Multi-Objective Controller Synthesis with Uncertain Human Preferences

Shenghui Chen*
University of Texas at Austin, USA
shenghui.chen@utexas.edu

Kayla Boggess*
University of Virginia, USA
kjb5we@virginia.edu

David Parker
University of Birmingham, UK
d.a.parker@cs.bham.ac.uk

Lu Feng
University of Virginia, USA
lu.feng@virginia.edu

ABSTRACT

Complex real-world applications of cyber-physical systems give rise to the need for *multi-objective controller synthesis*, which concerns the problem of computing an optimal controller subject to multiple (possibly conflicting) criteria. The relative importance of objectives is often specified by human decision-makers. However, there is inherent uncertainty in human preferences (e.g., due to artifacts resulting from different preference elicitation methods). In this paper, we formalize the notion of *uncertain human preferences*, and present a novel approach that accounts for this uncertainty in the context of multi-objective controller synthesis for Markov decision processes (MDPs). Our approach is based on mixed-integer linear programming and synthesizes an optimally permissive multi-strategy that satisfies uncertain human preferences with respect to a multi-objective property. Experimental results on a range of large case studies show that the proposed approach is feasible and scalable across varying MDP model sizes and uncertainty levels of human preferences. Evaluation via an online user study also demonstrates the quality and benefits of the synthesized controllers.

KEYWORDS

Multi-Objective Controller Synthesis, Markov Decision Processes, Uncertain Human Preferences

1 INTRODUCTION

Controller synthesis—which offers automated techniques to synthesize controllers that satisfy certain properties— has been increasingly used in the design of cyber-physical systems (CPS), including applications such as semi-autonomous driving [33], robotic planning [22], and human-in-the-loop CPS control [14]. Many complex real-world CPS applications give rise to the need for *multi-objective controller synthesis*, which computes an optimal controller subject to multiple (possibly conflicting) criteria. Examples are synthesizing an optimal controller to maximize safety while minimizing fuel consumption for an automotive vehicle, or synthesizing an optimal robotic controller to minimize the mission completion time while minimizing the risk in disaster search and rescue. An optimal solution to multi-objective controller synthesis should account for the trade-off between multiple objective properties. There may not exist a single global solution that optimizes each individual objective property simultaneously. Instead, a set of *Pareto optimal* points can be computed: those for which no objective can be optimized further without worsening some other objectives.

For many applications that involve human decision-makers, they can be presented with these Pareto optimal solutions to decide which one to choose. Alternatively, humans can specify *a priori* their preferences about the relative importance of objectives, which are then used as weights in the multi-objective controller synthesis to compute an optimal solution based on the weighted sum of objectives. We can ask humans to assign objective weights directly; however, sometimes it can be difficult for them to come up with these values. As surveyed in [26], there exist many different approaches for eliciting human preferences, such as ranking, rating, and pairwise comparison. Various preference elicitation methods can yield different weight values as artifacts. Moreover, human preferences can evolve over time and vary across multiple users. Thus, there is inherent uncertainty in human preferences.

In this work, we study the problem of multi-objective controller synthesis with uncertain human preferences. *To the best of our knowledge, this is the first work that takes into account the uncertainty of human preferences in multi-objective controller synthesis.* We address the following research challenges: How to mathematically represent the uncertainty in human preferences? How to account for uncertain human preferences in multi-objective controller synthesis? How to generate a succinct representation of the synthesis results? And how to evaluate the synthesized controllers?

Specifically, we focus on the modeling formalism of Markov decision processes (MDPs), which have been popularly applied for the controller synthesis of CPS that exhibit stochastic and nondeterministic behavior (e.g., robots [22], human-in-the-loop CPS [14]). In recent years, theories and algorithms have been developed for the formal verification and controller synthesis of MDPs subject to multi-objective properties [4, 8, 11, 15, 16, 20]. However, none of the existing work takes into account the uncertainty in human preferences.

We formalize the notion of *uncertain human preferences* as an interval weight vector that comprises a convex set of weight vectors over objectives. Since each weight vector corresponds to some controller that optimizes the weighted sum of objectives, an interval weight vector would yield a set of controllers (i.e., MDP strategies). We adopt the notion of *multi-strategy* [10] to succinctly represent a set of MDP strategies. A (deterministic, memoryless) multi-strategy specifies multiple possible actions in each MDP state. Thus, a multi-strategy represents a set of compliant MDP strategies, each of which chooses an action that is allowed by the multi-strategy in each MDP

*Equal contribution. This research was conducted when Shenghui Chen was a student at the University of Virginia.

state. We define the soundness of a multi-strategy with respect to a multi-objective property, and an interval weight vector representing uncertain human preferences. We also quantify the permissivity of a multi-strategy by measuring the degree to which actions are allowed in (reachable) MDP states. A sound, permissive multi-strategy can enable more flexibility in CPS design and execution. For example, if an action in an MDP state becomes infeasible during the system execution (e.g., some robotic action cannot be executed due to an evolving and uncertain environment), then alternative actions allowed by the multi-strategy can be executed instead, still guaranteeing satisfaction of the human preferences.

We develop a mixed-integer linear programming (MILP) based approach to synthesize a sound, optimally permissive multi-strategy with respect to a multi-objective MDP property and uncertain human preferences. Our approach is inspired by [10], which presents an MILP-based method for synthesizing permissive strategies in stochastic games (of which MDPs are a special case). However, there are several key differences in our encodings. First, we solve multi-objective optimization problems, while [10] is for a single objective. Second, we have a different soundness definition for the multi-strategy and need to track the values of both lower and upper bounds of each objective, while [10] only considers one direction. Lastly, we have a different definition of permissivity which only considers reachable states under a multi-strategy.

We evaluate the proposed approach on a range of large case studies. The experimental results show that our MILP-based approach is scalable to synthesize sound, optimally permissive multi-strategies for large models with more than 10^6 MDP states. Moreover, the results show that increasing the uncertainty of human preferences yields more permissive multi-strategies.

In addition, we evaluate the quality of synthesized controllers via an online user study with 100 participants using Amazon Mechanical Turk. The study results show that strategies synthesized based on human preferences are more favorable, perceived as more accurate, and lead to better user satisfaction, compared to arbitrary strategies. In addition, multi-strategies are perceived as more informative and satisfying than less permissive (single) strategies.

Contributions. We summarize the major contributions of this work as follows.

- We formalized the notion of uncertain human preferences, and developed an MILP-based approach to synthesize a sound, optimally permissive multi-strategy for a given multi-objective MDP property and uncertain human preferences.
- We implemented the proposed approach and evaluated it on a range of large case studies to demonstrate its feasibility and scalability.
- We designed and conducted an online user study to evaluate the quality and benefits of the synthesized controllers.

Paper Organization. In the rest of the paper, we introduce some background about MDPs and multi-objective properties in Section 2, formalize uncertain human preferences in Section 3, develop the controller synthesis approach in Section 4, present experimental results in Section 5, describe the user study in Section 6, survey related work in Section 7, and draw conclusions in Section 8.

2 BACKGROUND

In this section, we introduce the necessary background about MDPs and multi-objective properties.

A *Markov decision process (MDP)* is a tuple $\mathcal{M} = (S, s_0, A, \delta)$, where S is a finite set of states, $s_0 \in S$ is an initial state, A is a set of actions, and $\delta : S \times A \to Dist(S)$ is a probabilistic transition function with $Dist(S)$ denoting the set of probability distributions over S. Each state $s \in S$ has a set of *enabled* actions, given by $\alpha(s) \stackrel{\text{def}}{=} \{a \in A | \delta(s, a) \text{ is defined}\}$. A *path* through $\mathcal{M}$ is a sequence $\pi = s_0 a_0 s_1 a_1 \ldots$ where $a_i \in \alpha(s_i)$ and $\delta(s_i, a_i)(s_{i+1}) > 0$ for all $i \geq 0$. We say that a state s is *reachable* if there exists a finite path starting from s_0 and ending in s as the last state. Let *FPaths* (*IPaths*) denote the set of finite (infinite) paths through $\mathcal{M}$.

