\tilde{s}_p)(s_1) = \delta(s_p)(s_1).$$

B. Strategy Synthesis

In the next step, we introduce the specification of the robot into the game. First, we translate ϕ into a DFA A_ϕ. We then construct a stochastic game with a reachability objective that is determined from the accepting states of the DFA. In this game, finding an almost-surely winning strategy translates to satisfying the specification ϕ.

Definition 9 (Synthesis Game)**.** *Given the interaction model $\widetilde{G} = ((\widetilde{S}, \widetilde{E}), Act \cup \{\epsilon\}, \tilde{\delta}, s_0, AP_r, L)$ and a DFA $A_\phi = (Q, 2^{AP_\phi}, q_0, \Delta, F)$, where $AP_r \subset AP_\phi$, $AP_h = AP_\phi \setminus AP_r$ and $\Sigma_h = 2^{AP_h}$, we create a labelled stochastic game*

$$\widehat{G} = ((\widehat{S}, \widehat{E}), Act \cup \{\epsilon\}, \hat{\delta}, \hat{s}_0, AP_r, \widehat{L}),$$

as follows:

1) State space. *$\widehat{S} = \widetilde{S} \times Q$. All states carry information about the current state of the interaction model and the DFA.*
2) Edges. *$\widehat{E} = \widehat{E}_1 \uplus \widehat{E}_2 \uplus \widehat{E}_p$. The state of the DFA is updated on edges in $\widehat{E}_p$. All other edges keep their original mapping, carrying over the current DFA state.*
   a) *$\widehat{E}_1 = \{(\hat{s}_1, a, \hat{s}_2) \mid \hat{s}_1 = (\tilde{s}_1, q) \in \widehat{S}_1, a \in Act, \hat{s}_2 = (\tilde{s}_p, q) \in \widehat{S}_2$, and $(\tilde{s}_1, a, \tilde{s}_p) \in \widetilde{E}\}$*
   b) *$\widehat{E}_2 = \{(\hat{s}_2, \epsilon, \hat{s}_p) \mid \hat{s}_2 = (\tilde{s}_2, q) \in \widehat{S}_2, \hat{s}_p = (\tilde{s}_p, q) \in \widehat{S}_p$, and $(\tilde{s}_2, \epsilon, \tilde{s}_p) \in \widetilde{E}\}$*
   c) *$\widehat{E}_p = \{(\hat{s}_p, \hat{s}_1) \mid \hat{s}_p = (s_p, q, \sigma) \in \widehat{S}_p, \hat{s}_1 = (\tilde{s}_1, q') \in \widehat{S}_1, \Delta(q, \sigma \cup \widetilde{L}(\tilde{s}_1)) = q')$ and $\tilde{s}_1 \in Supp(\tilde{\delta}(\tilde{s}_p))\}$.*
3) Probabilistic Transition Function. *$\hat{\delta} : \widehat{S}_p \to Dist(\widehat{S}_1)$. Given $\hat{s}_p = (s_p, q, \sigma) \in \widehat{S}_p$ and $\hat{s}_1 = (\tilde{s}_1, q') \in \widehat{S}_1$, we define:*

$$\hat{\delta}(\hat{s}_p)(\hat{s}_1) = \begin{cases} \tilde{\delta}(\tilde{s}_p)(\tilde{s}_1), & \text{if } \Delta(q, \sigma \cup \widetilde{L}_s(\tilde{s}_1)) = q' \\ 0, & \text{otherwise.} \end{cases}$$

4) Initial State. $\hat{s}_0 = (s_0, q_0)$.

5) State Labelling Function. $\widehat{L} : \widehat{S}_1 \rightarrow AP_r$. *Given* $\hat{s}_1 = (s_1, q) \in \widehat{S}_1$, $\widehat{L}(\hat{s}_1) = L(s_1)$.

