Causal Explanations for Sequential Decision Making

SAMER B. NASHED*, University of Massachusetts Amherst, USA
SAADUDDIN MAHMUD, University of Massachusetts Amherst, USA
CLAUDIA V. GOLDMAN, Hebrew University, Israel
SHLOMO ZILBERSTEIN, University of Massachusetts Amherst, USA

Stochastic sequential decision-making systems — such as Markov decision processes and their variants — are increasingly used in areas such as transportation, healthcare, and communication. However, the ability to explain these systems' outputs to non-technical end users has not kept pace with their widespread adoption. This paper addresses that gap by extending prior work and presenting a unified framework for generating causal explanations of agent behavior in sequential decision-making settings, grounded in the structural causal model (SCM) paradigm. Our framework supports the generation of multiple, semantically distinct explanations for agent actions — capabilities that were previously unattainable. In addition to introducing a novel taxonomy of explanations for MDPs to guide empirical investigation, we develop both exact and approximate causal inference methods within the SCM framework. We analyze their applicability and derive run-time bounds for each. This leads to the proposed algorithm, MeanRESP, which operates flexibly across a spectrum of approximations tailored to external constraints. We further analyze the sample complexity and error rates of approximate MeanRESP, and provide a detailed comparison of its outputs—under varying definitions of responsibility—with popular Shapley-value-based methods. Empirically, we performed a series of experiments to evaluate the practicality and effectiveness of the proposed system, focusing on real-world computational demands and the validity and reliability of metrics for comparing approximate and exact causal methods. Finally, we present two user studies that reveal user preferences for certain types of explanations and demonstrate a strong preference for explanations generated by our framework compared to those from other state-of-the-art systems.

JAIR Associate Editor: Scott Sanner

JAIR Reference Format:
Samer B. Nashed, Saaduddin Mahmud, Claudia V. Goldman, and Shlomo Zilberstein. 2025. Causal Explanations for Sequential Decision Making. *Journal of Artificial Intelligence Research* 83, Article 17 (July 2025), 62 pages. DOI: 10.1613/jair.1.18126

1 Introduction

Systems for automated sequential decision making are now used in a variety of applications, including autonomous driving [159], healthcare [14], and communication networks [5], among many others. Moreover, their rate of proliferation is predicted to increase [95, 136, 52]. Whether designed for the general public or as aids in specific industries or applications, these systems often interface primarily with people who are not experts in the science of decision making and AI, or even well-versed in the principles and concepts of automation more broadly. Thus understanding, managing, and mitigating the risks of misuse, disuse, and abuse of autonomous systems in general has been a topic of considerable interest for some time [121]. Among other findings, researchers have established

*Corresponding Author.

Authors' Contact Information: Samer B. Nashed, snashed@cs.umass.edu, ORCID: 0009-0008-3010-2071, University of Massachusetts Amherst, Amherst, Massachusetts, USA; Saaduddin Mahmud, smahmud@cs.umass.edu, ORCID: 0000-0001-8767-0450, University of Massachusetts Amherst, Amherst, Massachusetts, USA; Claudia V. Goldman, claudia.goldman@mail.huji.ac.il, Hebrew University, Jerusalem, Jerusalem, Israel; Shlomo Zilberstein, ORCID: 0000-0001-9817-7848, shlomo@cs.umass.edu, University of Massachusetts Amherst, Amherst, Massachusetts, USA.

This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2025 Copyright held by the owner/author(s).
DOI: 10.1613/jair.1.18126

that developing trust is required for the adoption and proficient use of AI systems [82, 143, 162, 66], and it is widely accepted that autonomous agents that can explain their decisions help promote trust [23, 51, 101].

However, there are many challenges in generating such explanations. Consider, for example, an autonomous vehicle (AV) that stopped behind a truck for a long time. The passenger may wonder whether the AV is waiting for the truck to move, waiting for an opportunity to pass the truck, or is dealing with some technical problem. Generating suitable explanations of such a system is hard due to the complexity of planning, which may involve large state spaces, stochastic actions, imperfect observations, and complicated objectives. Furthermore, useful explanations must somehow reduce the internal reasoning process to a form understandable by a user who likely does not know all of the algorithmic details. Another challenging aspect is the heterogeneity of possible operational contexts and interaction with users whose expectations and prior knowledge may vary substantially. For example, in the above AV scenario, the explanation provided to a driver evaluating whether or not to intervene and take control of the vehicle may differ from that given to a passenger. In addition, different planning, learning, and decision-making algorithms may not provide universal mechanisms for explanation due to fundamental differences in the available information.

The debate on the definition, taxonomy, and purpose of explanations has been well represented in the cognitive science, psychology, and philosophy literature for a long time. While still active, there are several insights for which there is a broad consensus [104, 105], and we use this knowledge to motivate our approach. Scholars studying explanations largely agree that requests for explanations are often motivated by a mismatch between the mental model of the requester and a logical conclusion based on observation [54, 55, 58, 57, 86, 157], which creates a form of generalized model reconciliation problem [22]. Researchers also agree that explanations often require counterfactual analysis [92, 83, 56, 85], which in turn requires causal determination [158, 130, 84].

There are several computational paradigms for causal analysis, including those based on conditional logic [80, 39], and statistics [36]. Among the most well-studied paradigms is the *structural causal model* (SCM), which has been popularized in computer science by Halpern and Pearl [44, 45, 43]. SCMs, also known as structural equation models (SEMs), are an object of study in their own right [13] as they form an important generalization of causal Bayesian networks while retaining the critical benefit of producing interpretable directed graphs. Their expressive power, relative simplicity in interpretation, and popularity in the broader community make them an attractive tool for analyzing the output of sequential decision-making systems. In particular, we study one significant class of decision-making models, the Markov decision process (MDP). MDPs and their derivatives have several properties that make them a popular and effective method for tackling a variety of real-world decision-making problems, most notably their ability to encode uncertainty in action outcomes as well as their generality and modeling flexibility. Thus, there is an obvious desire to generate explanations for the output of these models.

This work extends previous work [114] in which we propose a framework, based on SCMs, for applying causal analysis to the behavior of sequential decision-making agents that use MDPs as their decision-making models. Our framework creates an SCM representation of the computation needed to derive an optimal policy for an MDP and applies causal inference to identify variables whose current values cause certain agent behavior. These variables can then be used to generate explanations, for example, by completing natural language templates. Our framework provides two main benefits. First, it is theoretically sound, based on concepts and formalisms from the causality literature, while many existing approaches use heuristics [31, 71, 154, 67]. Second, it is more flexible, allowing us to identify multiple semantically distinct types of explanans, whereas existing approaches often explain events in terms of a single set of variables in the decision-making model. For example, they may use *only* state factors or *only* reward variables, whereas we may use any set, increasing our framework's applicability.

Our analysis culminates in an algorithm, MeanRESP, and a process for translating MDP models into structures that facilitate causal analysis. MeanRESP may be applied to settings beyond the analysis of MDPs, is compatible with several definitions of cause and responsibility [24], and admits several approximate techniques for problems where exact inference is not tractable. We also propose several measures for comparing approximate MeanRESP

outputs to exact methods. These measures capture a range of differences and underscore the difficulty of devising a single measure for evaluating objects as complicated, nuanced, and context-dependent as explanations.

We present both theoretical and empirical results covering a range of questions regarding our framework's applicability, efficiency, and efficacy. Theoretically, we determine the domain of problems for which our framework is exact and provide some worst-case run-time bounds for the algorithms presented. We also include results on the correctness and sampling error rates for causal and responsibility determination for approximate MeanRESP. Empirically, we follow up on these theoretical results, measuring computation use, sampling error rates, and convergence rates in practice, and testing the proposed measures. We also measure disagreement between MeanRESP and Shapley-value-based methods when applied to neural networks representing both learned MDP policies and classifiers. Furthermore, we present results from two user studies. The first compares the proposed method to existing state-of-the-art heuristic methods and we find statistically significant preferences in favor of explanations generated via causal reasoning. The second investigates if and how user preferences for different explanations may change depending on their level of agency with respect to the agent providing the explanation in an autonomous driving scenario. In all, these results establish important properties of the proposed framework, highlight several of its benefits compared to existing approaches, and outline promising directions for future research on explaining stochastic planning systems.

2 Related Work

Automatically generating explanations for different AI systems is a rapidly growing field of research, and at a high level, there are two primary characteristics that create a natural taxonomy of these works: first, whether the system is used for planning or prediction, either discrete or continuous, and second, whether the computation is performed in an explicitly derived model or a learned, frequently black-box, model. Although not perfectly correlated, the prevalence of model-based planning and the popularity of black-box regression models have given rise to two distinct areas of focus, each with their own techniques and philosophies. Work on explaining the output of black-box machine learning algorithms [108, 88, 70] often uses the terms explainable or interpretable machine learning (XML). Many XML efforts focus on feature attribution, and these concepts have been applied to a large number of specific applications, including the natural sciences [127, 163], finance [17], industry [40, 41], and healthcare [90, 124], with a common approach being to compute and analyze Shapley values[1] [137, 128] and their approximations [2]. Many similar algorithms have been shown to be variations of Shapley value algorithms [89], and this insight has both popularized the application of Shapley values to the generation of explanations and encouraged many follow-up works, including modifying the original algorithm so that user goals can inform and constrain the solution set [156] and new methods for constructing the background dataset [3]. Due to the black-box nature of state-of-the-art deep reinforcement learning techniques, there have also been several applications of Shapley values to explain the policies learned by these systems [53]. Although the networks ultimately represent MDP policies, elements of the decision-making model such as the reward, transition, and value function are not explicitly represented and thus are not accessible to Shapley value analysis or any other explanation generation system.

Recent work has highlighted additional shortcomings of these approaches for explanation generation. Notably, these include the numerous plausible interpretations of how to apply these concepts to specific systems [145, 102],

[1]Named after their inventor, economist Lloyd Shapley, Shapley values assign a distribution of partial payoffs (or parts of the total surplus) generated by a coalition of players in a cooperative game. They have since been adopted for explaining mostly black-box models in machine learning by reinterpreting "surplus" or "payoff" as "model output" and substituting "players" for "input features", leading to a high-level formalization of the form: Shapley value for feature x_i in model $\pi = \frac{1}{n} \sum_{X|x_i \notin X} \frac{\pi(X \cup x_i) - \pi(X)}{N}$, where n is the total number of features and N is the number of subsets of features that have size $|X|$ and do not contain x_i. Their popularity has been driven in part by the existence of several nice properties, including efficiency, symmetry, linearity, and null player, which align with common intuitions regarding how Shapley values should be interpreted under certain conditions.

as well as challenges in handling conditional effects on the distribution of certain features, and potential ambiguity in interpreting the output—particularly in the context of actionable recourse [75]. Moreover, systems like SHAP assign values to single input variables, and do not capture the simultaneous effect of multiple counterfactual variable assignments. Overall, Shapley-value-based methods may apply favorably to some machine learning settings, which is why we include them in part of our analysis, but do not seem appropriate for understanding and communicating the reasoning that occurs within stochastic sequential decision-making systems, such as agents who use MDPs. While it is possible to apply MeanRESP to black-box systems, and we do compare the approximation error of one of the original proponents of Shapley-value-based feature attribution [142] and different versions of approximate MeanRESP, our primary focus is on the analysis of planning models.

Work on explainable planning (XAIP) has been somewhat more varied and generally aims to either explain the outputs of planning models and algorithms or modify the planning algorithms so that they produce plans that are inherently more explainable [35, 21, 7]. These concepts have been captured in a variety of metrics and definitions [20], which measure different aspects of how a plan's execution intuitively aligns with user expectations. There has also been substantial work on observer-aware systems, where plans are constructed, executed, or implicitly communicated with the explicit notion of the presence of an observer, typically in a collaborative setting [28, 140, 106, 107, 42, 148]. While valuable, we view the philosophies motivating these approaches to be largely orthogonal to, and potentially compatible with, our own. Thus, most XAIP research has been devoted to deterministic planners, or analyzing plans while still planning or after they have been executed.