A *strategy* (also called a policy) is a function $\sigma : FPaths \to Dist(A)$ that resolves the nondeterministic choice of actions in each state based on the execution history. A strategy σ is *deterministic* if $\sigma(\pi)$ is a point distribution for all π, and *randomized* otherwise. A strategy σ is *memoryless* if the action choice $\sigma(\pi)$ depends only on the last state of π. In this work, we focus on deterministic, memoryless strategies.[1] Thus, we can simplify the definition of strategy to a function $\sigma : S \to A$. Let $\Sigma_{\mathcal{M}}$ denote the set of all (deterministic, memoryless) strategies for $\mathcal{M}$. A strategy $\sigma \in \Sigma_{\mathcal{M}}$ induces a probability measure over *IPaths*, denoted by $Pr^{\sigma}_{\mathcal{M}}$, in the standard fashion [21].

A *reward function* of $\mathcal{M}$ takes the form $r : S \times A \to \mathbb{R}$. The *total reward* along an infinite path $\pi = s_0 a_0 s_1 a_1 \ldots$ is given by $r(\pi) \stackrel{\text{def}}{=} \sum_{t=0}^{\infty} r(s_t, a_t)$. The *expected total reward* for $\mathcal{M}$ under a strategy σ is denoted by $E^{\sigma}_{\mathcal{M}}(r) \stackrel{\text{def}}{=} \int_{\pi} r(\pi)\, dPr^{\sigma}_{\mathcal{M}}$. We say that $\mathcal{M}$ under strategy σ satisfies a *reward predicate* $[r]_{\sim b}$ where $\sim \in \{\geq, \leq\}$ is a relational operator and b is a rational reward bound, denoted $\mathcal{M}, \sigma \models [r]_{\sim b}$, if the expected total reward $E^{\sigma}_{\mathcal{M}}(r) \sim b$. A reward predicate $[r]_{\sim b}$ is *satisfiable* in MDP $\mathcal{M}$ if there exists a strategy $\sigma \in \Sigma_{\mathcal{M}}$ such that $\mathcal{M}, \sigma \models [r]_{\sim b}$. If b is unspecified, we can ask numerical queries, denoted $[r]_{\min} \stackrel{\text{def}}{=} \inf\{x \in \mathbb{R} \mid [r]_{\leq x} \text{ is satisfiable}\}$ and $[r]_{\max} \stackrel{\text{def}}{=} \sup\{x \in \mathbb{R} \mid [r]_{\geq x} \text{ is satisfiable}\}$.

A *multi-objective property* $\phi = ([r_1]_{\bowtie_1}, \ldots, [r_n]_{\bowtie_n})$, where $\bowtie_i \in \{\min, \max\}$, aims to minimize and/or maximize n objectives of expected total rewards simultaneously. For the rest of the paper, we assume that the multi-objective property is of the form $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$. A maximizing objective $[r_i]_{\max}$ can be converted to a minimizing objective by negating rewards. Checking ϕ on MDP $\mathcal{M}$ yields a set of Pareto optimal points that lie on the boundary of the set of *achievable values*:

$$X = \{\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n \mid ([r_1]_{\leq x_1}, \ldots, [r_n]_{\leq x_n}) \text{ is satisfiable}\}.$$

We say that a point $\boldsymbol{x^*} = (x_1^*, \ldots, x_n^*) \in X$ is *Pareto optimal* if there does not exist another point $\boldsymbol{x} = (x_1, \ldots, x_n) \in X$ such that $x_i \leq x_i^*$ for all i and $x_j \neq x_j^*$ for some j. A multi-objective reward predicate $([r_1]_{\leq x_1}, \ldots, [r_n]_{\leq x_n})$ is satisfiable in MDP $\mathcal{M}$ if there exists a strategy $\sigma \in \Sigma_{\mathcal{M}}$ such that $\mathcal{M}, \sigma \models [r_i]_{\leq x_i}$ for all i. The set of achievable values X for ϕ is convex [16].

Given a multi-objective property $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$ and a weight vector $\boldsymbol{w} \in \mathbb{R}^n$, the *expected total weighted reward sum* is

[1] For the types of MDP properties considered in this work, there always exists a deterministic, memoryless strategy in the solution set [15, 16, 29].

Figure 1: An example map for robotic planning in urban search and rescue missions. The robot aims to navigate to the victim (star) location with the shortest distance while minimizing the risk of bypassing (red) fire zones.

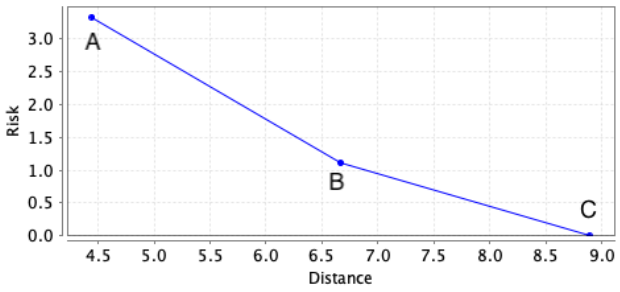


Figure 2: Pareto curve for multi-objective robotic planning. Purple, blue, and green routes in Figure 1 correspond to Pareto optimal points A, B and C, respectively.

$E^{\sigma}_{\mathcal{M}}(\boldsymbol{w}\cdot\boldsymbol{r}) \stackrel{\text{def}}{=} \sum_{i=1}^{n} w_i E^{\sigma}_{\mathcal{M}}(r_i)$ for any strategy $\sigma \in \Sigma_{\mathcal{M}}$. We say that a strategy σ^* is *optimal* with respect to ϕ and $\boldsymbol{w}$, if $E^{\sigma^*}_{\mathcal{M}}(\boldsymbol{w}\cdot\boldsymbol{r}) = \inf\{E^{\sigma}_{\mathcal{M}}(\boldsymbol{w}\cdot\boldsymbol{r}) \mid \sigma \in \Sigma_{\mathcal{M}}\}$. The strategy σ^* also corresponds to a Pareto optimal point for ϕ [16].

For simplicity, in this paper, we make the assumption that an MDP has a set of *end states*, which are reached with probability 1 under any strategy, and have zero reward and no outgoing transitions to other states. This simplifies our analysis by ensuring that the expected total reward is always finite. A variety of useful objectives for real-world applications can be encoded under these restrictions, for example, minimizing the distance, time, or incurred risk to complete a navigation task for robotic planning, or maximizing the safety and driver trust to complete a trip for autonomous driving.

Example 2.1. Figure 1 shows a map for urban search and rescue missions taken from the RoboCup Rescue Simulation Competition [1]. Consider a scenario where the robot aims to find an optimal route satisfying two objectives: (1) minimizing the travel distance to reach the rescue location, and (2) minimizing the risk of bypassing fire zones. We model the problem as an MDP where each road junction in the map is represented by an MDP state. In each state, the robot can move along the road with probability 0.9 and stay put with probability 0.1 due to noisy sensors. We define two reward functions dist and risk to measure the distance (i.e., the number of road blocks navigated) and the risk (i.e., the number of fire zones bypassed), respectively. Figure 2 shows the Pareto curve for the multi-objective property $\phi = ([\mathsf{dist}]_{\min}, [\mathsf{risk}]_{\min})$. The convex set of achievable values for ϕ includes any point on the Pareto curve and in the area above. There are three Pareto optimal points (A, B, C) corresponding to three deterministic, memoryless MDP strategies illustrated as purple, blue, and green routes in Figure 1, respectively. The rest of the Pareto curve (e.g., any point on the solid line between A and B, or the solid line between B and C) is achievable only if the robot takes randomized strategies.

3 UNCERTAIN HUMAN PREFERENCES

3.1 Formalization of Preferences

Preferences are often represented as weights reflecting humans' opinions about the relative importance of different criteria in multi-objective optimization [26]. Following this convention, we denote a *preference* over n objectives as a weight vector $\boldsymbol{w} = (w_1, \ldots, w_n) \in \mathbb{R}^n$ where $w_i \geq 0$ for $1 \leq i \leq n$ and $\sum_{i=1}^{n} w_i = 1$.

Such weight vectors can be obtained by eliciting human preferences in different ways. A naive approach is to ask for direct human input of weight values for objectives; however, it may be difficult for humans to come up with these values in practice. A popular preference elicitation method is pairwise comparison [34], in which humans answer queries such as: "Do you prefer objective i or objective j?" for each pair of objectives. We can then derive weights (e.g., via finding eigenvalues of pairwise comparison matrices) as described in [3, 9]. There are many other methods (e.g., Likert scaling, rating, ranking) for eliciting preferences weights, as surveyed in [26]. Eliciting preferences from the same person using various methods can yield different weight vectors as artifacts. In addition, if the controller synthesis needs to account for multiple human decision-makers' opinions, then a range of weight vectors can be resulted from eliciting multiple humans' preferences.