The objective of the game is to reach an accepting state of the DFA:

$$\psi = \{s_0 s_1 s_2 \dots \in \Pi \mid \exists k \geq 0, s_k = (s, q) \in S, q \in F\} \quad (4)$$

If the robot can find an almost-surely winning strategy in $\widehat{G}$ for ψ, it satisfies ϕ regardless of what the human does. If not, additional assumptions on the human's actions, or more precisely their effect on the atomic proposition, are necessary to offer guarantees. To synthesise sufficient assumptions, we leverage the fact that any linear-time property, such as a property expressed in LTL$_f$ can be decomposed into a safety and a liveness component [1]. Based on this observation, we generate sufficient assumptions split into two parts: Safety assumptions and fairness assumptions expressed as a set of player-2 edges from $\widehat{G}$. Safety assumptions $E_s \subseteq E_2$ are edges that player 2 can never take. Fairness assumptions $E_l \subseteq E_2$ of edges that need to be chosen fairly (i.e. infinitely many times upon infinitely many visits to their outgoing state).

Definition 10 (Sufficient Assumptions[25])**.** *Given an objective ψ, a safety assumption $E_s \subseteq E_2$ is sufficient if player 1 has an almost-sure winning strategy for the objective*

$$AssumeSafe(\psi, E_s) = \psi \cup \{\pi = s_0 s_1 s_2 \dots \mid \exists k \geq 0, (s_k, s_{k+1}) \in E_s\}$$

A fairness assumption E_l is sufficient if player 1 has an almost-sure winning strategy for the objective

$$AssumeFair(\psi, E_l) = \psi \cup \{\pi = s_0 s_1 s_2 \dots \mid \exists (s, s') \in E_l \text{ s.t. } s_k = s \text{ for infinitely many } k \text{ but } s_{k+1} = s' \text{ only finitely often.}\}$$

Furthermore, a set of sufficient assumptions E_* is the *weakest* assumption if there exists no smaller set $|E'_*| \leq |E_*|$, $E'_* \in E_2, * \in \{s, l\}$ is sufficient. The weakest sufficient assumptions can be algorithmically computed [25]. We refer to [25][10] for full definition, proof and the algorithm to compute sufficient and the weakest sufficient safety and fairness assumptions.

A key insight is that given a set of the weakest sufficient assumptions E_s, E_l, there exists an almost-surely winning strategy in $\widehat{G}$ for the objective $AssumeFair(AssumeSafe(\psi, E_s), E_l)$. In other words, assuming that edges from E_s are never taken, while edges form E_l are fairly taken, specification ϕ is guaranteed to be satisfied. In other words, if the human is able and willing to never change the atomic propositions that trigger edges E_s in $\widehat{G}$ and to eventually change the atomic propositions that trigger edges E_l in $\widehat{G}$, the robot will be able to provably accomplish its task.

C. Communicating the Assumptions via Advice

After computing the weakest sufficient assumptions E_s and E_l and the corresponding robot's strategy, the assumptions have to be communicated to the human in a way they can easily understand and follow. Hence, instead of communicating these sets directly, we propose to monitor the robot and the human (i.e. the state of the game $\widehat{G}$) and issue advice at runtime. In particular, if one of the outgoing edges of the current state is found in the safety assumption set E_s, we identify the propositions σ from $(\tilde{s_2}, \tilde{s_p}) \in E_s$, and $\tilde{s_p} = (s_p, q, \sigma)$. The safety assumption is then communicated as forbidden changes to the atomic propositions. In Example III.1, this could be for instance communicating to the human that they should not interact with the patty when the robot does so. The fairness assumptions can be communicated as a suggestion to make a particular change to atomic propositions. In Example III.1, this could be for instance suggesting to the human that they should eventually interact with the sauce while the robot waits for their help. If this advice is followed at all times, the assumptions are fulfilled and hence the guarantees hold. We give a concrete example of communicating the assumptions via advice below in Section IV.

IV. Using Least-Limiting Advice in an Online User Study

The full source code for game construction, solving and assumption computation is available on GitHub[1]. We used Python 3 to construct and manipulate the stochastic games and PRISM-games [19] to solve instances. Source code of the study and raw result data are in a separate repository[2]. The interactive game is written in Rust and compiled to WASM, so participants could play in their browser. The repository also includes other experimental materials such as questionnaires.

We designed an online user study where participants play an interactive game with a robot. We examine if humans are able and willing to follow computed assumptions given by the robot, how they evaluate the experience, and the effect of communicating assumptions on the specification satisfaction.

