Multi-Objective Controller Synthesis 
with Uncertain Human Preferences

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

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

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

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 non-deterministic 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.
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 permis-

siveness 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 stochastic games (of which MDPs are a special case). However, an MILP-based method for synthesizing permissive strategies in stochastic games (of which MDPs are a special case). However, an MILP-based method for synthesizing permissive strategies in stochastic games (of which MDPs are a special case). However, an MILP-based method for synthesizing permissive strategies in stochastic games (of which MDPs are a special case).

In this work, we focus on deterministic, memoryless strategies. Thus, we can simplify the definition of strategy to a function \( \sigma : S \times A \rightarrow A \).

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 \( M = (S, a, \delta, \gamma) \), where \( S \) is a finite set of states, \( a \in S \) is an initial state, \( A \) is a set of actions, and \( \delta : S \times A \rightarrow \text{Dist}(S) \) is a probabilistic transition function with \( \text{Dist}(S) \) denoting the set of probability distributions over \( S \). Each state \( s \in S \) has a set of enabled actions, given by \( a(s) = \{ a \in A | \delta(s, a) \text{ is defined} \} \). A path through \( M \) is a sequence \( \pi = s_0a_1s_1 \ldots \) where \( a_i \in a(s_i) \) 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 \( FPaths(Paths) \) denote the set of finite (infinite) paths through \( M \).

A strategy (also called a policy) is a function \( \sigma : FPaths \rightarrow \text{Dist}(A) \) that resolves the nondeterministic choice of actions in each state based on the execution history. A strategy \( \sigma \) is deterministic if \( \sigma(\pi) \) is a point distribution for all \( \pi \), and randomized otherwise. A strategy \( \sigma \) is memoryless if the action choice \( \sigma(\pi) \) depends only on the last state of \( \pi \). In this work, we focus on deterministic, memoryless strategies. Thus, we can simplify the definition of strategy to a function \( \sigma : S \times A \rightarrow A \).

A reward function of \( M \) takes the form \( r : S \times A \rightarrow \mathbb{R} \). The total reward along an infinite path \( \pi = s_0a_1s_1 \ldots \) is given by \( r(\pi) = \sum_{i=0}^{\infty} r(s_i, a_i) \). The expected total reward for \( M \) under a strategy \( \sigma \) is denoted by \( E_M^\sigma(r) = \int r(\pi) dP_M^\sigma \). We say that \( M \) under strategy \( \sigma \) satisfies a reward predicate \( [r]_{\leq b} \) if the expected total reward \( E_M^\sigma(r) \leq b \). A reward predicate \( [r]_{\geq b} \) is satisfiable in MDP \( M \) if there exists a strategy \( \sigma \in \Sigma_M \) such that \( M, \sigma \models [r]_{\geq b} \).

A multi-objective property \( \phi = ([r_1]_{\leq b_1}, \ldots, [r_n]_{\min}) \), where \( b_\min \in [\min, \max] \), 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 \( \phi = ([r_1]_{\min}, \ldots, [r_n]_{\max}) \). A maximizing objective \([r]_{\min} \) can be converted to a minimizing objective by negating rewards. Checking \( \phi \) on MDP \( M \) yields a set of Pareto optimal points that lie on the boundary of the set of achievable values:

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

We say that a point \( x^* = (x_1^*, \ldots, x_n^*) \in X \) is Pareto optimal if there does not exist another point \( x = (x_1, \ldots, x_n) \in X \) 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 \( M \) if there exists a strategy \( \sigma \in \Sigma_M \) such that \( M, \sigma \models [r]_{\leq x_i} \) for all \( i \). The set of achievable values \( X \) for \( \phi \) is convex [16].

Given a multi-objective property \( \phi = ([r_1]_{\min}, \ldots, [r_n]_{\max}) \) and a weight vector \( \mathbf{w} \in \mathbb{R}^n \), the expected total weighted reward sum is
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 $\phi = (\{\text{dist}\}_{\text{min}}, \{\text{risk}\}_{\text{min}})$. The convex set of achievable values for $\phi$ 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 $w = (w_1, \ldots, w_n) \in \mathbb{R}^n$ 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 $\tilde{w} = ([w_{i,1}, \overline{w}_i], \cdots, [w_{n,1}, \overline{w}_n])$, 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 $w$ belongs to an interval weight vector $\tilde{w}$, denoted $w \in \tilde{w}$, 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 $\tilde{w} = ([0.2, 0.7], [0.5, 0.9])$. Figure 3 shows a geometrical interpretation of uncertain human preferences represented by $\tilde{w}$. 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 $\tilde{w}$, representing uncertain human preferences.
3.2 Multi-Strategy for MDPs

Recall from Section 2 that an optimal MDP strategy \( \sigma^* \) with respect to a multi-objective property \( \phi \) and a weight vector \( \psi \) corresponds to a Pareto optimal point that optimizes the weighted sum of objectives. Thus, an interval weight vector \( \psi \) representing uncertain human preferences yields a set of Pareto optimal points corresponding to the set of multi-strategies \( \Theta_M \) that is sound with respect to \( \phi \) and \( \psi \).