In order to capture the inherent uncertainty of human preferences, we define *uncertain human preferences* as an interval weight vector $\tilde{\boldsymbol{w}} = ([\underline{w}_1, \overline{w}_1], \cdots, [\underline{w}_n, \overline{w}_n])$, where $\underline{w}_i$ ($\overline{w}_i$) is the lower (upper) weight bound for objective i, and $0 \leq \underline{w}_i \leq \overline{w}_i \leq 1$. We say that a weight vector $\boldsymbol{w}$ belongs to an interval weight vector $\tilde{\boldsymbol{w}}$, denoted $\boldsymbol{w} \in \tilde{\boldsymbol{w}}$, if $\underline{w}_i \leq w_i \leq \overline{w}_i$ for all i. An interval weight vector comprises a convex set of weight vectors, providing a compact representation of uncertain human preferences.

Example 3.1. Suppose $\tilde{\boldsymbol{w}} = ([0.2, 0.7], [0.5, 0.9])$. Figure 3 shows a geometrical interpretation of uncertain human preferences represented by $\tilde{\boldsymbol{w}}$. We intersect each dashed line representing the lower or upper objective bounds with the solid line representing $w_1 + w_2 = 1$, and obtain a pair of weight vectors $(0.2, 0.8)$ and $(0.5, 0.5)$ corresponding to the extreme points of the feasible solution set. We highlight in red the range of all possible weight vectors that belong to $\tilde{\boldsymbol{w}}$, representing uncertain human preferences.

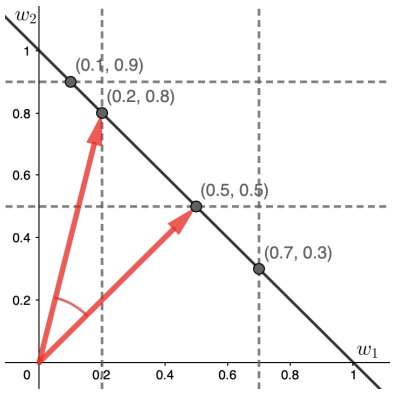


Figure 3: Geometrical interpretation of an interval weight vector $\tilde{\boldsymbol{w}} = ([0.2, 0.7], [0.5, 0.9])$ representing uncertain human preferences.

3.2 Multi-Strategy for MDPs

Recall from Section 2 that an optimal MDP strategy σ^* with respect to a multi-objective property ϕ and a weight vector $\boldsymbol{w}$ corresponds to a Pareto optimal point that optimizes the weighted sum of objectives. Thus, an interval weight vector $\tilde{\boldsymbol{w}}$ representing uncertain human preferences yields a set of Pareto optimal points and corresponding MDP strategies. We will use the notion of *multi-strategy* from *permissive controller synthesis* [10] to succinctly represent a set of strategies as follows.

A (deterministic, memoryless) multi-strategy for MDP $\mathcal{M}$ is a function $\theta : S \to 2^A$, defining a set of *allowed* actions $\theta(s) \subseteq \alpha(s)$ in each state $s \in S$. Let $\Theta_{\mathcal{M}}$ denote the set of all multi-strategies for $\mathcal{M}$. We say that a (deterministic, memoryless) strategy σ *complies* with multi-strategy θ, denoted $\sigma \lhd \theta$, if $\sigma(s) \in \theta(s)$ for all states $s \in S$. We require that $\theta(s) \neq \emptyset$ for any state s that is reachable under some strategy that complies with θ.

Given a reward predicate $[r]_{\sim b}$, we say that multi-strategy θ is *sound* with respect to $[r]_{\sim b}$ if $\mathcal{M}, \sigma \models [r]_{\sim b}$ for every strategy σ that complies with θ. We then say that a multi-strategy is sound for an uncertain set of human preferences if it is sound with respect to upper and lower bounds on each objective induced by a set of weight intervals. More precisely, given a multi-objective property $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$ and an interval weight vector $\tilde{\boldsymbol{w}}$, we say that multi-strategy θ is sound with respect to $\phi, \tilde{\boldsymbol{w}}$ if it is sound with respect to $[r_i]_{\geq \underline{b}_i}$ and $[r_i]_{\leq \overline{b}_i}$ for all i, where $\underline{b}_i = \inf\{x_i | \boldsymbol{x} = (x_1, \ldots, x_n) \in X_{\tilde{\boldsymbol{w}}}\}$, $\overline{b}_i = \sup\{x_i | \boldsymbol{x} = (x_1, \ldots, x_n) \in X_{\tilde{\boldsymbol{w}}}\}$, and $X_{\tilde{\boldsymbol{w}}}$ denotes the set of Pareto optimal points corresponding to $\tilde{\boldsymbol{w}}$. The intuition is that, due to convexity, any weight vector $\boldsymbol{w} \in \tilde{\boldsymbol{w}}$ must correspond to a Pareto optimal point within a space bounded by extreme points of $X_{\tilde{\boldsymbol{w}}}$. Later we develop Algorithm 1 in Section 4 to compute values of $\underline{b}_i$ and $\overline{b}_i$.

We quantify the *permissivity* of multi-strategy θ by measuring the degree of actions allowed in (reachable) MDP states. Let $\lambda(\theta) \stackrel{\text{def}}{=} \sum_{s \in S^\theta} (|\alpha(s)| - |\theta(s)|)$ be a *penalty function* where $S^\theta \subseteq S$ is the set of reachable states under θ. We say that a sound multi-strategy θ^* for $\mathcal{M}$ is *optimally permissive* if $\lambda(\theta^*) = \inf\{\lambda(\theta) \mid \theta \in \Theta_{\mathcal{M}}$ is sound with respect to ϕ and $\tilde{\boldsymbol{w}}\}$.

4 CONTROLLER SYNTHESIS APPROACH

4.1 Problem Statement

Given an MDP $\mathcal{M} = (S, s_0, A, \delta)$, a multi-objective property $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$, and an interval weight vector $\tilde{\boldsymbol{w}}$ representing uncertain human preferences, how can we synthesize an optimally permissive multi-strategy $\theta \in \Theta_{\mathcal{M}}$ that is sound with respect to ϕ and $\tilde{\boldsymbol{w}}$?

4.2 MILP-based Solution

We present a mixed-integer linear programming (MILP) based approach to solve the above problem. We use binary variables $\eta_{s,a} \in \{0, 1\}$ to encode whether a multi-strategy θ allows action $a \in \alpha(s)$ in state $s \in S$ of MDP $\mathcal{M}$. We use real-valued variables $\mu_{i,s}$ and $\nu_{i,s}$ to represent the minimal and maximal expected total reward for the ith objective from state s, under any strategy complying with θ. We set $\mu_{i,s} = \nu_{i,s} = 0$ for any end states in the MDP. The MILP encoding is:

$$\underset{\eta_{s,a} \in \{0,1\}, \mu_{i,s} \in \mathbb{R}, \nu_{i,s} \in \mathbb{R}}{\text{minimize}} \; c \cdot \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \eta_{s,a}) + \sum_{i=1}^{n} (\nu_{i,s_0} - \mu_{i,s_0}) \tag{1a}$$

subject to

$$\forall s \in S : \sum_{a \in \alpha(s)} \eta_{s,a} \leq c \cdot \sum_{(t,a) \in \rho(s)} \eta_{t,a} \tag{1b}$$

$$\forall s \in S : c \cdot \sum_{a \in \alpha(s)} \eta_{s,a} \geq \sum_{(t,a) \in \rho(s)} \eta_{t,a} \tag{1c}$$

$$\forall 1 \leq i \leq n, \forall s \in S, \forall a \in \alpha(s) : \mu_{i,s} \leq \sum_{t \in S} \delta(s,a)(t) \cdot \mu_{i,t} + r_i(s,a) + c \cdot (1 - \eta_{s,a}) \tag{1d}$$

$$\forall 1 \leq i \leq n, \forall s \in S, \forall a \in \alpha(s) : \nu_{i,s} \geq \sum_{t \in S} \delta(s,a)(t) \cdot \nu_{i,t} + r_i(s,a) - c \cdot (1 - \eta_{s,a}) \tag{1e}$$

$$\forall 1 \leq i \leq n : \mu_{i,s_0} \geq \underline{b}_i \tag{1f}$$

$$\forall 1 \leq i \leq n : \nu_{i,s_0} \leq \overline{b}_i \tag{1g}$$

where c is a large scaling constant[2] and we let $\rho(s) \stackrel{\text{def}}{=} \{(t, a) \mid \delta(t, a)(s) > 0$ and $t \neq s\}$ denote the set of incoming transitions to a state $s \in S$.