A. Interactive Game Design

A snapshot of the interactive game is illustrated in Figure 1. The robot's capabilities, MDP model, and specification are discussed in Example III.1. Before playing the game, the participants are given instructions that explain the game in detail. They control the character represented by the human icon via arrow keys. Every time the participant moves, the robot moves as well.

The participants are tasked to assemble hamburgers alongside their robotic co-worker and to collectively produce as many burgers as possible in a given time limit. For this, they have to collect all necessary ingredients from ingredient trays and then deliver the burger to the plate at the top. An inventory of ingredients is displayed on the left of the screen. Collected

[1] https://github.com/KTH-RPL-Planiacs/human-advisers
[2] https://github.com/KTH-RPL-Planiacs/human-adviser-study

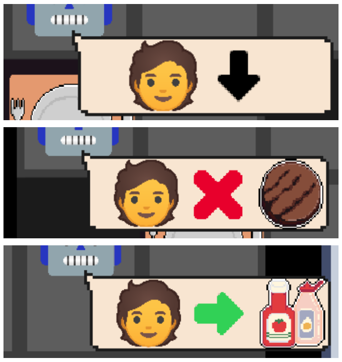

Fig. 3. From top to bottom: Next-Move Advice, Safety Advice and Fairness Advice

	Least Limiting	Next Move	None
Perceived Intelligence			
Incompetent-Competent	4.05 (0.91)	3.42 (1.54)	3.81 (1.14)
Ignorant-Knowledgeable	4.22 (0.85)	3.56 (1.52)	3.97 (1.00)
Irresponsible-Responsible	3.89 (0.96)	3.33 (1.31)	3.78 (1.07)
Unintelligent-Intelligent	3.78 (1.16)	3.44 (1.44)	3.78 (1.22)
Perceived Safety			
Anxious-Relaxed	4.17 (1.00)	3.11 (1.47)	3.94 (1.31)
Agitated-Calm	4.06 (0.98)	3.03 (1.54)	4.00 (1.27)
Quiescent-Surprised	3.78 (1.05)	3.39 (1.22)	3.44 (1.18)
Perceived Hindrance			
Unpredictable-Predictable	3.95 (0.91)	3.53 (1.34)	3.81 (1.14)
Incompliant-Compliant	3.86 (0.95)	2.97 (1.40)	3.69 (1.21)
Unrestrictive-Restrictive	3.86 (0.95)	3.89 (1.01)	3.47 (1.23)

TABLE I

MEANS AND STANDARD DEVIATIONS (IN PARENTHESES) OF THE QUESTIONNAIRE RATINGS.

ingredients are coloured, otherwise they are greyed out. On the right, a timer shows the remaining time until the game is over and a counter displays the total amount of delivered burgers.

The participants are aware that for safety reasons, the human and the robot should not reach into the same tray at the same time. If the participant and the robot move into the same ingredient tray at the same time, they crash and the screen fades to black, the currently collected ingredients are lost and the robot's and participant's positions are reset to the starting positions. The synthesis game $\widetilde{G}$ restarts to the initial state also every time the robot satisfies the specification. Advice is communicated via pictograms. Safety advice forbids certain atomic proposition changes (coming, e.g. from the safety assumption E_s) is communicated by a red cross as illustrated in Figure 3 in the middle. Fairness advice suggests that a certain atomic proposition should change eventually (coming, e.g. from the fairness assumption E_l) is communicated by a green arrow as illustrated in Figure 3 in the middle. Finally, advice that dictates to the human its next move is communicated via a black arrow as illustrated in Figure 3 in the bottom.

B. Study design

We compare our approach (*LeastLimiting*) to two baselines:

- *NoAdvice*: The robot does not issue any advice.
- *NextMove*: The robot communicates to the user exactly which move to take in every step. The moves are determined by computing a cooperative strategy for the synthesis game. In other words, we treat the human as fully controllable. Following the *NextMove* advice leads to provable satisfaction of the robot's specification Equation 1; this approach however does not focus on the advice to be the least limiting.

The main purpose of our study was to answer (1) whether the *LeastLimiting* advice lead to less specification violations than *NoAdvice* and (2) whether the *LeastLimiting* advice were perceived as less limiting than the *NextMove* advice.