A (deterministic, memoryless) multi-strategy for MDP \( M \) is a function \( \theta : S \rightarrow 2^A \), defining a set of allowed actions \( \theta(s) \subseteq \alpha(s) \) in each state \( s \in S \). Let \( \Theta_M \) denote the set of all multi-strategies for \( M \). We say that a (deterministic, memoryless) strategy \( \sigma \) complies with multi-strategy \( \theta \), denoted \( \sigma \models \theta \), 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 \( \theta \).

Given a reward predicate \( [r]_{\leq b} \), we say that multi-strategy \( \theta \) is sound with respect to \( [r]_{\leq b} \) if \( M, \sigma \models [r]_{\leq b} \) for every strategy \( \sigma \) that complies with \( \theta \). 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 \( \phi = ([r_1]_{\min} \cdots [r_n]_{\min}) \) and an interval weight vector \( \psi \), we say that multi-strategy \( \theta \) is sound with respect to \( \phi \cdot \psi \) if it is sound with respect to \( [r_i]_{\geq h_i} \) and \( [r_i]_{\leq l_i} \) for all \( i \), where \( b_i = \inf\{x_i | x = (x_1, \ldots, x_n) \in X'_\psi \}, \quad b_i = \sup\{x_i | x = (x_1, \ldots, x_n) \in X'_\psi \}, \quad \) and \( X'_\psi \) denotes the set of all points corresponding to \( \psi \). The intuition is that, due to convexity, any weight vector \( \psi \in \bar{\psi} \) must correspond to a Pareto optimal point within a space bounded by extreme points of \( X'_\psi \). Later we develop Algorithm 1 in Section 4 for computing values of \( b_i \) and \( \bar{b}_i \).

We quantify the permissivity of multi-strategy \( \theta \) by measuring the degree of actions allowed in (reachable) MDP states. Let \( \lambda(\theta) \deq \sum_{s \in S} (|\alpha(s)| - |\theta(s)|) \) be a penalty function where \( S^0 \subseteq S \) is the set of reachable states under \( \theta \). We say that a sound multi-strategy \( \theta^* \) for \( M \) is optimally permissive if \( \lambda(\theta^*) = \inf\{\lambda(\theta) | \theta \in \Theta_M \} \) is sound with respect to \( \phi \) and \( \psi \).

4 CONTROLLER SYNTHESIS APPROACH

4.1 Problem Statement

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

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 \( \theta \) allows action \( a \in \alpha(s) \) in state \( s \in S \) of MDP \( M \). We use real-valued variables \( \mu_{s,a} \) and \( v_{l,s} \) to represent the minimal and maximal expected total reward for the \( i \)th objective from state \( s \), under any strategy complying with \( \theta \). We set \( \mu_{s,a} = v_{l,s} = 0 \) for any end states in the MDP. The MILP encoding is:

\[
\begin{align*}
\text{minimize} \quad & c \cdot \sum_{s \in S} \sum_{a \in \alpha(s)} (1 - \eta_{s,a}) \\
\text{subject to} \quad & \forall s \in S : \sum_{a \in \alpha(s)} \eta_{s,a} \leq c \cdot \sum_{(t,a) \in \rho(s)} \eta_{t,a} \quad (1b) \\
& \forall s \in S : c \cdot \sum_{a \in \alpha(s)} \eta_{s,a} \geq \sum_{(t,a) \in \rho(s)} \eta_{t,a} \quad (1c) \\
& \forall 1 \leq i \leq n, \forall s \in S, \forall a \in \alpha(s) : \\
& \mu_{s,a} \leq \sum_{t \in S} \delta(s,a)(t) \cdot \mu_{t,a} + r_t(s,a) + c \cdot (1 - \eta_{s,a}) \quad (1d) \\
& \forall 1 \leq i \leq n, \forall s \in S, \forall a \in \alpha(s) : \\
& v_{l,s} \geq \sum_{t \in S} \delta(s,a)(t) \cdot v_{l,t} + r_t(s,a) - c \cdot (1 - \eta_{s,a}) \quad (1e) \\
& \forall 1 \leq i \leq n : \mu_{s,a} \geq b_i \\
& \forall 1 \leq i \leq n : v_{l,s} \leq \bar{b}_i \quad (1g)
\end{align*}
\]