More closely aligned with the efforts described in this paper, there have been promising advances in incorporating counterfactual and contrastive reasoning into classical XAIP problems [74] and applying XAIP formalisms to classification models [97]. Such methods, for example, may analyze plans with respect to higher-level properties, determining mutually exclusive properties or sets of properties that different plans could satisfy [29]. Other methods propose generating feature-based (or state-factor-based) explanations, but with respect to action sequences in deterministic planning problems [135]. In general, these approaches have the potential to use more interpretable planning models to produce more user-friendly explanations. However, many applications operate in stochastic domains or require explanations in real time. On this front, research on explanations of stochastic planners is relatively sparse; however, there are several notable existing efforts. Exact analysis or explanation, regardless of the framework, is often prohibitively expensive. Generally speaking, there are three alternatives: direct approximation of the underlying exact computation, use of heuristics, and explanation via summarization. We spend most of this paper understanding and comparing the first two approaches, but we briefly cover some summarization approaches for MDP policies first.

Summarization methods are typically more complex than their alternatives. These methods either simplify (summarize) the original problem and then provide an exact explanation of the simplified reasoning problem, or summarize the solution (e.g., policy) post-hoc and explain the simplified policy. For example, originally Brázdil et al. [2015], and later Russell and Santos [2019], use decision trees to approximate a given policy and analyze the decision nodes to determine which state factors are most influential for immediate reward. Panigutti et al. [2020] used similar methods to explain classifiers. Bustin and Goldman [2024] summarize MCTS trees by estimating the entropy of subtrees. Pouget et al. [2020] identify key state-action pairs via spectrum-based fault localization, wherein they repeatedly compare trajectories from a given policy to trajectories given counterfactual policies and try to summarize the initial policy in terms of a smaller number of key states or decision points that have a disproportionate impact on the quality of the trajectories. Linear temporal logic [6] and the HIGHLIGHTS approach [63] have also been used to create summaries of policies for explanation purposes. Other approaches use the power of summarization implicitly, for example to create alternate (possibly smaller) MDPs in which the expected and observed action are the same [33].

It is theoretically possible to combine summarization techniques with our framework by applying MeanRESP to an abstract [81] version of an MDP, although we do not explore this in detail in this paper. Summarization

methods are appealing because they parallel our intuitions about simplification in a number of other settings, such as analogizing during an explanation [27], science communication [132], and even other AI tools, such as automated text simplification [116, 141] or summarization [4]. However, these methods are often driven by heuristics and may be hard to generalize to different planners and models.

Methods for deriving explanations of the behavior of MDP agents through a more direct application of heuristics are better represented in the literature. Many of these methods apply the spirit of counterfactual reasoning without performing any sort of formal causal analysis. For example, Elizalde et al. [2009] identify important state factors by generating counterfactual states (state factors are assigned new values) and then analyzing how the value function changes given perturbations to different state factors. State factor perturbations that result in large changes in the value function are said to be more important, even if it is not possible to transition from the current state to the counterfactual state in the MDP. Wang et al. [2016] try to explain policies of partially observable MDPs (POMDPs) by communicating the relative likelihoods of different events or levels of belief. However, research clearly indicates that humans are not good at using this kind of numerical information [104].

A more common heuristic approach is to analyze, and thus produce explanations that reference, the reward function. Khan et al. [2009] present a technique to explain policies for factored MDPs by analyzing the expected occupancy frequency of states with extreme reward values. Instead of looking at how the policy is affected by the reward function overall, Juozapaitis et al. [2019] analyze how extreme reward values impact action selection in decomposed-reward RL agents, and Bertram and Wei [2018] examine reward sources in deterministic MDPs. Later, Sukkerd et al. [2020] proposed explaining factored MDPs by annotating them with "quality attributes" (QAs) related to independent, measurable cost functions. The explanations describe the QA objectives, the expected consequences of the QA values given a policy, and how those values contribute to the expected cost of the policy. The system also explains whether the policy achieves the best possible QA values simultaneously, or if there are competing objectives that required reconciliation, and proposes counterfactual alternatives. Thus, it explains entire policies, not individual actions, using custom graphics and natural language templates, the latter of which have become the de facto standard for automatic explanations. Overall, while they are computationally cheap and easy to implement, heuristic methods have limited scope in the explanations they provide, as they typically analyze only a single component of the MDP models, and do not have many theoretical advantages, if any.

Recently, Madumal et al. [2020] proposed the use of structural causal models for explaining MDPs, using SCMs to encode the influence of particular actions available to the agent. This approach was used in a model-free, reinforcement learning setting to learn the structural equations as multivariate regression models during training. However, it requires several strong assumptions, including the prior availability of a graph representing causal direction between variables, discrete actions, and the existence of sink states. Triantafyllou et al. [2022] also construct SCMs representing decentralized partially observable MDPs during policy execution, using variables that represent observations and rewards, among other things, that occur during deployment. The counterfactual analysis then concerns alternative actions. Here, the focus is on extending the definitions of cause and responsibility to the multi-agent setting in light of an agent's own ability to manipulate its level of responsibility. These concepts are loosely related to recent, more comprehensive work on causality in multi-agent settings, and games in particular [48]. In contrast, our proposed framework focuses on allowing causal analysis of all the components of MDPs using a single set of algorithms, is concentrated on the more applicable vanilla MDP model, and establishes a more rigorous experimental justification for causal explanations of stochastic planners. Moreover, it remains theoretically well-justified as it rests on a concrete theory of causality and can be easily extended for cases where approximate reasoning is required, including model-free planners.

In summary, there has been a large body of work on explainable AI systems in general, but relatively little on explainable stochastic planning. Moreover, most of those limited efforts use heuristics, making MeanRESP one of the few causality-based methods for the automatic generation of explanations of stochastic planners. Those who do propose using SCMs for explaining MDPs and their variants in both planning and learning scenarios

Table 1. Important notations, summarized from Halpern and Pearl [2005].

Notation	Meaning
X	A set of decision variables, $X = \{X_1, X_2, X_3\}$
x	An assignment of values to the set X, $\{X_1 = x_1, X_2 = x_2, X_3 = x_3\}$
$\mathcal{P}(X)$	Power set of X
$\mathcal{D}(X)$	Domain of the joint assignments of all $x \in X$
$x' \leftarrow x\|X'$	x' is the restriction of x to X', e.g. if $X' = \{X_1\}$ and $x = \{X_1 = x_1, X_2 = x_2, X_3 = x_3\}$, then $x' = \{X_1 = x_1\}$
$x \leftarrow [x\langle x']$	Replace values of x with values from x', e.g. if $x = \{X_1 = x_1, X_2 = x_2\}$ and $x' = \{X_1 = b\}$, then $x = \{X_1 = b, X_2 = x_2\}$

have likewise based their analysis on formal definitions of causality and responsibility [46, 24]. However, our method and its approximate variants offer an increased level of generality, simplicity, and efficiency by building upon an existing responsibility attribution method called RESP, introduced by Bertossi et al. [2020] to explain classification outcomes, making it a particularly compelling framework for generating explanations of stochastic sequential decision-making systems.

3 Background

Here, we review some concepts and notation relevant to the three main formalisms this paper builds upon: Markov decision processes, structural causal models, and our working definitions of weak and actual causes and responsibility. Table 1 provides a reference for common notation.

3.1 Markov Decision Processes

A Markov decision process is a model for reasoning in fully observable, stochastic environments [9]. That is, environments in which the agent knows precisely the current state of the world, but has some uncertainty over the resulting state of the world conditioned on some action it may take. Formally, we define an MDP as a tuple $M = \langle S, A, T, R, d, \gamma \rangle$, where

- S is a finite set of states. $s \in S$ may be expressed in terms of a set of *state factors*, $\langle f_1, f_2, \ldots, f_N \rangle$, such that s indexes a unique assignment of values to the factors f;
- A is a finite set of actions;
- $T : S \times A \times S \to [0, 1]$ is a transition function, representing the probability of reaching state $s' \in S$ after performing action $a \in A$ in state $s \in S$;
- $R : S \times A \times S \to \mathbb{R}$ is a reward function, representing the expected immediate reward of reaching state $s' \in S$ after performing action $a \in A$ in state $s \in S$;
- $d : S \to [0, 1]$ is a start state distribution, representing the probability of starting in state $s \in S$;
- γ is a discount factor, representing the degree of preference for immediate rewards relative to future rewards. $0 \leq \gamma < 1$.

A solution to an MDP is a policy $\pi : S \to A$ indicating that an action $\pi(s) \in A$ should be performed in a state $s \in S$. In some types of MDPs policies may be stochastic, and thus solutions may be written as $\pi(a|s)$, which is the probability of choosing action a when in state s. A policy π induces a value function $V^\pi : S \to \mathbb{R}$ representing the expected discounted cumulative reward $V^\pi(s) \in \mathbb{R}$ for each state $s \in S$.

The objective of an MDP solver is to find an optimal policy, π^*, that maximizes the expected discounted cumulative reward for every state $s \in S$. This is equivalent to satisfying the Bellman optimality equation:

$$V^*(s) = \max_{a \in A} \sum_{s' \in S} T(s, a, s')[R(s, a, s') + \gamma V^*(s')]. \tag{1}$$

There are several important properties of Equation (1). First, it is a recurrence relation, where the value of state s depends on the value of possible successor states s'. Second, given a value function, a policy may be derived; if the value function is optimal, the resulting policy will also be optimal. For any given state, the process of calculating $V^*(s)$, and thus $\pi^*(s)$, can be represented by a directed graph, which we will exploit later in the construction of data structures to facilitate causal analysis.

3.2 Structural Causal Models

Structural causal models [45, 46] are a framework for describing systems in a way that captures the causal influence of some variables on other variables. Like Bayesian networks, they offer a directed graphical representation of these influences, breaking causality or attribution problems down into three components. Formally, SCMs model scenarios, defined as tuples $\mathcal{S} = \langle \mathcal{U}, \mathcal{V}, \mathcal{M} \rangle$, where

- $\mathcal{U}$ is a set of exogenous variables, called the *context*, which are required to define the scenarios but should not be identified as causal. These variables describe some condition of the world and are considered fixed for a given scenario.
- $\mathcal{V}$ is a set of variables, known as the endogenous variables, which may be causes.
- $\mathcal{M}$ is a set of equations modeling how variables in $\mathcal{U}$ and $\mathcal{V}$ affect the variables in $\mathcal{V}$.

All variables in the world are either elements of $\mathcal{U}$ or elements of $\mathcal{V}$, and $\mathcal{U} \cap \mathcal{V} = \emptyset$. In our case, $\mathcal{V}$ are variables internal to the MDP reasoning process of the agent, such as rewards or transitions, and thus represent potential causes for the resultant policy—what we observe as agent behavior. The decision of which variables to assign to $\mathcal{U}$ and $\mathcal{V}$ is a design choice that we discuss in detail later, and usually concerns application relevance and computational cost. For example, though the presence of oxygen in the atmosphere is required for combustion, and may be necessary for a complete model describing a house fire, we typically would not want to identify its presence as one of the causes of the fire and thus may decide to assign this variable to the context.

In the above example, we would call the assignment of a value to the "is-there-a-house-fire" variable (say H), an *event*. We use ϕ to denote an event. Thus, we could equally well describe, for a given point in time, either the existence ($\phi = [H = \textsc{True}]$) or non-existence ($\phi = [H = \textsc{False}]$) of a house fire as an event. In general, events may refer to assignments of more than one variable simultaneously, and the main focus of this paper is identifying the causes of events given some scenario $\mathcal{S}$. We now describe how to encode the components of an SCM within a directed graph.

SCMs may represent directed graphs of arbitrary topology. However, most inference requires causal graphs that are directed *acyclic* graphs, or DAGs, where nodes are variables and edges denote cause-effect relations. Further improvements can be made if the causal graph is *layered* [30]. A layered causal graph (LCG) is defined given an event ϕ, for which we want to determine the causes, and a set of variables $X \subseteq \mathcal{V}$, which we would like to evaluate as causal or not (Fig. 1). An LCG is a DAG whose nodes are partitioned into non-intersecting layers $(S^k, \ldots, S^0)$, where for every edge $A \to B$ there exists some $i \in \{1, \ldots, k\}$ such that $A \in S^i$ and $B \in S^{i-1}$. Further, $X \subseteq S^k$, and $\phi \in S^0$.