The objective function (1a) minimizes the total number of disallowed actions in all states plus the sum of expected total rewards over all objectives in the initial state. The latter serves as a tie-breaker between solutions with the same permissivity, favoring tighter reward bounds.

Constraints (1b) and (1c) enforce that no action is allowed for s if it is unreachable from any other state under the multi-strategy, and at least one action should be allowed otherwise. For the initial state s_0, we assume that there is always an allowed incoming transition, and $\sum_{(t,a) \in \rho(s_0)} \eta_{t,a} = 1$. Constraints (1d) and (1e) encode the recursion for expected rewards in each step, which are trivially satisfied

[2]Constant c is chosen to be larger than the expected total reward for any objective, from any state and under any objective.

Algorithm 1 Precomputing objective bounds

Input: An MDP $\mathcal{M}$, a multi-objective property ϕ, and an interval weight vector $\tilde{\boldsymbol{w}}$ for uncertain human preferences

Output: An interval vector $\boldsymbol{b} = ([\underline{b}_1, \overline{b}_1], \ldots, [\underline{b}_n, \overline{b}_n])$ for expected total reward bounds over n objectives

1: Initialize $\underline{b}_i = \infty$ and $\overline{b}_i = -\infty$ for $1 \leq i \leq n$
2: $W \leftarrow$ Find the set of extreme points of $\tilde{\boldsymbol{w}}$
3: **for all** weight vector $\boldsymbol{w} \in W$ **do**
4: $\quad \boldsymbol{x} = (x_1, \ldots, x_n) \leftarrow$ Find a Pareto optimal point for ϕ that corresponds to $\boldsymbol{w}$
5: $\quad$ **for** $1 \leq i \leq n$ **do**
6: $\quad\quad$ **if** $\underline{b}_i \geq x_i$ **then**
7: $\quad\quad\quad \underline{b}_i = x_i$
8: $\quad\quad$ **end if**
9: $\quad\quad$ **if** $\overline{b}_i \leq x_i$ **then**
10: $\quad\quad\quad \overline{b}_i = x_i$
11: $\quad\quad$ **end if**
12: $\quad$ **end for**
13: **end for**
14: **return** $\boldsymbol{b}$

when $\eta_{s,a} = 0$, that is, action a is disallowed in state s. Constraints (1f) and (1g) guarantee that, for each objective i, the expected total reward in the initial state under the multi-strategy satisfies the lower and upper bounds $\underline{b}_i$ and $\overline{b}_i$, which are precomputed using Algorithm 1.

Given an interval weight vector $\tilde{\boldsymbol{w}}$ representing uncertain human preferences, Algorithm 1 (line 2) first finds the set of extreme points in $\tilde{\boldsymbol{w}}$, denoted W. This can be done by applying standard methods for finding extreme points in a convex set [35]. Next, for each weight vector $\boldsymbol{w} \in W$, Algorithm 1 (line 4) finds a Pareto optimal point $\boldsymbol{x} = (x_1, \ldots, x_n)$ for ϕ, which yields the minimal expected total weighted reward sum under any strategy of the MDP $\mathcal{M}$. Here, we apply the value iteration-based method in [16] for the computation of Pareto optimal points. Finally, Algorithm 1 (line 3-13) loops through all weight vectors in W to determine the smallest lower bound $\underline{b}_i$ and the greatest upper bound $\overline{b}_i$ of the expected total reward for each objective i.

Example 4.1. We apply the proposed approach to synthesize an optimally permissive multi-strategy for MDP $\mathcal{M}$ modeled in Example 2.1 that is sound with respect to $\phi = ([\text{dist}]_{\min}, [\text{risk}]_{\min})$ and $\tilde{\boldsymbol{w}} = ([0.2, 0.7], [0.5, 0.9])$. Following Example 3.1, $\tilde{\boldsymbol{w}}$ gives a convex set of weight vectors with two extreme points $(0.5, 0.5)$ and $(0.2, 0.8)$. We also find out that weight vectors $(0.5, 0.5)$ and $(0.2, 0.8)$ correspond to Pareto optimal points B and C in Figure 2, respectively. Thus, applying Algorithm 1 yields an interval vector $\boldsymbol{b} = ([6.66, 8.89], [0, 1.12])$ for the expected total reward bounds over ϕ.

The MILP encoding minimizes $c \cdot \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \eta_{s,a}) + \sum_{i=1}^{2} (\nu_{i,s_0} - \mu_{i,s_0})$. We can select $c = 1000$ as the scaling factor constant in this example.

Constraints (1b) and (1c) are instantiated, for example, for the initial state s_0 as:

$$\eta_{s_0,\text{south}} + \eta_{s_0,\text{west}} \leq c$$
$$c \cdot (\eta_{s_0,\text{south}} + \eta_{s_0,\text{west}}) \geq 1$$

Figure 4: The synthesized multi-strategy in Example 4.1.

Constraints (1d) and (1e) are instantiated, for example, for the first objective $[\text{dist}]_{\min}$, state s_0, and action west as:

$$\mu_{1,s_0} \leq 0.9 \cdot \mu_{1,s_1} + 0.1 \cdot \mu_{1,s_0} + 1 + c \cdot (1 - \eta_{s_0,\text{west}})$$
$$\nu_{1,s_0} \geq 0.9 \cdot \nu_{1,s_1} + 0.1 \cdot \nu_{1,s_0} + 1 - c \cdot (1 - \eta_{s_0,\text{west}})$$

Constraints (1f) and (1g) are instantiated, for example, for the first objective $[\text{dist}]_{\min}$ as: $\mu_{1,s_0} \geq 6.66$ and $\nu_{1,s_0} \leq 8.89$.

The MILP encoding uses 15 binary variables to encode $\eta_{s,a}$, 44 real-valued variables to encode $\mu_{i,s}$ and $\nu_{i,s}$, and a total number of 90 constraints. It takes less than 1 second to solve the MILP problem using the Gurobi optimization toolbox [18]. The solution yields a multi-strategy as illustrated by the orange lines in Figure 4. The synthesized multi-strategy is sound with respect to ϕ and $\tilde{\boldsymbol{w}}$. There are two strategies complying with the multi-strategy, corresponding to Pareto optimal points B and C in Figure 2. The multi-strategy is also optimally permissive. Such a permissive multi-strategy could be useful in assisting humans' decision-making, by informing them about multiple allowable action choices in states. In addition, it offers flexibility for the system execution. If the robot finds that certain action cannot be executed due to the evolving environment (e.g., fire spreading), it may execute an alternative actions allowed by the multi-strategy while still guaranteeing soundness.

4.3 Correctness

The correctness of our proposed approach, with respect to the problem statement in Section 4.1, is stated below and the proof is given in the appendix.