where \( c \) is a large scaling constant\(^2\) and we let \( \rho(s) \deq \{(t,a) | \delta(t,a)(s) > 0 \} \neq s \) 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 $M$, a multi-objective property $\phi$, and an interval weight vector $\hat{w}$ for uncertain human preferences

**Output:** An interval vector $\hat{b} = \{(\hat{b}_1, \hat{b}_1), \ldots, (\hat{b}_n, \hat{b}_n)\}$ for expected total reward bounds over $n$ objectives

1. Initialize $\hat{b}_i = \infty$ and $\hat{b}_i = -\infty$ for $1 \leq i \leq n$
2. $W \leftarrow$ Find the set of extreme points of $\hat{w}$
3. for all weight vector $w \in W$ do
   4. $x = (x_1, \ldots, x_n) \leftarrow$ Find a Pareto optimal point for $\phi$ that corresponds to $w$
   5. for $1 \leq i \leq n$ do
      6. if $\hat{b}_i \geq x_i$ then
      7. $\hat{b}_i \leftarrow x_i$
      end if
      8. if $\hat{b}_i \leq x_i$ then
      9. $\hat{b}_i \leftarrow x_i$
     end if
    end for
   end for
14. return $\hat{b}$

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 $\hat{b}_i$ and $\hat{b}_i$, which are precomputed using Algorithm 1.

Given an interval weight vector $\hat{w}$ representing uncertain human preferences, Algorithm 1 (line 2) first finds the set of extreme points in $\hat{w}$, denoted $W$. This can be done by standard methods for finding extreme points in a convex set [35]. Next, for each weight vector $w \in W$, Algorithm 1 (line 4) finds a Pareto optimal point $x = (x_1, \ldots, x_n)$ for $\phi$, which yields the minimal expected total weighted reward sum under any strategy of the MDP $M$. 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 $\hat{b}_i$ and the greatest upper bound $\hat{b}_i$ 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 $M$ modeled in Example 2.1 that is sound with respect to $\phi$ and $\hat{w}$, which is given in the appendix.

The MILP encoding minimizes $c \cdot \sum_{s \in S} \sum_{a \in \Omega(s)} (1 - \eta_{s,a}) + \sum_{i=1}^n (\nu_{s,0} - \mu_{s,0})$. 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:

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

Constraints (1d) and (1e) are instantiated, for example, for the first objective $\eta_{s_0,\text{description}}$ and action $a$ as:

$$\eta_{s_0,\text{description}} \leq \nu_{s_0,\text{description}} + \eta_{s_0,\text{description}} + c \cdot (1 - \eta_{s_0,\text{description}})$$

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

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 $M$ be an MDP, $\phi = \{[r_1]_{\text{min}}, \ldots, [r_n]_{\text{min}}\}$ be a multi-objective property and $\hat{w}$ be an interval weight vector representing uncertain human preferences. There is a sound, optimally permissive multi-strategy $\theta$ in $M$ with respect to $\phi$ and $\hat{w}$ whose permissive penalty is $\lambda(\theta)$, if and only if there is an optimal assignment to the MILP instance (1a)-(1g) which satisfies $\lambda(\theta) = \sum_{s \in S} \sum_{a \in \Omega(s)} (1 - \eta_{s,a})$. 
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 the number of permissive states.