3.3 Weak Causes, Actual Causes, and Responsibility

Here, we review our working definitions of weak cause, actual cause, and responsibility. Note that setting values for context variables $U = u$ induces values for all endogenous variables $\mathcal{V}$. That is, absent intervention, the assignment $U = u$ completely determines $\mathcal{V} = v$.

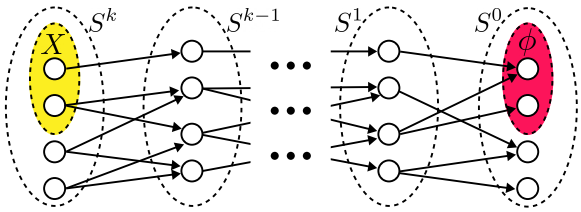


Fig. 1. A layered causal graph.

Definition 1. *Let $X \subseteq \mathcal{V}$ be a subset of the endogenous variables, and let x be a specific assignment of values for those variables. Given an event ϕ, defined as a logical expression, for instance $\phi = (\neg a \wedge b)$, a* weak cause *of ϕ satisfies the following conditions:*

(1) *Given the context $U = u$, $X = x$ and ϕ hold.*
(2) *Some $W \subseteq (\mathcal{V} \setminus X)$, $W = w$, and $X = x'$ exist such that:*
  A) given $U = u$, $X = x'$, and $W = w$, $\neg\phi$ holds.
  B) for all $W' \subseteq W$ and $Z \subseteq \mathcal{V} \setminus (X \cup W)$,
    given $U = u$, $w' = w|W'$, and $X = x$, ϕ holds for any $Z = z$.

Here, x' is simply any assignment of variables in X that is not identical to x. For a single, Boolean variable this would be its negation; in general, as few as one element of X may deviate from its original value. Thus, x' represents some possible alternate or counterfactual state. Similarly, assignments to the 'witness' set $W = w'$ (sometimes also called the contingency set) can be interpreted similarly. The main distinction between X and W is simply that we are testing for the causal strength of variables in X specifically, and not W. The same is true for Z.

Condition 1) is asserting that, given some context $U = u$, the variable assignment $X = x$ and the event ϕ actually take place. Condition 2A) ensures that the variables in X have influence on the event ϕ through their ability to falsify the event in some setting, while Condition 2B) says that, given context $U = u$, $X = x$ alone is sufficient to cause ϕ, independent of some other variables W. This and similar definitions of cause are often called "but-for" definitions. There is a related definition [44] in which condition 2B) is replaced with the following, simpler statement: for all $Z \subseteq \mathcal{V} \setminus (X \cup W)$, where $W = w$ and $Z = z$ given $U = u$, ϕ holds when $X = x$. There is also a more restrictive form of cause, the actual cause [45] (see Definition 2). For a more extensive treatment of Definition 1, as well as a number of interesting examples, we recommend reading Halpern and Pearl [2005].

Definition 2. *Let X be a weak cause. X is an* actual cause *if it is minimal. That is, if there is no such X' such that $X' \subset X$ and X' is also a weak cause.*

In practice, we find restricting explanations to contain only actual causes a helpful, though not necessarily sufficient, filtering mechanism. In addition to actual causality, we also use the following notion of responsibility [24].

Definition 3. *The* responsibility*, ρ, of a weak or actual cause X with witness set W is $\rho = \frac{1}{1+|W|}$.*

The responsibility score of the set of variables X is a measure of their ability to affect an outcome and is inversely proportional to the number of counterfactual variables required to achieve scenarios in which X satisfies the definition of weak cause. Intuitively, as fewer contingency variables are required for X to meet Definition 1, the responsibility score increases. That is, variables in X make up a large fraction of the counterfactual scenario.

4 A General Procedure for Automated Explanation Generation

Before explaining our system in detail, we present a general, abstract algorithm for automatically generating explanations that is often implicitly acknowledged, or used in part, in the design of such systems, but which we

have not seen explicitly described anywhere in the XAIP literature. Without such grounding, we feel it is easy for different work to rather ambiguously be applied to potentially many sub-problems for which it may be unintended, incomplete, or ill-suited. The following is inspired by a close reading of Miller [2019], although such steps are never explicitly mentioned. In furnishing a high-quality, automatically generated explanation, a system must:

(1) Determine the user's why-question. Specifically, determine the fact and the foil of a counterfactual scenario. For example, a fact a and foil a' represent the why-question "Why was action a taken and not action a'?"
(2) Given the fact and foil, the event in question may have many plausible or correct causes. We must find them, possibly using multiple types of information or reasons.
(3) Select a subset of causal variables from Step 2 to communicate to the user.
(4) Produce human-interpretable output (e.g., speech, text, images) based on the selected variables from Step 3.

We emphasize here that *all* of these steps represent difficult open research questions, especially considering the diversity of systems and scenarios for which we may want to generate explanations. This work focuses primarily on Step 2 and provides some results of significance related to Step 3. We assume that Step 1 has been addressed either through an external module or through system design. Although we use natural language templates to perform Step 4, this is primarily done to facilitate user study participation. We make no claims as to their efficacy relative to other potential modes of communication or even other possible text template designs.

MDPs, like other model-based planners, have a distinct advantage in explainability relative to their black-box counterparts as the key variables in the decision-making processes are typically represented explicitly within the model. The primary goal of the following sections is to understand how best to exploit this structure while 1) remaining theoretically solid, 2) remaining general and flexible with respect to the components of the model under analysis, and 3) not sacrificing the ability to apply this technique in some form to black box systems. We first focus on the underlying theory behind modeling MDP decision making with SCMs (§5-6) before introducing MeanRESP, the algorithm we use for most of our experiments, in §7. Readers seeking only an implementation should skip directly there. Figure 2 provides an overview of all algorithms we discuss in this paper and their respective roles in generating explanations for systems with different properties.

5 Structural Causal Models for MDPs

At a high level, we construct a causal model of the computation that solves for the policy of an MDP and then use this model to determine causes for agent actions, which can later be used online for explanation. Our goal is to explain agent behavior, which, if the agent is using an MDP for reasoning, is represented by partial policies executed by the agent during deployment. Thus, the most natural choice of ϕ consistent with this goal is a set of Boolean variables of the form $\pi_{sa} = [\pi(s) = a]$. Here, we use Iverson brackets to denote the Boolean evaluation for the statement $\pi(s) = a$. For example, if ϕ represents the event of taking action a in state s and not taking action a', we have

$$\phi = \langle [\pi(s) = a], [\pi(s) = a'] \rangle = \langle \textsc{True}, \textsc{False} \rangle.$$

This representation is somewhat redundant for deterministic policies, which is what we focus on in this paper. However, we note that it would in theory allow us to represent many types of queries on stochastic policies cleanly, for example if given a constrained MDP, as action probabilities are not always mutually exclusive. Collectively, the variables π_{sa} represent the policy of the MDP agent. Overall, there are $|S||A|$ variables, one for each state-action combination. We denote this set by Π. In the LCG representation, all such variables reside in the S^0 layer, although only a small fraction will be included in ϕ for any particular query.

In order to construct the LCG, we also need to determine $\mathcal{U}$ and $\mathcal{V}$. Depending on our choices, layers S^1-S^k will represent different parts of the MDP. In all variations below we can derive the agent's action for some state s,

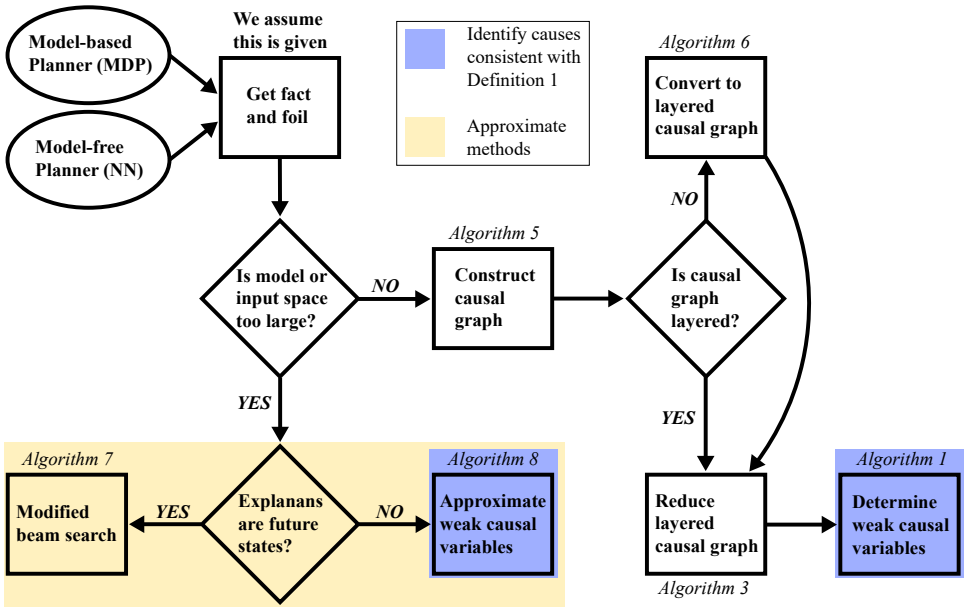


Fig. 2. Process flow diagram for generating explanations of a sequential decision-making agent. Algorithms referenced here are presented throughout the rest of the paper. Theoretically, all computation may be done offline, as the policy learned by a network or through value iteration does not change. However, since the number of fact-foil combinations is typically very large, at least some computation will likely occur online. When the causal structure of the problem is relatively simple and known in advance (as is the case for an MDP) it may be possible to replace Algorithm 5 with a more efficient, customized algorithm. The notion of a causal graph is presented as a useful abstraction which can be applied generally; as we will see, it is not strictly necessary in all cases.

given the LCG for its MDP and the values of all nodes without incoming edges, by passing values along edges and computing variables in subsequent layers of the graph until we reach layer S^0.

In the general case, our causal analysis follows four steps. (1) A causal graph is generated from the relevant MDP components. (2) The resulting graph is converted into a layered causal graph. (3) The layered graph is pruned to remove any irrelevant nodes and edges, given X and ϕ. (4) A recursive algorithm identifies sets of causal variables in the pruned graph. This approach provides a principled, general framework for causal inference on MDPs while simultaneously supporting several types of explanations. We first detail this process in layered MDPs (§5) and then discuss approximate methods for MDPs of arbitrary topology (§6). Further approximation and algorithms applicable to value function approximators such as neural networks are covered in §7.

5.1 Causal Models for Layered MDPs

We begin with the special case of layered MDPs, which contain both tree MDPs and finite-horizon MDPs, and for which our methods are exact (when the state space is discrete, up to discretization). Tree MDPs[2] are MDPs for which a tree may be built to describe all possible trajectories from any given state. Layered MDPs are a class of MDPs that we introduce subsequently in order to make our claims more precise. Although it is possible to create a single, monolithic causal graph that simultaneously represents all components of the MDP tuple, this is not helpful since it does not afford any additional types of inference, is less computationally efficient, and requires

[2]Not to be confused with a recent 'Tree MDP' description from reinforcement learning [131]

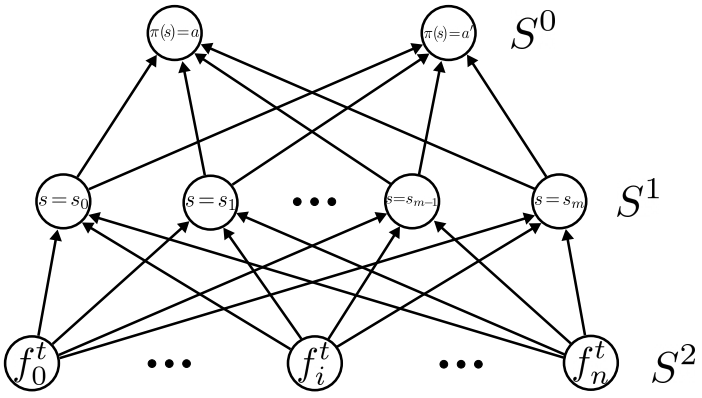


Fig. 3. A layered causal graph generated from an MDP representing the influence of state factors on the action.

substantial bookkeeping to maintain the layered property. Thus, we analyze two causal models that, together, can answer causal queries about all parts of MDPs considered in the previous literature.