Theorem 4.1. *Let $\mathcal{M}$ be an MDP, $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$ be a multi-objective property and $\tilde{\boldsymbol{w}}$ be an interval weight vector representing uncertain human preferences. There is a sound, optimally permissive multi-strategy θ in $\mathcal{M}$ with respect to ϕ and $\tilde{\boldsymbol{w}}$ whose permissive penalty is $\lambda(\theta)$, if and only if there is an optimal assignment to the MILP instance from (1a)-(1g) which satisfies $\lambda(\theta) = \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \eta_{s,a})$.*

Case Study			MDP Size		MILP Size			MILP Solution	
Name	Parameters	Preferences	#States	#Trans	#Binary	#Real	#Constraints	Time (s)	#Permissive States
uav	5	([0.1, 0.2], [0.8, 0.9]) ([0.1, 1], [0, 0.9])	28,401	40,373	29,897	113,604	176,394	2.2 7.8	1 1,496
	10	([0.1, 0.2], [0.8, 0.9]) ([0.1, 1], [0, 0.9])	56,901	80,873	59,897	227,604	353,394	5.4 20.2	1 2,996
	20	([0.1, 0.2], [0.8, 0.9]) ([0.1, 1], [0, 0.9])	113,901	161,873	119,897	455,604	707,394	15.3 43.8	1 5,996
taskgraph	30	([0.8, 1], [0, 0.2]) ([0.1, 1], [0, 0.9])	21,046	43,257	29,813	84,184	161,348	25.1 4.6	1,770 8,765
	40	([0.8, 1], [0, 0.2]) ([0.1, 1], [0, 0.9])	36,866	75,677	52,153	147,464	282,348	54.1 11.7	4,387 15,285
	50	([0.8, 1], [0, 0.2]) ([0.1, 1], [0, 0.9])	57,086	117,097	80,693	228,344	436,948	132.5 14.6	7,939 23,605
teamform	2	([0.8, 1], [0, 0.2]) ([0, 0.9], [0.1, 1])	1,847	2,288	2,191	7,388	12,462	1.9 2.3	146 189
	3	([0.8, 1], [0, 0.2]) ([0, 0.9], [0.1, 1])	12,475	15,228	14,935	49,900	84,694	timeout timeout	- -

Table 1: Experimental results illustrating performance of the proposed approach

4.4 Complexity Analysis

The size of an MILP problem is measured by the number of decision variables and the number of constraints. In the proposed MILP encoding, the number of binary variables is bounded by $O(|S| \cdot |A|)$, the number of real-valued variables is bounded by $O(n \cdot |S|)$, and the number of constraints is bounded by $O(n \cdot |S| \cdot |A|)$. MILP solvers work incrementally to synthesize a series of sound multi-strategies that are increasingly permissive. Therefore, we may stop early to accept a sound (but not necessarily optimally permissive) multi-strategy if computational resources are limited.

Prior to the MILP solution, we need to execute Algorithm 1, the most costly step of which is the computation of a Pareto optimal point in line 4. This is performed $|W|$ times, where $|W|$ is exponential in the number of objectives n. For each point, we compute a minimal weighted sum of expected total rewards for a given weight vector. This is done using the value iteration-based method of [16]. Value iteration does not have a meaningful time complexity, but is faster and more scalable than linear programming-based techniques in practice.

5 EXPERIMENTAL RESULTS

We have built a prototype implementation of the proposed approach, which uses the PRISM model checker [23] for computing Pareto optimal results of multi-objective synthesis in MDPs, and the Gurobi optimization toolbox [18] for solving MILP problems. The experiments were run on a laptop with a 2.8 GHz Quad-Core Intel Core i7 CPU and 16 GB RAM.

5.1 Case Studies

We applied our approach to three large case studies. [3]. For each case study, we used two interval weight vectors representing preferences with different uncertainty levels.

The first case study is adapted from [14], which considers the control of an unmanned aerial vehicle (UAV) that interacts with a human operator for road network surveillance, with a varying model parameter to count the operator's workload and fatigue level that may lead to degraded mission performance. The controller synthesis aims to balance two objectives of mission completion time and risk, based on the specified uncertain human preferences.

The second case study considers a task-graph scheduling problem inspired by [28]. The controller synthesis aims to compute an optimal schedule for a set of dependent tasks based on human preferences of different processors, with a varying model parameter of the digital clock counter.

The third case study models a team formation protocol [5] where a varying number of sensing agents cooperate to achieve certain tasks. The controller synthesis seeks to find an optimal schedule for these agents to meet the objectives of completing different tasks based on human preferences.

5.2 Results Analysis

Table 1 shows experimental results for these case studies. For each case study, we report the size of the MDP models in terms of the number of states and transitions, the size of the resulting MILP problems in terms of the number of decision variables (binary and real-valued) and constraints, the runtime for solving the MILP, and

[3]Files are available from: https://www.prismmodelchecker.org/files/iccps22

the number of permissive states (i.e., those with more than one allowed actions) in the synthesized controllers. We set a time-out of one hour for solving the MILP.

Unsurprisingly, the size of MILP problems increases with larger MDP models. But the results demonstrate that our approach can scale to large case studies. For example, it takes less than one minute to solve the resulting MILP problem of "uav 20" model with 113,901 MDP states, which includes 119,897 binary variables, 455,604 real-valued variables, and 707,394 constraints in the MILP. In most cases of "uav" and "taskgraph", a sound, optimally permissive multi-strategy is synthesized within one minute. However, the MILP solver failed to produce a feasible solution before time-out for some "teamform" cases, despite smaller MDP models than "uav" and "taskgraph". In addition, we observed that increasing the uncertainty level of preferences (i.e., larger intervals) leads to synthesized controllers with larger numbers of permissive states.

6 USER STUDY

We designed and conducted an online user study [4] to evaluate the synthesized controllers. We describe the study design in Section 6.1 and analyze the results in Section 6.2.

6.1 Study Design

Participants. We recruited 100 individuals with a categorical age distribution of 6 (18-24); 58 (25-34); 28 (35-49); 6 (50-64); and 1 (65+) using Amazon Mechanical Turk (AMT). To ensure data quality, our study recruitment criteria required that participants must be able to read English fluently and had performed at least 50 tasks previously with an above 90% approval rate on AMT. In addition, we injected attention check questions periodically during the study and rejected any response that failed attention checks.

Procedure. For each participant, we described the study purposes and asked them to consent to the study. After we asked about basic demographic information (e.g., age), the rest of the study consists of two phases: (i) eliciting human preferences, and (ii) evaluating the synthesized controllers.

First, we presented to each participant a grid map shown in Figure 5 and asked them to consider the planning problem for a robot to navigate from the start grid to the destination with three objectives: (1) minimizing the travel distance, (2) minimizing the risk encountered on route, and (3) maximizing the number of packages collected along the way. We used four different methods to elicit each participant's preferences over these objectives, including direct input of weight values (as illustrated in Figure 6a), Likert scaling (Figure 6b), pairwise comparison of objective names (Figure 6c), and pairwise comparison of optimal routes for individual objectives (Figure 6d). As described in Section 3, we can derive a weight vector over objectives from the results of each preference elicitation method. Thus, by aggregating these four weight vectors resulting from different elicitation methods, we obtained an interval weight vector to represent each participant's preferences.

Next, based on the elicited human preferences, we applied the proposed approach to synthesize optimal robotic controllers for three different grid maps (including Figure 5 and two other similar maps). We randomized the order of maps for different particiants

[4]This user study has obtained the Institutional Review Board (IRB) approval.

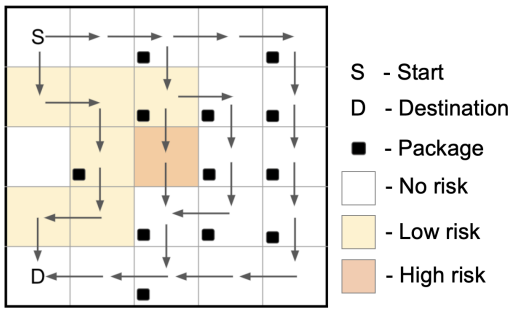
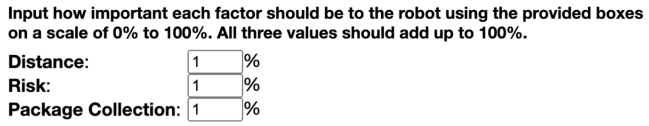
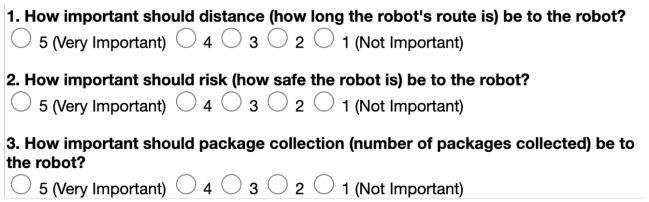


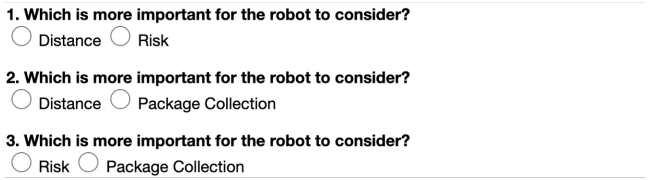
Figure 5: A grid map presented in the user study.