<table>
<thead>
<tr>
<th>Case Study</th>
<th>MDP Size</th>
<th>MILP Size</th>
<th>MILP Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#States</td>
<td>#Trans</td>
<td>#Binary</td>
</tr>
<tr>
<td>Name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uav</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.1, 0.2), (0.8, 0.9)</td>
<td>28,401</td>
<td>40,373</td>
<td>29,897</td>
</tr>
<tr>
<td>(0.1, 1), (0.0, 0.9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.1, 0.2), (0.8, 0.9)</td>
<td>56,901</td>
<td>80,873</td>
<td>59,897</td>
</tr>
<tr>
<td>(0.1, 1), (0.0, 0.9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.1, 0.2), (0.8, 0.9)</td>
<td>113,901</td>
<td>161,873</td>
<td>119,897</td>
</tr>
<tr>
<td>(0.1, 1), (0.0, 0.9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>taskgraph</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.8, 1), (0.0, 0.2)</td>
<td>21,046</td>
<td>43,257</td>
<td>29,813</td>
</tr>
<tr>
<td>(0.1, 1), (0.0, 0.9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.8, 1), (0.0, 0.2)</td>
<td>36,866</td>
<td>75,677</td>
<td>52,153</td>
</tr>
<tr>
<td>(0.1, 1), (0.0, 0.9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.8, 1), (0.0, 0.2)</td>
<td>57,086</td>
<td>117,097</td>
<td>80,693</td>
</tr>
<tr>
<td>(0.1, 1), (0.0, 0.9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>teamform</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.8, 1), (0.0, 0.2)</td>
<td>1,847</td>
<td>2,288</td>
<td>2,191</td>
</tr>
<tr>
<td>(0.0, 0.9), (0.1, 1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(0.8, 1), (0.0, 0.2)</td>
<td>12,475</td>
<td>15,228</td>
<td>14,935</td>
</tr>
<tr>
<td>(0.0, 0.9), (0.1, 1)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Experimental results illustrating performance of the proposed approach

\(^3\) 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 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 participants.

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.

This user study has obtained the Institutional Review Board (IRB) approval.
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 ($\chi^2_{\alpha=0.05} = 21.33$, CritVal $= 3.84$, $p \leq 0.00001$, Significant.). 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; $\alpha = 0.05$, $F(1,598) = 74.71$, $p \leq 0.00001$, Significant.). Figure 9 also shows that
users were significantly more satisfied with preference-based strategies than another arbitrary strategy through the plan (tANOVA: $\alpha=0.05$, $F(1,598) = 105.28$, $p \leq 0.00001$, 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 ($\chi^2: \alpha=0.05$, $\chi^2 = 0.12$, CritVal = 3.84, $p \leq 0.729$, 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 ($\chi^2: \alpha=0.05$, $\chi^2 = 54.61$, CritVal = 3.84, $p \leq 0.00001$, Significant.). Thus, the data supports H5.

Column 3 (Satisfaction) of Figure 10 shows users were more satisfied with multi-strategies 56% of the time ($\chi^2: \alpha=0.05$, $\chi^2 = 4.32$, CritVal = 3.84, $p \leq 0.038$, 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, FUN2MODE). 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

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 \( E_{M,a}^\sigma (r_i) \) denotes the expected total reward for reward structure \( r_i \) under strategy \( \sigma \) of MDP \( M \), from a particular starting state \( s \).

**Lemma A.1.** Let \( M = (S, \alpha, \delta, \delta) \) be an MDP, \( \phi = ([r_1]_{\min}, \ldots, [r_n]_{\min}) \) be a multi-objective property, and \( \theta \) 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 \( \mu_{i,s}, \nu_{i,s} \in \mathbb{R} \), for \( s \in S \) and \( 1 \leq i \leq n \), are a solution to the above inequalities if an only if \( \hat{\mu}_{i,s} = \inf_{\sigma \in \delta} E_{M,s}^\sigma (r_i) \) and \( \hat{\nu}_{i,s} = \sup_{\sigma \in \delta} E_{M,s}^\sigma (r_i) \).

**Proof.** The above follows from standard results on the solution of MDPs [29], noting that there is a separate set of inequalities for each \( \mu_i \) and \( \nu_i \) 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).

\[\square\]

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

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

**Direction (1).** We start by proving that, given a sound multi-strategy \( \theta \), we can construct a satisfying assignment \( \{\hat{\eta}_{s,a}, \hat{\mu}_{i,s}, \hat{\nu}_{i,s}\}_{s \in S, a \in A} \) 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 \( \theta \) 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 \( \theta \) equals to \( \sum_{s \in S} \sum_{a \in \theta(s)} (1 - \hat{\eta}_{s,a}) \). 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 \( \theta \) should have at least one allowed action.

We set \( \hat{\mu}_{i,s} = \inf_{\sigma \in \delta} E_{M,s}^\sigma (r_i) \) and \( \hat{\nu}_{i,s} = \sup_{\sigma \in \delta} E_{M,s}^\sigma (r_i) \). 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 \( \hat{\mu}_{i,s} \geq \tilde{b}_i \) and \( \hat{\nu}_{i,s} \leq \tilde{b}_i \). This gives the satisfaction of constraints (1f) and (1g).

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

\[\square\]