Definition 4. *A* layered Markov decision process *is an MDP where, for all states $s \in S$, the state transition graph rooted at state s of horizon h is a layered graph $\forall h \in \mathbb{N}$.*

State Factors. One of the most natural questions to ask about an action prescribed by an MDP policy is its dependence on a particular state or state factors. Once the appropriate SCM is constructed, it represents a *fixed*, possibly sub-optimal policy[3]. While we *cannot* change state factors to produce a different policy, we *can* understand how state factors affect action selection for a given policy. We construct this SCM by letting $\boldsymbol{\mathcal{U}}$ consist of all variables related to the reward function R, transition function T, start distribution d, and discount factor γ. Then, $\mathcal{V}$ can be defined as $F \cup S \cup \Pi$, where F is the set of variables representing state factors, $F = \{f_1, f_2, \ldots, f_N\}$. Formally, we have

$$\mathcal{V} = \bigcup_{s \in S, a \in A} \{\pi_{s,a}\} \cup \bigcup_{s \in S} \{s\} \cup \bigcup_{i=1}^{n} \{f_i\}$$

where f_i denotes the ith state factor. Finally, $\mathcal{M}$ is composed of the following three sets.

$$\mathcal{M}_F := \{[f_i = f_i^t]\}, \qquad \forall i \in \{1, \ldots, n\}.$$

Here f_i^t is the value of state factor i at time t. A given set of state factors $\langle f_1, \ldots, f_n \rangle \in f$ determines the state $s \in S$.

$$\mathcal{M}_S := \{[s = s_j]\} = \{[f_1^t \in s_j^{f_1}] \wedge \ldots \wedge [f_n^t \in s_j^{f_n}]\}, \qquad \forall s \in S,$$

where $s_j^{f_i}$ indicates the possible values for feature f_i given state s_j. Finally, we have equations representing action selection.

$$\mathcal{M}_A := \{[\pi(s) = a]\} = \{\pi_{sa} \wedge s\} \qquad \forall s \in S, a \in A.$$

Thus we define $\mathcal{M} := \mathcal{M}_F \cup \mathcal{M}_S \cup \mathcal{M}_A$.

Figure 3 shows the causal graph represented by this SCM. This definition of SCMs for state factors creates layered graphs with exactly three layers, and when the state space is discrete, it permits exact inference regardless of the underlying MDP topology. Importantly, while this representation has some redundancy when used for

[3]That is, we can only reason counterfactually about states with respect to one policy at a time. This SCM can represent any policy, as the variables in S^0 and their function of state may be set arbitrarily.

vanilla MDPs, this is required for other models, for example, when there is uncertainty in state factor and state values, leading to a lack of mutual exclusivity in variable values.

Rewards, Transitions, and Values. The second causal model that we present is used to analyze how reward, transition, and value functions causally influence action selection. Here, we let $\mathcal{U} = \{\gamma\}$ since it is essential for computing the effect of other variables in the system, but we are unlikely to consider this a direct cause of any behavior we want to explain. Further, we let

$$\mathcal{V} = \bigcup_{s,s' \in S, a \in A} \{T(s,a,s')\} \quad \cup \quad \bigcup_{s,s' \in S, a \in A} \{R(s,a,s')\} \quad \cup \quad \bigcup_{s \in S} \{V(s)\} \quad \cup \quad \bigcup_{s \in S, a \in A} \{\pi_{s,a}\}.$$

Finally, we let $\mathcal{M}$ be the set of equations needed to solve for a policy, for instance by value iteration. The first two sets do not depend on other endogenous variables.

$$\mathcal{M}_R := R(s,a,s') = R_a^{ss'}, \qquad \forall s, s' \in S, \forall a \in A;$$
$$\mathcal{M}_T := T(s,a,s') = T_a^{ss'}, \qquad \forall s, s' \in S, \forall a \in A.$$

The set of equations for the value at each state $s \in S$ is

$$\mathcal{M}_V := V(s) = \max_a \sum_{s' \in S} T_a^{ss'} [R_a^{ss'} + \gamma V(s')], \qquad \forall s \in S.$$

Finally, we have the set of equations for action selection.

$$\mathcal{M}_A := (\pi(s) = a_k) = \Big[\sum_{s' \in S} T_{a_k}^{ss'} V(s') = A_{max}^s \Big], \ \forall s \in S.$$

Here,

$$A_{max}^s = \max_a \Big(\sum_{s' \in S} T_{a_1}^{ss'} V(s'), \ldots, \sum_{s' \in S} T_{a_m}^{ss'} V(s') \Big).$$

Thus we define $\mathcal{M} := \mathcal{M}_R \cup \mathcal{M}_T \cup \mathcal{M}_V \cup \mathcal{M}_A$.

The resultant LCG, shown in Figure 4, is built conditioned on the agent's current state in order to focus computations on the state of interest for the query. If the agent moves to a new state, a new graph is built, since reward and transition variables associated with successor states may change. Here, we also see that some layers contain value variables conditioned on particular actions ($V^a(s_i)$). Though we do not do so in our experiments, it is also possible to use this structure to ask questions like "Why does the agent take action a in state s given that it is required to take action a' in state s_i?" This is done by fixing values or collapsing subsets of the graph such that $\pi_{s_i,a'} = \textsc{True}$ (or more generally, enforcing any arbitrary partial policy). In the extreme, all such variables can be collapsed into the existing value variables ($V(s_i)$), effectively removing the max operation and reducing the graph to reproduce policy evaluation at which point we can no longer generate counterfactual scenarios consistent with the problem statement. Most importantly, it is also possible to move variables from $\mathcal{V}$ to the context $\mathcal{U}$, reducing complexity at the cost of eliminating variables from causal analysis. For example, as we show in our experiments, we could move all reward variables to generate explanations using only transition variables.

Exactness Results. Layered MDPs are well-behaved since the SCMs formed by our construction naturally form LCGs. Given $\mathcal{M}$ and a state s_0, we can construct an LCG using the layered structure of the MDP. The following theorem and lemmas give a class of MDPs for which LCGs may be constructed and analyzed exactly.

Theorem 1. *Let M be a finite-horizon MDP. If G is a layered causal graph representing M at state s, then G preserves the cause-effect relationships in the reasoning process for action selection in M at s.*

Lemma 1.1. *Given a finite-horizon MDP with horizon h and start state s_0, there exists an equivalent layered MDP.*

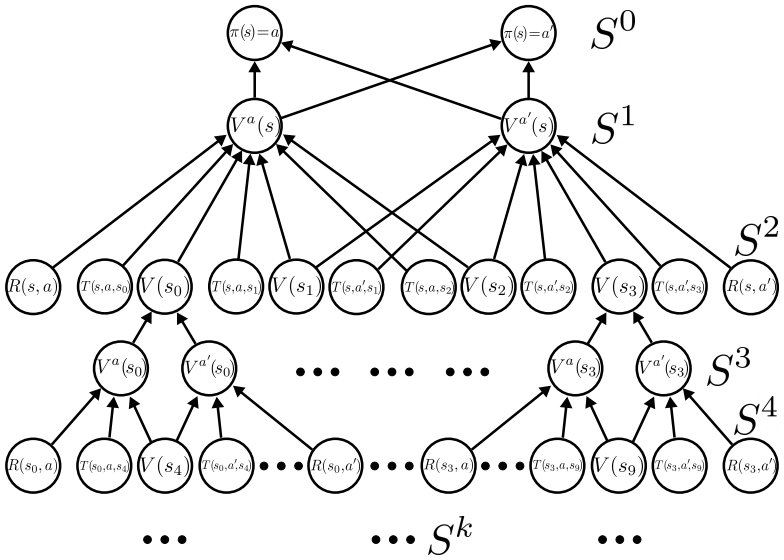


Fig. 4. A layered causal graph representing the effect of rewards, transitions, and values on action selection. $V^a(s)$ is the value of taking action a at state s. Depending on the type of analysis being done, zero-entries in the transition matrix may be represented by node omissions from the graph. Here, for example, $T(s, a', s_0) = 0$ and $T(s, a, s_3) = 0$. We have also written reward nodes as $R(s, a) = \sum_{s'} T(s, a, s')R(s, a, s')$ for readability, but in general the graph could be expanded more explicitly. Note also that this figure shows only 4 successor states in S^2, but in general there may be up to $|S|$.

Proof of Lemma 1.1: We prove this via an algorithm for constructing layered MDPs from finite-horizon MDPs. For any start state s_0, let Γ be the state transition graph rooted at s_0 such that a directed edge from node s_i to s_j exists if and only if $\exists a \in A$ s.t. $T(s_i, a, s_j) > 0$.

Next, we run Breadth First Search on Γ starting at s_0 without an explored list until all paths of length $\leq h$ have been explored and recorded. From these recorded paths, create a tree with root node s_0, appending "k" to state IDs at the kth level of the tree. After the tree is built, aggregate any duplicate nodes, preserving their edges. Such nodes will only occur within the same layer of the tree. Thus, after aggregation, the resulting state transition graph may not be a tree, but will be layered.

Finally, for all actions $a \in A$, states $s \in S$, and layers $k = \{0, \ldots, h\}$ in the original MDP, set $T_\Gamma(s_i^k, a, s_j^{k'}) = T(s_i, a, s_j)$ iff $k + 1 = k'$; set $R_\Gamma(s_i^k, a, s_j^{k'}) = R(s_i, a, s_j)$ iff $k + 1 = k'$. Construct the layered MDP $M_L = \langle S^{1:k}, A, T_\Gamma, R_\Gamma, d_{S^{1:k}}, \gamma \rangle$. □

Lemma 1.2. *If G and H are causal graphs of finite Bayesian networks, and there exists a homomorphism $G \longrightarrow H$, then G and H preserve cause-effect relationships.*

Proof of Lemma 1.2: This result follows from Jacobs et al. [2019] and Otsuka and Saigo [2022].

Proof of Theorem 1: Let H be the causal graph representing action selection in M (similar to Figure 4, but with arbitrary cycles due to the topology of the state transition graph for M). Since M is finite-horizon, then by Lemma 1.1 we can create an equivalent layered MDP, M_L. M_L has a causal graph, G (e.g., Figure 4), representing how actions are selected for any state s via policy derivation.

We can construct a function ψ that induces a homomorphism $\psi : G \to H$ in the following way: map all nodes for variables $V(s_i)$ or $T(s, a, s_i)$ in G, to the node representing s_i in H. [4] Next, map all nodes in G for variables $R(s, a, s')$ to any node $n \in H$ such that $\psi(T(s, a, s_i)) = n$. Since the homomorphism $\psi : G \to H$ exists, and since MDPs may be represented as Bayesian networks, then by Lemma 1.2, G captures all cause-effect relationships for action selection. □

5.2 Causal Inference for Layered MDPs

Given an LCG, we can perform causal inference to determine causal variables with respect to an event ϕ. Given $X \subseteq \mathcal{V}$ and ϕ, such that $\phi \cap X = \emptyset$, we would like to check if setting $X = x$ causes ϕ. That is, we would like to test whether $X = x$ satisfies Definition 1.

We now develop a naive, exact algorithm to determine weak causality. At a high level, Algorithm 1 proceeds up the causal chain in the LCG, recursively identifying causes one layer at a time. That is, sets of variables in S^1 are identified as causes of events in S^0. Those variables then assume the role of the event(s) and their causes are identified in S^2. This process repeats until the algorithm reaches layer S^k, the last layer of the LCG.