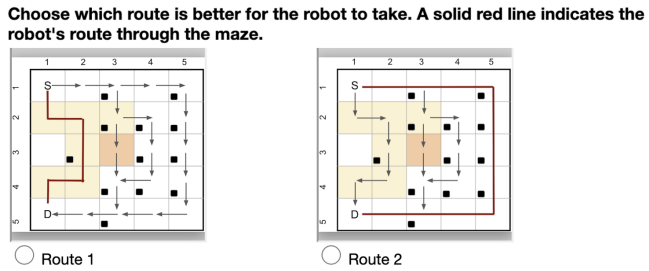
(a) Direct input of weight values.



(b) Likert scaling.



(c) Pairwise comparison of objective names.



(d) Pairwise comparison of routes that optimize individual objectives (e.g., route 1 for distance and route 2 for risk).

Figure 6: Four different methods for eliciting preferences.

to counterbalance the ordering confound effect. For each map, we asked participants a set of questions to evaluate the synthesized controllers. We describe the evaluation design, including manipulated factors, dependent measures, and hypotheses as follows.

Manipulated factors and dependent measures. We performed a within-subject experiment in which all participants were exposed to all evaluation conditions. We manipulated two independent factors: preferences and permissivity for the controller synthesis. For each map, we first presented a pair of MDP strategies (visualized as plans in the grid map) side by side: one is a sound strategy synthesized based on the elicited preferences, and the other is an arbitrary strategy unsound for preferences. Figure 7 shows the list of evaluation questions. We asked participants about their satisfaction and perceived accuracy of each plan. We also asked them to choose which plan they preferred.

Then, we presented side by side a strategy (visualized as a single route plan) and a multi-strategy (visualized as a possible multiple route plan), which are both synthesized based on the elicited preferences but with different degrees of permissivity. We asked participants to compare the synthesized strategy and multi-strategy in terms of favor ("Which route do you like better?"), informativity ("Which route provides more information?"), and satisfaction ("Which route are you more satisfied with?"). The exact questionnaire can be found in Figure 8.

Hypotheses. We made the following hypotheses based on the two manipulated factors.

Comparing strategies synthesized based on the elicited preferences and arbitrary strategies:

- **H1:** Preference-based strategies are more favorable than unsound arbitrary strategies.
- **H2:** Preference-based strategies are perceived as more accurate than unsound arbitrary strategies.
- **H3:** Preference-based strategies yield better satisfaction than unsound arbitrary strategies.

Comparing strategies and multi-strategies synthesized based on the elicited preferences:

- **H4:** Multi-strategies are more favorable than strategies.
- **H5:** Multi-strategies are perceived as more informative than strategies.
- **H6:** Multi-strategies yield better satisfaction than strategies.

6.2 Results Analysis

Comparing preference-based and arbitrary strategies. To evaluate hypothesis H1, we utilize a chi-squared test [32] to prove the statistical significance in the frequency of strategy selection, assuming an expected frequency of 50/50 to represent a random selection of strategies by users. We use an alpha value of 0.05 and thus retain a confidence level of 95% for our hypotheses. We assume a null hypothesis that the user selection of strategies will be random. We find that users favor preference-based strategies about 63% of the time overall (χ^2:α= 0.05, χ^2 = 21.33, CritVal = 3.84, p≤0.00001, Significant.); they choose preference-based strategies over arbitrary strategies more often for all three maps (71%, 59%, 60%). *Thus, the data supports H1.*

To evaluate hypotheses H2 and H3, shown in Figure 9 we employ one-way repeated measures ANOVA tests [32] to prove the statistical significance of the mean of all responses between preference-based strategies and arbitrary strategies. We use an alpha value of 0.05 and assume a null hypothesis that users will perceive preference accuracy and be satisfied with both strategies at a similar rate. We find that users rated preference-based strategies as significantly more accurate to their objective preferences (rANOVA:α= 0.05, F(1,598) = 74.71, p≤0.00001, Significant.). Figure 9 also shows that

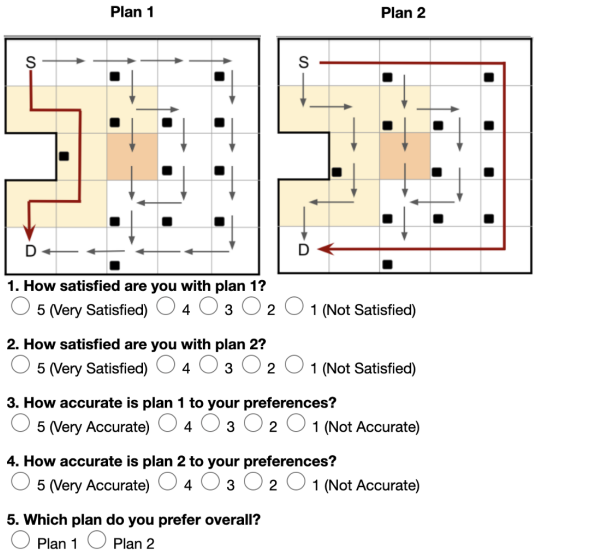


Figure 7: Evaluation of a synthesized strategy compared to an arbitrary strategy. Users were told these were possible robotic plans generated based on their input preferences, but not which plan was actually arbitrary.

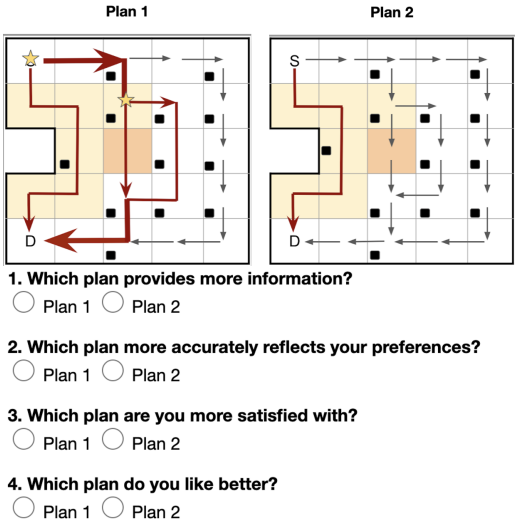


Figure 8: Evaluation comparison of a synthesized multi-strategy (plan 1) and a synthesized strategy (plan 2). Users were told these were possible robotic plans generated based on their input preferences. Stars indicate permissive states with multiple allowed actions.

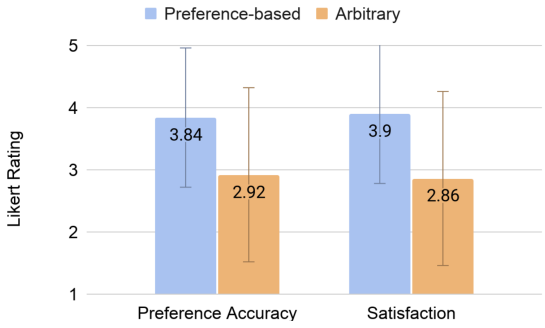


Figure 9: Mean and standard deviation of 5-point Likert ratings about perceived preference accuracy and user satisfaction for preference-based and arbitrary strategies.

Figure 10: Pairwise comparison of the synthesized strategies and multi-strategies regarding overall favor, informativity, and satisfaction.

users were significantly more satisfied with preference-based strategies than another arbitrary strategy through the plan (rANOVA:α= 0.05, F(1,598) = 105.28, p≤0.00001, Significant.). *Thus, the data supports H2 and H3.*

Comparing strategies and multi-strategies. We use chi-squared tests with an expected frequency of 50/50 and an alpha value of 0.05 to evaluate hypotheses H4, H5, and H6 with the study results shown in Figure 10.