We recorded the following data in-game:

- total steps taken by the human
- number of burgers produced by the robot
- number of burgers produced by the human
- number of specification violations measured through violation of safety constraints in (3)

After playing the game for two minutes, we asked the participants to rate their impressions of the robot on semantic differential scales from 1-5 between different adjective pairs. The questionnaire aims to study three aspects: perceived intelligence, perceived safety and perceived hindrance. All attribute pairs are shown in Figure IV-B. For perceived intelligence and safety, we used the corresponding parts from the Godspeed questionnaire [3]. Since the "perceived hindrance" scale does not belong to a validated questionnaire, we analysed its component items individually. In addition, we also collected anonymous demographics data.

C. Study Results

We conducted the study on Amazon Mechanical Turk, and invited 120 participants evenly split over the three conditions. Of the 120 participants, 50 self-described as female, 57 as male, and 2 as non-binary; their age ranged from 24 to 72 years old (median = 36). The vast majority (N = 85) self-described as native English speakers, 5 as native-like, 14 as fluent, and 5 did not respond. Finally, we asked them about their experience with playing video games: 42 people reported playing every day, 46 weekly, 14 monthly, and 7 did not respond. We paid each participant $2 plus a performance bonus based on the amount of burgers they produced in the game. The study took on average 18 minutes, with the game taking a fixed two minutes. We conducted the study in accordance with the ethical guidelines of the hosting institution.

After data collection, some participants had to be excluded due to poor performance, or for signing up to the study twice. As a result, the final dataset contained data from 109 participants.

All analyses were performed in R version 4.2.1. For the questionnaire data, we conducted one-way ANOVAs to see if the adviser mode had any effect on the different scales. We found that adviser mode significantly influenced perceived safety ($F(2, 105) = 6.27$, $MSE = 0.63$, $p = .003$, $\hat{\eta}^2_G = .107$); post-hoc pairwise comparisons using the Tukey HSD test showed that the LeastLimiting condition was perceived as safer than the Next Move condition ($p = .009$). The effect of the adviser mode approached significance at the .05 significance level for perceived intelligence ($F(2, 106) = 2.88$, $MSE = 1.16$, $p = .061$, $\hat{\eta}^2_G = .052$), with the LeastLimiting condition being perceived as more intelligent than the Next Move condition (Tukey HSD $p = .05$). Finally, for perceived hindrance, we only found an effect of adviser mode in the Incompliant - Compliant adjective pair ($F(2, 106) = 5.65$, $MSE = 1.44$, $p = .005$, $\hat{\eta}^2_G = .096$), with the LeastLimiting condition being perceived as more compliant than the Next Move condition (Tukey HSD $p = .005$).

We conducted a Poisson regression on the number of safety violations, and we found that there were significantly fewer violations in the LeastLimiting condition than in the No-Advice baseline ($b = 0.62$, 95% CI $[0.08, 1.18]$, $z = 2.21$, $p = .027$). At the same time, burger productivity remained roughly the same between the two. The total amount of burgers produced were 142 in the LeastLimiting condition, 154 in the NoAdvice condition and 76 in the NextMove condition.

V. Discussion & Conclusion

We presented a novel, formal methods-based approach to a human-in-the-loop task planning. We proposed to synthesize and communicate advice, which – if followed – enables the robot to successfully accomplish its task. The advice is based on the weakest sufficient assumptions in stochastic games. Our results suggest that a robot communicating this type of advice is perceived as safer, more intelligent and more compliant than a robot giving more frequent advice based on sufficient, but stronger assumptions. Our study results also indicate that our advice leads to less violations of the robot's task specification when compared to not communicating any advice.

The presented theory also applies when modelling multiple human actors instead of only one, with each actor only controlling a subset of AP_h. The assumption computation remains unchanged, but communicating assumptions for multiple human actors remains future research. Since our user study features only the single-human case, we restricted the notation for the sake of readability.

In this work, we opted to use non-verbal communication in the form of pictograms, communicating only a subset of possible assumptions. The best method for conveying assumptions remains unstudied, especially for real robots.Other non-verbal methods [9], such as light- and sound-based methods [27][24] present interesting directions for future research. Based on existing works on generating natural language from logical statements [6], studying if natural language improves communication of assumptions also remains future research.