This algorithm borrows mathematical structures and notation developed by Eiter and Lukasiewicz [2006] that they use to establish a theorem (reproduced below) on identification of weak causes. We first introduce these two additional constructs before proceeding to the algorithm description. Here, $\phi_{xw}(u)$ is the value of ϕ given context $U = u$ and the assignments of variables $X = x$ and $W = w$. Loosely, elements $p \in \mathbf{p}$ and $q \in \mathbf{q}$ represent satisfying assignments w.r.t. conditions 2A and 2B of Definition 1, respectively. The conditions placed on the domains of F and w also have analogs in Definition 1. These constructs are general and do not have particular meaning with respect to MDPs, other than that the variables within p, q, F, or w will be parts of the transition function, reward function, or state factors depending on the desired type of explanation. In the case of reward or transition explanations, the relationship between variables in R^i and those in R^{i+1} is that those in R^{i+1} are relevant for calculating optimal actions one timestep farther into the future.

$$
\begin{aligned}
R^0 = \{&(\mathbf{p}, \mathbf{q}, F) | F \subseteq S^0, \mathbf{p}, \mathbf{q} \subseteq \mathcal{D}(F), \\
&\exists w \in \mathcal{D}(S^0 \setminus F) \forall p, q \in \mathcal{D}(F) : \\
&p \in \mathbf{p} \text{ iff } \neg\phi_{pw}(u), \\
&q \in \mathbf{q} \text{ iff } \phi_{[q\langle Z(u)]w'}(u) \\
&\forall W' \subseteq S^0 \setminus F, w' = w | W', Z \subseteq F \setminus S^k\},
\end{aligned}
$$

and

$$
\begin{aligned}
R^i = \{&(\mathbf{p}, \mathbf{q}, F) | F \subseteq S^i, \mathbf{p}, \mathbf{q} \subseteq \mathcal{D}(F), \\
&\exists w \in \mathcal{D}(S^0 \setminus F) \exists (\mathbf{p}', \mathbf{q}', F') \in R^{i-1} \forall p, q \in \mathcal{D}(F) : \\
&p \in \mathbf{p} \text{ iff } F'_{pw}(u) \in \mathbf{p}', \\
&q \in \mathbf{q} \text{ iff } F'_{[q\langle Z(u)]w'}(u) \in \mathbf{q}' \\
&\forall W' \subseteq S^0 \setminus F, w' = w | W', Z \subseteq F \setminus S^k, \text{ for } i > 0\}.
\end{aligned}
$$

Theorem 2. *(From Eiter and Lukasiewicz [2006]) Let $\mathcal{S} = (\mathcal{U}, \mathcal{V}, \mathcal{M})$ be a causal model. Let $X \subseteq \mathcal{V}$, $x \in \mathcal{D}(X)$, $u \in \mathcal{D}(U)$, and let ϕ be an event. Let $(S^0, \ldots, S^k)$ be a layering of $G(\mathcal{S})$ relative to X and ϕ, and let R^k be defined as above. Then, $X = x$ is a weak cause of ϕ under u in $\mathcal{S}$ iff (1) given $U = u$, $X = x$ and ϕ holds, and (2) $\exists(\mathbf{p}, \mathbf{q}, X) \in R^k$ such that $\mathbf{p} \neq \emptyset$ and $x \in \mathbf{q}$.*

[4] Figure 4 is an expanded version of the graph G, where the odd layers (representing max(·) operations) have been explicitly factored out to illustrate the possibility of modeling different policies.

Algorithm 1 DETERMINE WEAK CAUSES

1: **Input:** Layered causal graph G, context $U = u$, variables X, event ϕ
2: **Output:** Sets of weak causal variables $C_{\mathcal{W}} \subseteq X$ of ϕ.
3: $R^0 \leftarrow \emptyset$, $S^0, S^k \leftarrow$ layers of G containing ϕ, X
4: **for all** $F \in \mathcal{P}(S^0)$ **do**
5: $\quad \mathbf{p}, \mathbf{q} \leftarrow \emptyset$
6: $\quad$ **for all** $w \in \mathcal{D}(S^0 \setminus F)$ **do**
7: $\quad\quad$ **for all** $p \in \mathcal{D}(F)$ **do**
8: $\quad\quad\quad$ **if** $\neg\phi$ given p and w **then**
9: $\quad\quad\quad\quad \mathbf{p} \leftarrow p \cup \mathbf{p}$
10: $\quad\quad$ **for all** $q \in \mathcal{D}(F)$ **do**
11: $\quad\quad\quad b \leftarrow$ TRUE
12: $\quad\quad\quad$ **for all** $W' \in \mathcal{P}(S^0 \setminus F)$ **do**
13: $\quad\quad\quad\quad w' \leftarrow w|W'$
14: $\quad\quad\quad\quad$ **for all** $Z \in \mathcal{P}(F \setminus S^k)$ **do**
15: $\quad\quad\quad\quad\quad z' \leftarrow Z(u)$
16: $\quad\quad\quad\quad\quad$ **if** $\neg\phi$ given q, z, and w' **then**
17: $\quad\quad\quad\quad\quad\quad b \leftarrow$ FALSE
18: $\quad\quad\quad\quad\quad\quad$ **break**
19: $\quad\quad\quad\quad$ **if** $\neg b$ **then**
20: $\quad\quad\quad\quad\quad$ **break**
21: $\quad\quad\quad$ **if** b **then**
22: $\quad\quad\quad\quad \mathbf{q} \leftarrow q \cup \mathbf{q}$
23: $\quad R^0 \leftarrow (\mathbf{p}, \mathbf{q}, F) \cup R^0$
24: $R^- \leftarrow R^0$, $l \leftarrow 1$
25: **while** $l \leq k$ **do**
26: $\quad R \leftarrow$ RECURRENCESTEP(S^l, S^k, R^-, u)
27: $\quad R^- \leftarrow R$, $l \leftarrow l + 1$
28: $C_{\mathcal{W}} \leftarrow \emptyset$
29: **for all** $(\mathbf{p}, \mathbf{q}, F) \in R^-$ **do**
30: $\quad$ **if** $\mathbf{p} \neq \emptyset$ **and** $x \in \mathbf{q}$ **then**
31: $\quad\quad C_{\mathcal{W}} \leftarrow x \cup C_{\mathcal{W}}$
32: **return** $C_{\mathcal{W}}$

Algorithm 1 is split into an initial step and a recurrence step that together compute $R^0, \ldots, R^k$. Lines 7-9 check condition 2A from Definition 1, while lines 10-22 check condition 2B. Condition 1 is always satisfied, as ϕ represents the agent's actual policy.

The recurrence step (Algorithm 2) applies the same reasoning to the output of the initial step. Specifically, the outer loop (line 4) looks at all possible subsets of variables, F, in the ith layer. Variables not in F are assigned values w one at a time, eventually looping over all possible sets of values (line 6). Then, for every tuple R^- from layer $i - 1$ (line 7), we check the conditions for $p \in \mathbf{p}$ (lines 8-10) and $q \in \mathbf{q}$ (lines 11-23). Finally, for a given set F, we add all the qualifying p, q to the tuple R (line 24). The final result is a family of sets of causal variables $C_{\mathcal{W}}$, where $C^i_{\mathcal{W}} \subseteq X$, each of which satisfies Definition 1 with respect to the original event ϕ. We direct interested readers to Eiter and Lukasiewicz [2006] for a detailed treatment of Theorem 2 and the definitions of R^0 and R^k.

Thus, Algorithm 1 supports analysis of LCGs as in Figure 4 and, if we restrict X to the state factors in the MDP, LCGs as in Figure 3. However, it is not efficient and does not exploit any structure in MDPs. To increase efficiency, we can often prune some irrelevant nodes and edges from the graph, based on the members of event ϕ

Algorithm 2 Recurrence Step

Input: Layers S^i, S^k, tuples R^-, context $U = u$
Output: Set of tuples R.
$R \leftarrow \emptyset$
for all $F \in \mathcal{P}(S^i)$ **do**
  $\mathbf{p} \leftarrow \emptyset$; $\mathbf{q} \leftarrow \emptyset$
  for all $w \in \mathcal{D}(S^i \setminus F)$ **do**
    for all $(\mathbf{p}', \mathbf{q}', F') \in R^-$ **do**
      for all $p \in \mathcal{D}(F)$ **do**
        if F', given p and w, is in $\mathbf{p}'$ **then**
          $\mathbf{p} \leftarrow p \cup \mathbf{p}$
      for all $q \in \mathcal{D}(F)$ **do**
        $b \leftarrow$ True
        for all $W' \in \mathcal{P}(S^i \setminus F)$ **do**
          $w' \leftarrow w|W'$
          for all $Z \in \mathcal{P}(F \setminus S^k)$ **do**
            $z' \leftarrow Z(u)$
            if F', given q, z', and w', is **not** in q' **then**
              $b \leftarrow$ False
              break
          if $\neg b$ **then**
            break
        if b **then**
          $\mathbf{q} \leftarrow q \cup \mathbf{q}$
  $R \leftarrow (\mathbf{p}, \mathbf{q}, F) \cup R$
return R

and variables X. Graphs absent such variables are called *strongly reduced*. Eiter and Lukasiewicz [2006] provide an inclusive disjunction over the following conditions for removing a variable X_i from an LCG.

(1) $X_i \in X$ is not connected via variables in $\mathcal{V} \setminus X$ to ϕ.
(2) X_i is neither a direct parent of a variable in ϕ nor part of a chain connecting X to ϕ.

Algorithm 3 applies these criteria to an LCG G_{s_0}, producing a strongly reduced LCG $G_{s_0}^{\phi X}$. Depending on the structure of the MDP, such reductions may be substantial. In particular, as most MDPs have relatively sparse transition functions, the number of states that may be reached within h actions could be significantly below the theoretical worst case of $|S|$, resulting in a much smaller strongly reduced LCG.

After generating the set of weak causes $\mathcal{C}_\mathcal{W}$, we can determine actual causes by checking the minimality condition (Definition 2), using Algorithm 4. Algorithm 4 iterates through weak causal sets from smallest to largest, finding common subsets and eliminating supersets similar to basic prime-finding algorithms. In line 4, the family of weak causal sets is sorted in ascending order of size. In lines 5 and 6, an indicator is set that is False if a particular weak cause has not been checked and True if it has. In line 7, the family of causal sets is iterated over from smallest to largest. Line 8 checks if we have already either processed the set or determined it to be non-minimal. If neither of these is true, we proceed. In lines 9 and 10, we add the set to the family of actual causes and mark it as having been checked. In line 11, we iterate over all strictly larger weak causal sets. If we find a larger weak causal set containing the current actual cause as a subset, we know that it is not an actual cause and eliminate it from the analysis by marking it as having already been checked.

Algorithm 3 Reduce Causal Graph

Input: Layered causal graph G_{s_0}, explanans X, event ϕ
Output: Strongly reduced layered causal graph $G_{s_0}^{\phi X}$
$G_{s_0}^{\phi X} \leftarrow G_{s_0}$
for all $x \in X$ **do**
  if $\nexists$ path from x to some $y \in \phi$ **then**
    remove x and its edges from $G_{s_0}^{\phi X}$
  if $\forall$ paths from x to ϕ, $\exists x' \in X$ along the path **then**
    remove x and its edges from $G_{s_0}^{\phi X}$
for all $v \in \mathcal{V} \setminus (X \cup \phi)$ **do**
  if $\nexists y \in \phi$ such that v is a direct parent **and** $\nexists x \in X, y \in \phi$ such that $v \in x \rightarrow y$ **then**
    remove v and its edges from $G_{s_0}^{\phi X}$
return $G_{s_0}^{\phi X}$

Algorithm 4 Determine Actual Causes from Weak Causes

Input: Set of weak causes $\mathcal{C}_W$
Output: Set of actual causes $\mathcal{C}_A$.
$\mathcal{C}_A \leftarrow \emptyset$
$\mathcal{C}'_W \leftarrow \textsc{Sort}(\mathcal{C}_W)$ ▷ By size, in ascending order
$\vec{B} \leftarrow \vec{0}$ ▷ $\vec{B} \in \mathbb{B}^{|\mathcal{C}_W|}$
for all $F \in \mathcal{C}'_W$ **do**
  if $\neg B(F)$ **then**
    $\mathcal{C}_A \leftarrow F \cup \mathcal{C}_A$
    $B(F) \leftarrow \textsc{True}$
    for all $F' \in \mathcal{C}'_W$ such that $|F'| > |F|$ **do**
      if $F \subset F'$ **then**
        $B(F') \leftarrow \textsc{True}$
return $\mathcal{C}_A$

Overall, this collection of algorithms is powerful. However, real-valued variables such as rewards and transitions must be converted to discrete variables. For this, we assume a discretization scheme. For example, reward variables could have discrete domains bounded by the min and max of the original reward function. While this offers an actionable speed versus accuracy tradeoff, it can be hard to know exactly how to set up such a scheme. Furthermore, the variables in X must be located in the same layer in the causal graph. Although this restriction complicates the analysis somewhat, it does synergize well with the sequential nature of the decision-making problem by naturally representing the flow of value from proximal rewarding states to the current state.