Column 1 (Overall Favor) of Figure 10 shows that users do not significantly favor multi-strategies over less permissive strategies (χ^2:α= 0.05, χ^2 = 0.12, CritVal = 3.84, p≤0.729, Not Significant.), only slightly favoring multi-strategies to a single strategy counterpart. *Thus, the data rejects H4.*

Column 2 (Informativity) of Figure 10 shows users agreed about 71% of the time that multi-strategies provided them more information (χ^2:α= 0.05, χ^2 = 54.61, CritVal = 3.84, p≤0.00001, Significant.). *Thus, the data supports H5.*

Column 3 (Satisfaction) of Figure 10 shows users were more satisfied with multi-strategies 56% of the time (χ^2:α= 0.05, χ^2 = 4.32, CritVal = 3.84, p≤0.038, Significant.). *Thus, the data supports H6.*

Summary. We accept all hypotheses except H4 based on the statistical analysis. The user study results show that it is beneficial to synthesize strategies that account for human preferences. In addition, multi-strategies are more informative and yield better user satisfaction. However, sometimes less is more, participants do not always favor multi-strategies over strategies that are simpler to understand.

7 RELATED WORK

Human preferences. Mathematical models of human preferences have been studied broadly in the field of social choice theory [2]. There are many different representations of human preferences, for example, encoded as reward functions for robot trajectory planning [31] and deep reinforcement learning [6], or specified using temporal logics [25, 27]. In the context of multi-objective optimization [26], preferences are represented as weights indicating the relative importance of objectives. Optimization methods can vary depending on when and how humans articulate their preferences. Humans can indicate their preferences *a priori* before running the optimization algorithm, they can progressively provide input during the optimization process, or they can select *a posteriori* a solution point from a set of Pareto optimal results. Our work considers *a priori* elicitation of human preferences represented as weights for multiple objectives.

Multi-objective controller synthesis for MDPs. Multi-objective optimization has been well-studied in operation research and engineering [26, 30]. In recent years, multi-objective optimization for MDPs has been considered from a formal methods perspective [4, 8, 11, 15, 16, 20], which presents theories and algorithms for verifying multi-objective properties, synthesizing strategies, and approximating Pareto curves. More recently, such techniques have been applied to multi-objective robot path planning [24] and multi-objective controller synthesis for autonomous systems that account for human operators' workload and fatigue levels [14]. However, existing work does not account for the uncertainty of human preferences in the relative importance of objectives.

There is a line of work (e.g., [7, 19, 37]) considering uncertain MDPs where transition probabilities and rewards are represented as an uncertain set of parameters or intervals. Our work is different in the sense that we consider the uncertainty in human preferences of different objectives.

Our proposed approach is based on mixed-integer linear programming (MILP). There exist several MILP-based solutions to compute counterexamples and witnesses for MDPs [12, 13, 17, 36]. However, these methods are not directly applicable for controller synthesis which is a different problem. The most relevant work is [10] that presents an MILP-based method for permissive controller synthesis of probabilistic systems. As discussed in Section 1, our approach is inspired by [10] but has several key differences (e.g., [10] does not consider the controller soundness with respect to multi-objective properties and human preferences).

8 CONCLUSION

In this paper, we developed a novel approach that accounts for uncertain human preferences in the multi-objective controller synthesis for MDPs. The proposed MILP-based approach synthesizes a sound, optimally permissive multi-strategy with respect to a multi-objective property and an uncertain set of human preferences. We

implemented and evaluated the proposed approach on three large case studies. Experimental results show that our approach can be successfully applied to synthesize sound, optimally permissive multi-strategies with varying MDP model size and uncertainty level of human preferences. In addition, we designed and conducted an online user study with 100 participants using Amazon Mechanical Turk, which shows statistically significant results about user satisfaction of the synthesized controllers.

There are several directions to explore for possible future work. First, we will extend our approach for a richer set of multi-objective properties beyond expected total rewards, such as the temporal logic-based multi-objective properties considered in [15]. Second, we will extend our approach for a variety of probabilistic models beyond MDPs, such as stochastic games and POMDPs. Last but not least, we will apply our approach to a wider range of real-world CPS applications (e.g., autonomous driving, smart cities).

ACKNOWLEDGMENTS

This work was supported in part by National Science Foundation grants CCF-1942836 and CNS-1755784, and European Research Council under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 834115, FUN2MODEL). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the grant sponsors.

REFERENCES

[1] H Levent Akin, Nobuhiro Ito, Adam Jacoff, Alexander Kleiner, Johannes Pellenz, and Arnoud Visser. 2013. Robocup rescue robot and simulation leagues. *AI magazine* 34, 1 (2013), 78–78.

[2] Kenneth J Arrow. 2012. *Social choice and individual values*. Vol. 12. Yale university press.

[3] Jonathan Barzilai. 1997. Deriving weights from pairwise comparison matrices. *Journal of the operational research society* 48, 12 (1997), 1226–1232.

[4] Krishnendu Chatterjee, Rupak Majumdar, and Thomas A Henzinger. 2006. Markov decision processes with multiple objectives. In *Annual symposium on theoretical aspects of computer science*. Springer, 325–336.

[5] Taolue Chen, Marta Kwiatkowska, David Parker, and Aistis Simaitis. 2011. Verifying team formation protocols with probabilistic model checking. In *International Workshop on Computational Logic in Multi-Agent Systems*. Springer, 190–207.

[6] Paul F Christiano, Jan Leike, Tom B Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 4302–4310.

[7] Murat Cubuktepe, Nils Jansen, Sebastian Junges, Joost-Pieter Katoen, and Ufuk Topcu. 2020. Scenario-based verification of uncertain mdps. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 287–305.

[8] Florent Delgrange, Joost-Pieter Katoen, Tim Quatmann, and Mickael Randour. 2020. Simple strategies in multi-objective MDPs. In *Proc. 26th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'20) (LNCS, Vol. 12078)*. Springer, 346–364.

[9] Theo K Dijkstra. 2013. On the extraction of weights from pairwise comparison matrices. *Central European Journal of Operations Research* 21, 1 (2013), 103–123.

[10] Klaus Dräger, Vojtěch Forejt, Marta Kwiatkowska, David Parker, and Mateusz Ujma. 2015. Permissive Controller Synthesis for Probabilistic Systems. *Logical Methods in Computer Science* 11, 2 (2015).

[11] Kousha Etessami, Marta Kwiatkowska, Moshe Y Vardi, and Mihalis Yannakakis. 2007. Multi-objective model checking of Markov decision processes. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 50–65.

[12] Lu Feng, Mahsa Ghasemi, Kai-Wei Chang, and Ufuk Topcu. 2018. Counterexamples for robotic planning explained in structured language. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 7292–7297.

[13] Lu Feng, Laura Humphrey, Insup Lee, and Ufuk Topcu. 2016. Human-interpretable diagnostic information for robotic planning systems. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1673–1680.

[14] Lu Feng, Clemens Wiltsche, Laura Humphrey, and Ufuk Topcu. 2015. Controller synthesis for autonomous systems interacting with human operators. In *Proceedings of the acm/ieee sixth international conference on cyber-physical systems*. 70–79.

[15] Vojtěch Forejt, Marta Kwiatkowska, Gethin Norman, David Parker, and Hongyang Qu. 2011. Quantitative multi-objective verification for probabilistic systems. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 112–127.

[16] Vojtěch Forejt, Marta Kwiatkowska, and David Parker. 2012. Pareto curves for probabilistic model checking. In *International Symposium on Automated Technology for Verification and Analysis*. Springer, 317–332.

[17] Florian Funke, Simon Jantsch, and Christel Baier. 2020. Farkas certificates and minimal witnesses for probabilistic reachability constraints. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, 324–345.

[18] LLC Gurobi Optimization. 2021. Gurobi Optimizer Reference Manual.

[19] Ernst Moritz Hahn, Vahid Hashemi, Holger Hermanns, Morteza Lahijanian, and Andrea Turrini. 2017. Multi-objective robust strategy synthesis for interval Markov decision processes. In *International Conference on Quantitative Evaluation of Systems*. Springer, 207–223.

[20] A. Hartmanns, S. Junges, J.-P. Katoen, and T. Quatmann. 2020. Multi-cost bounded tradeoff analysis in MDP. *Journal of Automated Reasoning* 64, 7 (2020), 1483–1522.

[21] John G Kemeny, J Laurie Snell, and Anthony W Knapp. 2012. *Denumerable Markov chains*. Vol. 40. Springer Science & Business Media.

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

[23] Marta Kwiatkowska, Gethin Norman, and David Parker. 2011. PRISM 4.0: Verification of probabilistic real-time systems. In *International conference on computer aided verification*. Springer, 585–591.