Acknowledgments

This work is partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation and the Swedish Research Council (VR) (project no. 2017-05102). This research has been carried out as part of the Vinnova Competence Center for Trustworthy Edge Computing Systems and Applications at KTH Royal Institute of Technology. The authors are also affiliated with Digital Futures. The authors thank Lisa Dünte for providing graphics for the user study and Alexis Linard and Patrick Hammer for their helpful feedback on the initial draft.

References

[1] Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT press, 2008.

[2] Zeynep Barlas. When robots tell you what to do: Sense of agency in human-and robot-guided actions. *Consciousness and cognition*, 75:102819, 2019.

[3] Christoph Bartneck, Dana Kulić, Elizabeth Croft, and Susana Zoghbi. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International journal of social robotics*, 1:71–81, 2009.

[4] Julien Bernet, David Janin, and Igor Walukiewicz. Permissive strategies: from parity games to safety games. *RAIRO-Theoretical Informatics and Applications*, 36(3): 261–275, 2002.

[5] Roderick Bloem, Rüdiger Ehlers, Swen Jacobs, and Robert Könighofer. How to handle assumptions in synthesis. *arXiv preprint arXiv:1407.5395*, 2014.

[6] Adrian Boteanu, Jacob Arkin, Siddharth Patki, Thomas Howard, and Hadas Kress-Gazit. Robot-initiated specification repair through grounded language interaction. *arXiv preprint arXiv:1710.01417*, 2017.

[7] Guilhem Buisan and Rachid Alami. A Human-Aware Task Planner Explicitly Reasoning About Human and Robot Decision, Action and Reaction. In *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pages 544–548, 2021.

[8] Colin F Camerer. *Behavioral game theory: Experiments in strategic interaction*. Princeton university press, 2011.

[9] Elizabeth Cha, Yunkyung Kim, Terrence Fong, Maja J Mataric, et al. A survey of nonverbal signaling methods for non-humanoid robots. *Foundations and Trends® in Robotics*, 6(4):211–323, 2018.

[10] Krishnendu Chatterjee, Thomas A Henzinger, and Barbara Jobstmann. Environment assumptions for synthesis. In *International Conference on Concurrency Theory*, pages 147–161. Springer, 2008.

[11] YuFeng Chen, ZhiWu Li, Mohamed Khalgui, and Olfa Mosbahi. Design of a maximally permissive liveness-enforcing Petri net supervisor for flexible manufacturing systems. *IEEE Transactions on automation science and engineering*, 8(2):374–393, 2010.

[12] Giuseppe De Giacomo and Moshe Y Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *IJCAI'13 Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 854–860. Association for Computing Machinery, 2013.

[13] Lu Feng, Clemens Wiltsche, Laura Humphrey, and Ufuk Topcu. Synthesis of human-in-the-loop control protocols

for autonomous systems. *IEEE Transactions on Automation Science and Engineering*, 13(2):450–462, 2016.

[14] David Gundana and Hadas Kress-Gazit. Event-Based Signal Temporal Logic Tasks: Execution and Feedback in Complex Environments. *IEEE Robotics and Automation Letters*, 7(4):10001–10008, 2022.

[15] Keliang He, Andrew M. Wells, Lydia E. Kavraki, and Moshe Y. Vardi. Efficient Symbolic Reactive Synthesis for Finite-Horizon Tasks. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8993–8999, 2019. doi: 10.1109/ICRA.2019.8794170.

[16] Jesper G Henriksen, Jakob Jensen, Michael Jørgensen, Nils Klarlund, Robert Paige, Theis Rauhe, and Anders Sandholm. Mona: Monadic second-order logic in practice. In *International Workshop on Tools and Algorithms for the Construction and Analysis of Systems*, pages 89–110. Springer, 1995.

[17] Sebastian Junges, Nils Jansen, Joost-Pieter Katoen, Ufuk Topcu, Ruohan Zhang, and Mary Hayhoe. Model checking for safe navigation among humans. In *International Conference on Quantitative Evaluation of Systems*, pages 207–222. Springer, 2018.