6 Generalization and Approximation

Although layered MDPs encompass a large class of MDPs, Algorithm 1 has two key limitations. (1) It cannot represent infinite-horizon problems. (2) While the graph itself is straightforward to build for finite-horizon problems, very large problems or problems with large horizons may still be prohibitively expensive to analyze. In this section, we develop additional approximate algorithms to handle both finite- and infinite-horizon MDPs of arbitrary size and topology, either by constructing smaller, approximate causal models or by approximating more expensive inference processes.

6.1 Approximate Causal Models for MDPs

Here, we address limitation (1). There are many methods for building approximate causal graphs of arbitrary MDPs, depending on the available information. We assume the original graph is built using a generic, uninformed algorithm, such as Algorithm 5, based on Iwasaki and Simon [1986]. Algorithm 5 generates a causal graph given a structural causal model by first constructing a bipartite graph, where variables ($\mathcal{V}$) and equations ($\mathcal{M}$) are nodes, and edges exist between variable nodes and equation nodes if the equation contains that variable. Given the bipartite graph, Hopcroft-Karp is run to produce a perfect matching. Note that a perfect matching keeps the vertex set and selects a subset of the edges such that each vertex has 1 incident edge. This perfect matching is then used to build a *directed* (causal) graph containing only variables. Such an algorithm is not required if there is prior information regarding the causal structure. The resulting causal graph may not be unique if $\mathcal{M}$ contains circular dependencies [64, 149]. Since the Bellman update equation (Equation (1)) is a recurrence relation between MDP state values, non-layered structures are highly likely in general.

Algorithm 5 Construct Causal Graph

Input: Set of variables $\mathcal{V}$, set of equations $\mathcal{M}$
Output: Causal graph G
$\mathcal{B} \leftarrow$ ConstructBipartite($\mathcal{V}, \mathcal{M}$)
$E_{PM} \leftarrow$ Hopcroft-Karp($\mathcal{B}$)
$V \leftarrow \mathcal{V}, E \leftarrow \emptyset$
for all $v \in \mathcal{V}$ **do**
  // Q is a node in $\mathcal{B}$ representing an equation.
  for all $e(v, Q) \in$ Edges(v) **do**
    if $e \in E_{PM}$ **then**
      // V_Q is the set of variables in equation Q.
      for all $v' \in V_Q, v' \neq v$ **do**
        $E \leftarrow E \cup$ Edge(v', v)
    else
      for all $v' \in V_Q, v' \neq v$ **do**
        $E \leftarrow E \cup$ Edge(v, v')
$G \leftarrow \{E, V\}$
return G

Thus, we develop Algorithm 6, which, given state s_0 and causal graph G, removes these structures to produce an LCG G_{s_0} for causal analysis whenever the agent is in state s_0. We consider a horizon h and let variables associated with states not reachable within h actions form causal 'leaves' by removing their incoming causal edges. Remaining non-layered structures are corrected by removing edges such that states farther from s_0 causally influence states nearer to s_0, forming a simplified, finite-horizon version of the original MDP. These operations are executed simultaneously in Algorithm 6. In a sense, the causal influence within the planner flows from the horizon at time $t + h$ back in time to the present time t. Lines 4-8 label nodes, while lines 9-15 remove edges.

Once Algorithm 6 is run, the output is guaranteed to be an LCG. Thus, we can immediately run Algorithm 3 given some X and ϕ followed by Algorithm 1. Algorithm 6 produces an approximate causal model because it both disconnects some variables from the event entirely (to create an LCG with a finite number of layers) and also removes some of the causal pathways between variables that remain in the graph (to maintain the layered property). Figure 5 illustrates the transformation from an arbitrary causal graph to LCG.

Algorithm 6 Construct Layered Causal Graph

Input: Causal graph G, state transition graph H, current state s_0, horizon h
Output: Layered causal graph G_{s_0}
$G_{s_0}(E', \mathcal{V}') \leftarrow G(E, \mathcal{V})$
for all $V_{s_i} \subset \mathcal{V}'$ where V_{s_i} represents variables for state s_i **do**
  if s_i is reachable on H in $\kappa \leq h$ actions from s_0 **then**
    label all $v \in V_{s_i}$ with κ
  else
    label all $v \in V_{s_i}$ with ∞
for all $v \in \mathcal{V}'$ **do**
  if label$(v) = \infty$ **then**
    $E' \leftarrow E' \setminus \{$ all edges to/from $v\}$
    $\mathcal{V}' \leftarrow \mathcal{V}' \setminus v$
    continue
  for all $v' \in \mathcal{V}'$ **do**
    if label$(v) \geq$ label(v') **then**
      $E' \leftarrow E' \setminus \{$ directed edges from v' to $v\}$
return G_{s_0}

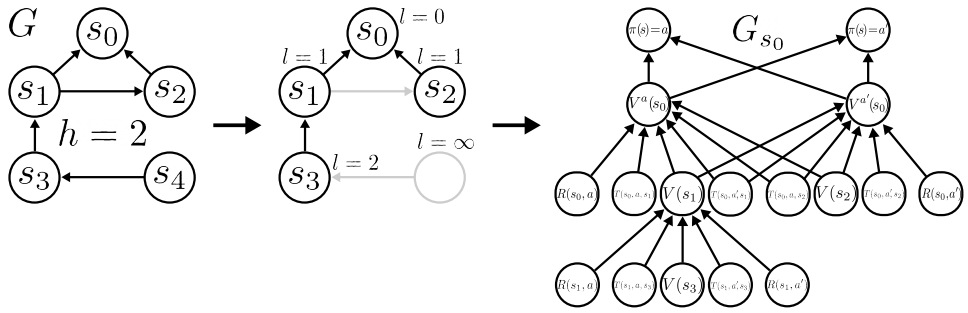


Fig. 5. Algorithm 6 operating on a causal graph G with start state s_0 and horizon $h = 2$ (state transition graph H not shown). (Left) Arrows in G denote causal influence flowing backwards in time, from distal states to proximal states, according to their reachability. (Center) States are labeled corresponding to their distance from s_0, and edges are pruned based on the labels. (Right) The final LCG is constructed from variables associated with states and connections remaining in the graph.

6.2 Approximate Causal Inference for MDPs

Regardless of topology, many MDPs are simply too expensive to analyze exactly, either because of the density of the transition function, the number of states, the length of the planning horizon, or the use of a high-fidelity discretization scheme for transition probabilities, rewards, or continuous state factors. Depending on the root cause of the complexity there are several strategies for simplification, some of which produce approximate results while others maintain exactness.

In real-world problems individual reward and transition variables are often not independent, but instead depend on a high-level rule. For example, reward may be proportional to the value of a state factor, or the transition function may encode identical slipping probabilities regardless of location, as in classic grid-world domains. These rules can constrain the transition and reward function to relatively low-dimensional manifolds, and we can discretize these manifolds to gain efficiency without sacrificing important possible worlds.

Formally, let $\Delta_{i,j}^{|S|}$ be a simplex representing the space of possible transition functions for state i and action j, that is, the possible values for $T(i, j, :)$. This space is infinite, so in order to enumerate counterfactual scenarios we will apply a discretization represented by Ω, thus restricting possible values of $T(i, j, :)$ to a finite set of points on $\Delta_{i,j}^{|S|}$. The space of possible transition functions for a given MDP is then $T(:, :, :) \in {}^{\Omega}\Delta_{1,1}^{|S|} \times \ldots \times {}^{\Omega}\Delta_{|S|,|A|}^{|S|}$. Of course, this is likely still an intractably large space. However, if the values of $T(i, j, :)$ and $T(m, n, :)$, for example, are not independent, then we can enumerate counterfactual scenarios conditioned on whatever rules relate the two values, which we call $\mathcal{R}$. In the classic grid-world domain, we can write

$$T(s, a, s') = \begin{cases} 1 - P(\text{slip}) & \text{if } s' \text{ is intended direction of } a \text{ given } s, \\ \frac{1}{2}P(\text{slip}) & \text{if } s' \text{ is left } \textit{or} \text{ right of intended direction of } a \text{ given } s, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Here, $\mathcal{R}$ represents the mapping $P(\text{slip}) \rightarrow T(:, :, :)$. Iterating over counterfactual versions of $\mathcal{R}$ is exponentially cheaper, while still visiting all possible worlds under $\mathcal{R}$, up to discretization, written $\mathcal{R}_\Omega$. The same technique can be applied to the reward function, which might naively be represented in $\mathbb{R}_\Omega^{|S||A||S|}$, but in practice is often generated according to similar high-level rules. Given such a structure, we can replace loops over, for example, all possible values of $T(s, a, s')$, and instead loop over the domain of the parameters in $\mathcal{R}$, discretized by Ω, which is much, much smaller. Often, this may be done while retaining exact solutions. It may also be combined with other methods to approximate this manifold representation and eliminate less interesting or less probable cases.

A more direct approach is to limit the sizes of W and Z, the benefit of which is a reduction in problem complexity at the cost of omitting potential weak causes. These restrictions, of course, diverge from Definition 1, but can be made in a principled way that preserves an order over the possible results. In particular, one may set $Z = \emptyset$ and/or $|W| \leq \beta$ for some $\beta \ll |\mathcal{V}|$. Limiting $|W|$ roughly corresponds to filtering weak causes based on their upper bound on responsibility from Definition 3. These strategies and their theoretical implications are covered in greater detail during our presentation and discussion of MeanRESP in §7.

Finally, if we want to look far into the future, or the branching factor of the state transition graph is large, the resulting LCG may be too large to analyze, even when limiting domains or $|W|$ and $|Z|$. Therefore, we may choose to perform causal analysis on value function variables of future states, as these variables are very efficient to analyze since they summarize the reward and transition dynamics. However, intervening directly on them breaks their consistency with respect to the Bellman optimality equation (Equation (1)), and thus results in inherently approximate inference. Instead of trying to fit this analysis to match Definition 1, we opt for a different approximate algorithm altogether. We can use a form of beam search to limit the intermediate events represented at each layer of the LCG. The idea is to measure the influence of variables on ϕ and then keep only the m most influential variables as the search progresses, where a formal definition of influence has been replaced with a heuristic. Beam search also requires the beam width m and the search depth limit h.

Algorithm 7 presents an overview. Generally, there are many reasonable definitions for the influence function. If the MDP has a strictly non-positive or non-negative value function, one straightforward definition for influence I is $I^\pi(s, s') = |V^\pi(s')T(s, \pi(s), s')|/|V^\pi(s)|$, which captures the portion of the value function at the current state s for which state s' is responsible under policy π. If the value function has both positive and negative values, we can still use the above equation without the absolute values, but the interpretation becomes slightly more complex. As written, influential future states are assumed to be at the same level of the planning graph, h actions away. This may be an issue if, for example, the policy prescribes an action which both serves to avoid a low-value state s' reachable in d' actions and also reach a high-value state s'' in d'' actions, where $d' < d'' < h$. Executing Algorithm 7 as is may not identify s' as being influential. However, saving the intermediate beams for each value of $d \in \{0, \ldots, h-1\}$ and analyzing this family of sets after beam search has terminated allows identification of such scenarios and thus identification of influential future states within horizon h rather than at exactly horizon h.