[24] Bruno Lacerda, David Parker, and Nick Hawes. 2017. Multi-objective policy generation for mobile robots under probabilistic time-bounded guarantees. In *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 27.

[25] Meilun Li, Zhikun She, Andrea Turrini, and Lijun Zhang. 2015. Preference planning for markov decision processes. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[26] R Timothy Marler and Jasbir S Arora. 2004. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization* 26, 6 (2004), 369–395.

[27] Noushin Mehdipour, Cristian-Ioan Vasile, and Calin Belta. 2020. Specifying user preferences using weighted signal temporal logic. *IEEE Control Systems Letters* 5, 6 (2020), 2006–2011.

[28] Gethin Norman, David Parker, and Jeremy Sproston. 2013. Model Checking for Probabilistic Timed Automata. *Formal Methods in System Design* 43, 2 (2013), 164–190.

[29] Martin L Puterman. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc.

[30] Bernard Roy and Philippe Vincke. 1981. Multicriteria analysis: survey and new directions. *European journal of operational research* 8, 3 (1981), 207–218.

[31] Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. 2017. Active Preference-Based Learning of Reward Functions.. In *Robotics: Science and Systems*.

[32] Howard J Seltman. 2012. *Experimental design and analysis*. Carnegie Mellon University Pittsburgh.

[33] Sanjit A Seshia, Shiyan Hu, Wenchao Li, and Qi Zhu. 2016. Design automation of cyber-physical systems: Challenges, advances, and opportunities. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 36, 9 (2016), 1421–1434.

[34] Louis L Thurstone. 1927. A law of comparative judgment. *Psychological review* 34, 4 (1927), 273.

[35] François Treves. 2016. *Topological Vector Spaces, Distributions and Kernels: Pure and Applied Mathematics, Vol. 25*. Vol. 25. Elsevier.

[36] Ralf Wimmer, Nils Jansen, Erika Ábrahám, Joost-Pieter Katoen, and Bernd Becker. 2014. Minimal counterexamples for linear-time probabilistic verification. *Theoretical Computer Science* 549 (2014), 61–100.

[37] Eric M Wolff, Ufuk Topcu, and Richard M Murray. 2012. Robust control of uncertain Markov decision processes with temporal logic specifications. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 3372–3379.

A PROOFS

Here, we prove the correctness of our MILP encoding, as stated in Theorem 4.1. This result adapts and extends the proof for the MILP encoding given in [10] (specifically the case for what are called *static* penalty schemes and deterministic multi-strategies). We require the following auxiliary lemma, where $E^{\sigma}_{\mathcal{M},s}(r_i)$ denotes the expected total reward for reward structure r_i under strategy σ of MDP $\mathcal{M}$, from a particular starting state s.

Lemma A.1. *Let $\mathcal{M} = (S, s_0, A, \delta)$ be an MDP, $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$ be a multi-objective property, and θ be a multi-strategy. Consider the inequalities for $s \in S$, $1 \leq i \leq n$:*

$$\mu_{i,s} \leq \min_{a \in \theta(s)} \sum_{t \in S} \delta(s,a)(t) \cdot \mu_{i,t} + r_i(s,a)$$

$$\nu_{i,s} \geq \max_{a \in \theta(s)} \sum_{t \in S} \delta(s,a)(t) \cdot \mu_{i,t} + r_i(s,a)$$

Then values $\hat{\mu}_{i,s}, \hat{\nu}_{i,s} \in \mathbb{R}$, for $s \in S$ and $1 \leq i \leq n$, are a solution to the above inequalities if an only if $\hat{\mu}_{i,s} = \inf_{\sigma \lhd \theta} E^{\sigma}_{\mathcal{M},s}(r_i)$ and $\hat{\nu}_{i,s} = \sup_{\sigma \lhd \theta} E^{\sigma}_{\mathcal{M},s}(r_i)$.

Proof. The above follows from standard results on the solution of MDPs [29], noting that there is a separate set of inequalities for each μ_i and ν_i and $1 \leq i \leq n$. This also relies on our assumption (see Section 4.1) that a designated set of zero-reward end states is always reached with probability 1, ensuring that expected total rewards are finite and removing the need to deal with zero-reward loops (whereas [10] deals with the latter through additional MILP variables and constraints). □

Theorem 4.1. *Let $\mathcal{M}$ be an MDP, $\phi = ([r_1]_{\min}, \ldots, [r_n]_{\min})$ be a multi-objective property and $\tilde{\boldsymbol{w}}$ be an interval weight vector representing uncertain human preferences. There is a sound, optimally permissive multi-strategy θ in $\mathcal{M}$ with respect to ϕ and $\tilde{\boldsymbol{w}}$ whose permissive penalty is $\lambda(\theta)$, if and only if there is an optimal assignment to the MILP instance from (1a)-(1g) which satisfies $\lambda(\theta) = \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \eta_{s,a})$.*

Proof. We prove that: (1) every multi-strategy θ induces a satisfying assignment to the MILP such that the permissive penalty $\lambda(\theta) = \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \eta_{s,a})$, and (2) vice versa.

Direction (1). We start by proving that, given a sound multi-strategy θ, we can construct a satisfying assignment $\{\hat{\eta}_{s,a}, \hat{\mu}_{i,s}, \hat{\nu}_{i,s}\}_{s \in S, a \in A, 1 \leq i \leq n}$ to the MILP constraints. For $s \in S$ and $a \in \alpha(s)$, we set $\hat{\eta}_{s,a} = 1$ if s is a reachable state under θ and $a \in \theta(s)$; otherwise, we set $\hat{\eta}_{s,a} = 0$.

Thus, the permissive penalty $\lambda(\theta)$ that counts the total number of disallowed actions in reachable states under θ equals to $\sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \hat{\eta}_{s,a})$. Constraints (1b) and (1c) are satisfied for all unreachable states, because both sides of the inequalities are zero. For reachable states, constraint (1b) is trivially satisfied if the scaling factor c is large enough; constraint (1c) is also satisfied, because a reachable state under strategy θ should have at least one allowed action.

We set $\hat{\mu}_{i,s} = \inf_{\sigma \lhd \theta} E^{\sigma}_{\mathcal{M},s}(r_i)$ and $\hat{\nu}_{i,s} = \sup_{\sigma \lhd \theta} E^{\sigma}_{\mathcal{M},s}(r_i)$. Constraints (1d) and (1e) are satisfied for $a \in \theta(s)$ thanks to Lemma A.1. If $a \notin \theta(s)$, then (1d) and (1e) are also trivially satisfied because $\hat{\eta}_{s,a} = 0$. By the soundness definition, we have $\hat{\mu}_{i,s_0} \geq \underline{b}_i$ and $\hat{\nu}_{i,s_0} \leq \overline{b}_i$. This gives the satisfaction of constraints (1f) and (1g).

Direction (2). Given a satisfying MILP assignment $\{\hat{\eta}_{s,a}, \hat{\mu}_{i,s}, \hat{\nu}_{i,s}\}_{s \in S, a \in A, 1 \leq i \leq n}$, we construct θ for $\mathcal{M}$ by putting $\theta(s) = \{a \in \alpha(s) \mid \hat{\eta}_{s,a} = 1\}$ for all $s \in S$. Thanks to constraints (1d) and (1e), and Lemma A.1, we have that $\hat{\mu}_{i,s} = \inf_{\sigma \lhd \theta} E^{\sigma}_{\mathcal{M},s}(r_i)$ and $\hat{\nu}_{i,s} = \sup_{\sigma \lhd \theta} E^{\sigma}_{\mathcal{M},s}(r_i)$ for each $1 \leq i \leq n$. Using also constraints (1f) and (1g), we have that, for any strategy $\sigma \lhd \theta$, $E^{\sigma}_{\mathcal{M},s_0}(r_i) \geq \hat{\mu}_{i,s_0} \geq \underline{b}_i$ and $E^{\sigma}_{\mathcal{M},s_0}(r_i) \leq \hat{\nu}_{i,s_0} \leq \overline{b}_i$; thus the multi-strategy θ is sound with respect to ϕ and $\tilde{\boldsymbol{w}}$. As in the reverse direction above, the permissiveness of θ is $\lambda(\theta) = \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \hat{\eta}_{s,a})$, taken from the objective function of the MILP. □