[18] Hadas Kress-Gazit, Morteza Lahijanian, and Vasumathi Raman. Synthesis for robots: Guarantees and feedback for robot behavior. *Annual Review of Control, Robotics, and Autonomous Systems*, 1:211–236, 2018.

[19] M. Kwiatkowska, G. Norman, D. Parker, and G. Santos. PRISM-games 3.0: Stochastic Game Verification with Concurrency, Equilibria and Time. In *Proc. 32nd International Conference on Computer Aided Verification (CAV'20)*, volume 12225 of *LNCS*, pages 475–487. Springer, 2020.

[20] Shen Li, Daehyung Park, Yoonchang Sung, Julie A Shah, and Nicholas Roy. Reactive task and motion planning under temporal logic specifications. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12618–12624. IEEE, 2021.

[21] Matthew R Maly, Morteza Lahijanian, Lydia E Kavraki, Hadas Kress-Gazit, and Moshe Y Vardi. Iterative temporal motion planning for hybrid systems in partially unknown environments. In *Proceedings of the 16th international conference on Hybrid systems: computation and control*, pages 353–362, 2013.

[22] Toki Migimatsu and Jeannette Bohg. Object-Centric Task and Motion Planning in Dynamic Environments. *IEEE Robotics and Automation Letters*, 5(2):844–851, 2020. doi: 10.1109/LRA.2020.2965875.

[23] Vasumathi Raman and Hadas Kress-Gazit. Automated feedback for unachievable high-level robot behaviors. In *2012 IEEE International Conference on Robotics and Automation*, pages 5156–5162. IEEE, 2012.

[24] Frederic Anthony Robinson, Mari Velonaki, and Oliver Bown. Smooth Operator: Tuning Robot Perception Through Artificial Movement Sound. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '21, page 53–62, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450382892. doi: 10.1145/3434073.3444658. URL https://doi.org/10.1145/3434073.3444658.

[25] Georg Friedrich Schuppe and Jana Tumova. Decentralized Multi-Agent Strategy Synthesis under LTL f Specifications via Exchange of Least-Limiting Advisers. In *2021 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 119–127. IEEE, 2021.

[26] Pratyusha Sharma, Balakumar Sundaralingam, Valts Blukis, Chris Paxton, Tucker Hermans, Antonio Torralba, Jacob Andreas, and Dieter Fox. Correcting Robot Plans with Natural Language Feedback. In *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, June 2022. doi: 10.15607/RSS.2022.XVIII.065.

[27] Sichao Song and Seiji Yamada. Bioluminescence-Inspired Human-Robot Interaction: Designing Expressive Lights That Affect Human's Willingness to Interact with a Robot. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '18, page 224–232, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450349536. doi: 10.1145/3171221.3171249. URL https://doi.org/10.1145/3171221.3171249.

[28] Megan Strait, Cody Canning, and Matthias Scheutz. Let Me Tell You! Investigating the Effects of Robot Communication Strategies in Advice-Giving Situations Based on Robot Appearance, Interaction Modality and Distance. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '14, page 479–486, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450326582. doi: 10.1145/2559636.2559670. URL https://doi.org/10.1145/2559636.2559670.

[29] Cristen Torrey, Susan R Fussell, and Sara Kiesler. How a robot should give advice. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 275–282. IEEE, 2013.

[30] Mycal Tucker, Julie Shah, Roger Levy, and Noga Zaslavsky. Towards human-agent communication via the information bottleneck principle. *arXiv preprint arXiv:2207.00088*, 2022.

[31] Jana Tumova and Dimos V Dimarogonas. Synthesizing least-limiting guidelines for safety of semi-autonomous systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 5714–5719. IEEE, 2016.

[32] Jana Tumova, Gavin C Hall, Sertac Karaman, Emilio Frazzoli, and Daniela Rus. Least-violating control strategy synthesis with safety rules. In *Proceedings of the 16th international conference on Hybrid systems: computation and control*, pages 1–10, 2013.

[33] Jana Tumova, Alejandro Marzinotto, Dimos V Dimarogonas, and Danica Kragic. Maximally satisfying LTL action planning. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1503–1510. IEEE, 2014.