Algorithm 7 Determine Influential Future States

Input: State transition graph from current state G_{s_0}, current state s_0, policy π, beam width m, horizon h.
Output: Set of important future states B.
$B \leftarrow \{s_0\}$, $d \leftarrow 0$
while $d < h$ **do**
  $B' \leftarrow \emptyset$
  for all $s \in B$ **do**
    for all successors s' of s **do**
      $B'[s'] \leftarrow$ Influence$(s, \pi(s), s')$
  $B' \leftarrow$ Sort(B')
  $B \leftarrow B'[1:m]$
return B

In practice, an effective way to generate explanations using this method is to find one or more highly influential states and then use some of their common state factors as explanans. In our experiments, we differentiate explanans generated using Algorithm 7 (state factors from future states) from those that use the state factors of the *current* state by using a different text template that describes their relation to the current action differently. More details can be found in §8.3 and Appendix A.

In domains which are much larger, this method may also be used as a pre-processing step to further prune the LCG, although we do not provide empirical results using this technique. We should also note that there are many reasonable definitions of influence beyond the expression we use. Furthermore, there may be ways to generate counterfactual scenarios by altering the value variables directly that maintain some level of consistency with the Bellman optimality equation, although we leave this for future exploration.

6.3 Metrics for Approximate Explanations

Explanations generated by exact methods are difficult to evaluate, and the same holds true for those generated via approximate algorithms. While user studies and in situ evaluations unquestionably remain the gold standard for evaluating explanations [62], these experiments are frequently expensive and time consuming. Given the large volume of potential approximation strategies and the resulting explanations they produce, it is natural to consider automated measures that, while imperfect, can nevertheless indicate large deviations from the explanations produced by exact methods. That is, we hypothesize that, while we cannot know the ultimate quality of an explanation, exact or approximate, via automated metrics alone, we can at least compare the similarity of approximate results along several dimensions to their exact counterparts using automated techniques.

Often, there exist multiple weak (or actual) causal sets for a given event. Thus, it is natural to restrict the output of, for example, Algorithm 1, to a maximum of k sets, creating 'top k' causal queries. There are three complementary pieces of potential information: 1) the existence or absence of a particular variable (or set) within the top k, 2) the rank of a particular set within the top k, and 3) a real-valued responsibility score associated with each set. Given (3), both (1) and (2) may be derived, and given (2), (1) may be derived, but not (3). Measuring the quality and similarity of top k results has been a topic of interest in databases and recommender systems for some time, with a variety of conceptualizations and implementations [32, 103, 147, 151]. Here, we propose several measures for comparing top k results in the context of automatic explanations, depending on the available information provided along with the top k causal sets. Some measures have been represented in different forms in the literature, and some may not be common in top k comparisons. In practice, as we show in the later evaluation, we expect a combination of these measures to yield the most insight.

We denote a function computing the similarity between the exact top k weak causal sets, ${}^kC^*_{\mathcal{W}}$, and the approximate top k weak causal sets, ${}^kC^\alpha_{\mathcal{W}}$, as $\mu({}^kC^*_{\mathcal{W}}, {}^kC^\alpha_{\mathcal{W}})$. Similarly, we denote the exact and approximate top k rankings, or orderings, over weak causal sets as ${}^k\mathcal{O}^*$ and ${}^k\mathcal{O}^\alpha$, respectively. Last, the sets of exact and approximate responsibility values are ${}^k\rho^*$ and ${}^k\rho^\alpha$, respectively. All measures μ are identical for actual causal sets.

6.3.1 *Causal Set Contents Only.*

The simplest Boolean comparison we can make is to check if ${}^kC^*_{\mathcal{W}} = {}^kC^\alpha_{\mathcal{W}}$. Thus,

$$\mu_{\mathbb{B}}({}^kC^*_{\mathcal{W}}, {}^kC^\alpha_{\mathcal{W}}) = \begin{cases} 0 & \text{if } {}^ix^* = {}^ix^\alpha \quad \forall \ {}^ix^* \in {}^kC^*_{\mathcal{W}}, {}^ix^\alpha \in {}^kC^\alpha_{\mathcal{W}}, i \in \{1, \dots, k\} \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

Obviously, $\mu_{\mathbb{B}}$ misses out on significant nuance. However, we should note that, because this is a top k result, it does admit many results which are not completely identical, but whose discrepancies occur outside the top k items, making it a weak but relevant baseline. Next, we consider the presence or absence of different variables within the top k causal sets, since in many cases the user will see only the causes and not their relative importance. Let $\mathcal{I}^* = \cup^k_{{}^ix^* \in {}^kC^*_{\mathcal{W}}} {}^ix^*$ and $\mathcal{I}^\alpha = \cup^k_{{}^ix^\alpha \in {}^kC^\alpha_{\mathcal{W}}} {}^ix^\alpha$, then we have

$$\mu_{\mathcal{I}}({}^kC^*_{\mathcal{W}}, {}^kC^\alpha_{\mathcal{W}}) = \left| |\mathcal{I}^\alpha \cup \mathcal{I}^*| - |\mathcal{I}^\alpha \cap \mathcal{I}^*| \right|. \tag{4}$$

This measure represents a slightly generalized form of Hamming distance, equivalent to some definitions of recall, providing a basis to understand what, if anything, has been erroneously included or omitted in an approximate family of weak causal sets. $\mu_{\mathcal{I}}$ will approach 0 as the number of samples increases, but not necessarily monotonically.

6.3.2 *Ranking.*

Given ${}^k\mathcal{O}^*$ and ${}^k\mathcal{O}^\alpha$, where ${}^io \in {}^k\mathcal{O}$ is the rank $(1, \dots, k)$ of ${}^ix$ within ${}^kC_{\mathcal{W}}$, we can calculate the rank correlation coefficients between ${}^k\mathcal{O}^*$ and ${}^k\mathcal{O}^\alpha$. The two most obvious choices are Kendall's τ and Spearman's ρ, the latter of which we denote using the subscript S, ρ_S, to differentiate it from responsibility.

$$\mu_\tau({}^k\mathcal{O}^*, {}^k\mathcal{O}^\alpha) = \frac{2}{k(k-1)} \sum_{j=2}^{k} \sum_{i=1}^{j-1} \operatorname{sgn}({}^io^* - {}^jo^*) \operatorname{sgn}({}^io^\alpha - {}^jo^\alpha), \tag{5}$$

and

$$\mu_{\rho_S}({}^k\mathcal{O}^*, {}^k\mathcal{O}^\alpha) = 1 - \frac{6 \sum_{i=1}^{k} ({}^io^* - {}^io^\alpha)^2}{k(k^2-1)}. \tag{6}$$

In our system such rankings would be generated using responsibility scores, so the existence of ${}^k\mathcal{O}^*$ without ${}^k\rho^*$ does not occur naturally. However, this is likely not the case for systems in general. Both μ_τ and μ_{ρ_S} will converge to 1 as the number of samples increases, though again not monotonically. In both this section and the next, since we do not know if the order (or the order of the responsibility scores) for exact and approximate methods are identical, we need to ensure that, when analyzing ordinal positions ${}^io^*$ and ${}^jo^\alpha$ (or scores ${}^i\rho^*$ and ${}^j\rho^\alpha$), they refer to the same underlying sets, that is, ${}^ix^* = {}^jx^\alpha$.

6.3.3 *Responsibility.*

Having access to a score for each causal set, in our case the raw responsibility scores, provides an opportunity for capturing additional nuance in our measures. Here, we present several options. The first and perhaps most

basic option is to calculate the Euclidean distance between the vectors representing ${}^k\rho^*$ and ${}^k\rho^\alpha$,

$$\mu_{\mathcal{E}}({}^k\rho^*, {}^k\rho^\alpha) = \Big(\sum_{i=1, j \mid {}^i x^* = {}^j x^\alpha}^{k} ({}^i\rho^* - {}^j\rho^\alpha)^2 \Big)^{\frac{1}{2}}. \tag{7}$$

As the number of samples increases, $\mu_{\mathcal{E}}$ approaches 0. However, this measure is not scale-invariant, either with respect to the magnitude of the scores or the size of k. So, depending on the application, quality judgments using this measure may be too permissive or too restrictive. Second, similar to μ_τ and μ_{ρ_S}, we can compute the correlation between ${}^k\rho^*$ and ${}^k\rho^\alpha$ using Pearson's r,

$$\mu_r({}^k\rho^*, {}^k\rho^\alpha) = \frac{k \sum_{i=1, j \mid {}^i x^* = {}^j x^\alpha}^{k} ({}^i\rho^* \, {}^j\rho^\alpha) - \sum_{i=1}^{k} {}^i\rho^* \sum_{i=1}^{k} {}^i\rho^\alpha}{\sqrt{k \sum_{i=1}^{k} ({}^i\rho^*)^2 - (\sum_{i=1}^{k} {}^i\rho^*)^2} \sqrt{k \sum_{i=1}^{k} ({}^i\rho^\alpha)^2 - (\sum_{i=1}^{k} {}^i\rho^\alpha)^2}}. \tag{8}$$

Here, as the number of samples increases, we expect μ_r to approach 1. This measure, as with others in this section, will generally take much longer to converge completely than measures based on ranking alone, since it is unlikely that approximate solutions will produce exactly the same scores as their exact counterparts, even if the variables ultimately highlighted remain the same. Depending on how these potential explanans are processed, differences in such scores could result in fundamentally different explanations being generated for users.

Similar in spirit to μ_r, we may understand the similarity between ${}^k\rho^*$ and ${}^k\rho^\alpha$ in a more geometric manner. Let $\bar{\rho}^* = \frac{1}{k} \sum_{i=1}^{k} {}^i\rho^*$ and $\bar{\rho}^\alpha = \frac{1}{k} \sum_{i=1}^{k} {}^i\rho^\alpha$ be the mean scores for the exact and approximate causal sets, respectively. We can find the slope of a least-squares line of best fit, where approximate values are plotted as functions of their exact values, as

$$\mu_m({}^k\rho^*, {}^k\rho^\alpha) = \frac{\sum_{i=1, j \mid {}^i x^* = {}^j x^\alpha}^{k} ({}^i\rho^* - \bar{\rho}^*)({}^j\rho^\alpha - \bar{\rho}^\alpha)}{\sum_{i=1}^{k} ({}^i\rho^* - \bar{\rho}^*)^2}. \tag{9}$$

As the number of samples increases, μ_m will approach 1. This measure is very sensitive to errors at the extremes (ranks 1 and k), but is less so to systematic errors that result in a more uniform 'shift' of approximate scores with respect to their exact counterparts.

So far, we have focused the more informed measures primarily on how well the top k approximate sets match the exact solution in terms of order and, to a lesser degree, score scale. All of these measures converge more slowly as k grows larger. Moreover, many of them are particularly susceptible to localized discrepancies of one form or another, including scale. Here, we present a final measure that offers an alternative emphasis. Let $||\rho^*||_1$ and $||\rho^\alpha||_1$ be the sum of *all* exact and approximate scores, respectively. Let $||{}^k\rho^*||_1$ and $||{}^k\rho^\alpha||_1$ define this quantity for the top k items. Then, we have

$$\mu_p(\rho^*, \rho^\alpha) = \frac{||{}^k\rho^\alpha||_1}{||\rho^\alpha||_1} \left(\frac{||{}^k\rho^*||_1}{||\rho^*||_1} \right)^{-1}. \tag{10}$$

As the number of samples increases μ_p will approach 1. This measures the relative score mass in the top k between the exact and approximate results, making it scale-invariant. Many scores rely either directly or indirectly on the number of analyses performed in their calculation. Since one of the most common strategies for approximation is to skip some of these analyses, this can decrease the effectiveness of scale-sensitive measures.

The measures presented here (evaluated further in §8.2.2) represent only a fraction of plausible choices, and many avenues remain unexplored. What is clear is that providing scores alongside causal sets can be helpful in determining what to show a user or even when to terminate anytime approximate computation. To this end, we next present MeanRESP, an algorithm for causal analysis that operates on a continuum of approximation, provides principled responsibility scores alongside identifying weak causes, and which may be applied to both MDPs as well as black box models such as neural networks.

7 MeanRESP

The previous sections outline a general pattern of analysis and a clear need for both approximate algorithms and the provision of additional ranking or scoring information to supplement causal determination. Moreover, we would like the scoring to be calculated as a by-product of causal determination and behave as follows:

(1) **Property 1:** A set of variables $X \subseteq F$ that is not a cause of the event ϕ should have $\rho = 0$. A set of variables $X \subseteq F$ that is a cause of the event ϕ should have $\rho > 0$.
(2) **Property 2:** As the cause allows a set of witness variables W, ρ should divide the causal responsibility among the cause X and witness W in a principled manner. That is, the larger the witness set required to identify X as causal, the lower ρ should be.
(3) **Property 3:** A relatively higher value of ρ for a cause $X \subseteq F$ should indicate the event ϕ is relatively more affected by the assignment $X = x$.

To meet these criteria, we propose an algorithm similar to that presented by Bertossi et al. [2020], based on the concepts of responsibility and blame from Chockler and Halpern [2004]. Algorithm 8, which we call MeanRESP as it essentially computes a mean responsibility score over sets of potential variables, iterates directly through possible weak causal sets (line 4), and then progressively checks larger sets W for assignments w that satisfy Definition 1. Lines 11-15 and 19-22 check conditions 2B and 2A, respectively. Finally, lines 23-27 compute the responsibility score ρ, used to determine whether X is weakly causal. In addition to finding weak causal sets consistent with Definition 1, ρ provides a ranking over causal sets. Note that MeanRESP returns a value greater than 0 whenever both 2A and 2B hold, ensuring Property 1. The first condition in Definition 1 always holds for a policy or classifier, and therefore is not explicitly checked. Additionally, accumulating responsibility scores in line 23, where the size of the witness set appears in the denominator, provides Property 2. Due to the accumulation in line 22, ρ retains proportionality to the fraction of assignments to X that satisfy the definition of weak cause. This gives MeanRESP Property 3.

7.1 Monte Carlo MeanRESP

Often, models are too large for exact inference. Moreover, we may wish to apply more restrictive versions of Definition 1, or extend the analysis to real-valued variables without applying discretization. We can address these problems by modifying MeanRESP. If variable domains are real-valued or finite but very large, we can approximate inference by sampling rather than iterating over the sets in lines 4, 5, 7, and 8, as shown in the pseudocode comments in Algorithm 8. The condition that $|W| = \beta$ on line 7 is retained. Thus, sampling may be constrained along several dimensions independently, based on the most expensive features of the problem. Counterfactual variable assignment and event pairs are constructed and counted in the same way, and non-zero responsibility scores still indicate weak causality. Monte Carlo MeanRESP recovers the exact solution in the limit, but the practical challenge becomes determining sampling domains that efficiently cover important counterfactual scenarios.

This approach turns out to produce a scaled form of the Shapley value in expectation; however, we note that the scores produced via MeanRESP are also applicable to sets of variables. If we assume witness set samples are equally divided among all allowed β-values[5], then we have the following general expression for the expected responsibility score:

$$\mathbb{E}_{\beta\sim\mathcal{U}(0,|F\setminus X|),W\sim\mathcal{P}_\beta(F\setminus X),w\sim\mathcal{D}(W),x'\sim\mathcal{D}(X)}\left[\frac{\phi(x_p)}{1+\beta}(\phi(x)-\phi(x_m))\right]. \tag{11}$$

[5]This is often a valid assumption, given small enough (though still, practically speaking, large) values of κ and η. More generally, the assumption of equiprobable W under different β values does not hold, since most $W \in \mathcal{P}(F \setminus X)$ have size close to $|F \setminus X|/2$.

Algorithm 8 MeanRESP

1: **Input:** Potential causal variables F, policy π, initial setting s_0
2: **Output:** Set of weak causes $C_{\mathcal{W}}$, responsibility scores $\mathcal{R}$.
3: $C_{\mathcal{W}}, \mathcal{R} \leftarrow \emptyset$
4: **for all** $X \in \mathcal{P}(F)$ **do** ▷ $1...\kappa$ sample $X \sim \mathcal{P}(F)$
5: **for all** $\beta = 0...|F \setminus X|$ **do** ▷ $\beta = 0...\beta_{UB} < |F \setminus X|$
6: $\sigma, t \leftarrow 0$
7: **for all** $W \in \mathcal{P}(F \setminus X)$ such that $|W| = \beta$ **do** ▷ $1...\eta$ sample $W \sim \mathcal{P}(F \setminus X)$
8: **for all** $w \in \mathcal{D}(W)$ **do** ▷ $1...\upsilon$ sample $w \sim \mathcal{D}(W)$
9: $b \leftarrow$ True
10: $t \leftarrow t + 1$
11: **for all** $W' \in \mathcal{P}(W)$ **do**
12: $w' \leftarrow w|W'$
13: $s' \leftarrow [s_0\langle w']$
14: **if** $\pi(s') \neq \pi(s_0)$ **then**
15: $b \leftarrow$ False
16: **break**
17: **if** $\neg b$ **then**
18: **continue**
19: **for all** $x' \in \mathcal{D}(X)$ such that $s_0|X \neq x'$ **do** ▷ $1...\chi$ sample $x' \sim \mathcal{D}(X)$
20: $s' \leftarrow [s_0\langle (x' \cup w)]$
21: **if** $\pi(s') \neq \pi(s_0)$ **then**
22: $\sigma \leftarrow \sigma + \frac{1}{|\mathcal{D}(X)|}$
23: $\rho \leftarrow \sigma/(t(1+\beta))$
24: **if** $\rho > 0$ **then**
25: $C_{\mathcal{W}} \leftarrow C_{\mathcal{W}} \cup X$
26: $\mathcal{R}$.Append(ρ)
27: **break**
28: **return** $C_{\mathcal{W}}$

Here, $x_p = s' = [s_0\langle w']$ and $x_m = s' = [s_0\langle (x' \cup w)]$ from lines 13 and 20 in Algorithm 8, respectively[6]. $\phi(\alpha) = 1$ if the event ϕ is true given α (e.g. $\phi(x_p) = \phi_{xw'}$ and $\phi(x_m) = \phi_{x'w}$). Here, $\phi(x_p) = 1$ if X satisfies condition 2B from Definition 1, and $\phi(x_m) = 0$ if X satisfies condition 2A from Definition 1. We can see in Equation (11) that, whenever $\phi(x_p) = 1$, $\phi(x) = 1$, and when $\phi(x_p) = 0$, the entire expression will be zero, regardless of the value of $\phi(x)$. Thus, we can replace $\phi(x)$ with $\phi(x_p)$, yielding

$$\mathbb{E}_{\beta\sim\mathcal{U}(0,|F\setminus X|),W\sim\mathcal{P}_\beta(F\setminus X),w\sim\mathcal{D}(W),x'\sim\mathcal{D}(X)}\left[\frac{\phi(x_p)}{1+\beta}(\phi(x_p)-\phi(x_m))\right]. \tag{12}$$

Removing the multiplicand, we recover a Monte Carlo approximation of the expected Shapley value:

$$\mathbb{E}_{\beta\sim\mathcal{U}(0,|F\setminus X|),W\sim\mathcal{P}_\beta(F\setminus X),w\sim\mathcal{D}(W),x'\sim\mathcal{D}(X)}\left[(\phi(x_p)-\phi(x_m))\right]. \tag{13}$$

Intuitively, responsibility can be thought of as distance-weighted Shapley value, where $1 + \beta$ captures the difference between the original input x and x_m, and $\phi(x_p)$ captures the difference in output between $\pi(x)$ and $\pi(x_p)$. The simplicity and efficiency of Monte Carlo MeanRESP make it an attractive option for many problems, and in theory it may be applied to other, more complicated variants of MDPs, such as partially observable MDPs.

[6]We use the notation x_p (for $x+$) and x_m (for $x-$) to match notation commonly used in Shapley value calculations. The terms $\phi(x_p)$ and $\phi(x_m)$ are similar to $\pi(X \cup x_i)$ and $\pi(X)$ from footnote 1, respectively.

In §8.1, we will further theoretically analyze this algorithm, and in §8.2 we show a variety of empirical results to this effect. First, however, we give an overview of a generalized version of MeanRESP compatible with multiple related but distinct versions of weak cause, essentially allowing us more flexibility beyond Definition 1.

7.2 Generalized MeanRESP

Although we have so far presented a relatively rigid framework for identifying causes of agent behavior, built on Definitions 1—3, there are several plausible versions of MeanRESP which all detect sets of variables that satisfy related definitions of weak cause. Before introducing a novel, generalized version of MeanRESP, we outline the most important axes of variation over which we would like to generalize. First, there are at least three different definitions of weak cause proposed by Halpern, and we show compatibility with both the updated definition [45] (RESP-UC, Algorithm 10) and the original [44] (RESP-OC, Algorithm 11). We omit a more recent, modified definition [43]. Not only will the choice of definition for weak cause affect the resultant responsibility scores, but it will also change what is identified as a cause. Some sets of variables will have non-zero responsibility scores under only one definition.

Second, the mean responsibility score can be calculated in two ways. It may be tallied over only the witness sets of size β_{min}, where β_{min} is the smallest β for which there exists a satisfying witness set (as in [114]). Or, it may be tallied over all witness sets, regardless of β, as in Algorithm 8. Actual causes with at least some small witness sets will receive lower responsibility scores under the latter design.

Third, as responsibility incrementally accrues with respect to an actual causal set, these increments can either be counted equally, or can be normalized by the size of the domain of the actual cause. We refer to this as the option to perform domain normalization, and the theory behind it is that with a larger domain the chance that some assignment $X = x'$ will meet the conditions of Definition 1 increases, and thus the responsibility should correspondingly decrease. This option is represented as a comment within the pseudocode.

None of these choices interfere with Properties 1—3 outlined earlier, but they may subtly alter the relative responsibility assigned to different weak or actual causes. As there is no clear reason based on first principles to prefer one choice over another, these decisions involve tradeoffs. For example, short circuiting after finding a single witness set of size β that satisfies Definition 1 will save compute time, but may give a slightly higher or lower responsibility score depending on whether the variables of interest are important under many counterfactual scenarios or only a few. Similarly, foregoing domain normalization may downplay the importance of singleton causes relative to causes composed of multiple variables or other singleton variables with larger domains.

Algorithm 9 Generalized MeanRESP

Input: All variables F, variables of interest (VoI) X, event ϕ, initial variable settings s_0, initial VoI assignment x, responsibility function RESP
Output: Mean responsibility scores ρ.
$\rho \leftarrow 0$
for all $\beta = 0...|F \setminus X|$ **do**
  $\sigma, T \leftarrow 0$
  for all $W \in \mathcal{P}(F \setminus X)$ such that $|W| = \beta$ **do**
    for all $w \in \mathcal{D}(W)$ **do**
      $T \leftarrow T + 1$
      $\sigma \leftarrow \sigma + \text{RESP}(\phi, X, W, w, s_0)$
  $\rho \leftarrow \rho + \frac{\sigma}{T}$
return $\frac{\rho}{|\mathcal{D}(X)|}$

Algorithm 10 RESP-UC

1: **Input:** ϕ, X, W, w, s_0
2: **Output:** Score σ
3: $\sigma \leftarrow 0$
4: **for all** $W' \in \mathcal{P}(W)$ **do**
5: $w' \leftarrow w|W'$
6: $x_p \leftarrow [s_0\langle w']$
7: **if** $\neg\phi(x_p)$ **then**
8: **return** σ
9: **for all** $x' \in \mathcal{D}(X)$ such that $s_0|X \neq x'$ **do**
10: $x_m \leftarrow [s_0\langle(x' \cup w)]$
11: **if** $\phi(x_m)$ **then**
12: $\sigma \leftarrow \sigma + 1$
13: **return** $\frac{\sigma}{1+|W|}$

Algorithm 11 RESP-OC

1: **Input:** ϕ, X, W, w, s_0
2: **Output:** Score σ
3: $\sigma \leftarrow 0$
4: $x_p \leftarrow [s_0\langle w]$
5: **if** $\neg\phi(x_p)$ **then**
6: **return** σ
7: **for all** $x' \in \mathcal{D}(X)$ such that $s_0|X \neq x'$ **do**
8: $x_m \leftarrow [s_0\langle(x' \cup w)]$
9: **if** $\phi(x_m)$ **then**
10: $\sigma \leftarrow \sigma + 1$
11: **return** $\frac{\sigma}{1+|W|}$

Algorithm 9 thus represents generalized MeanRESP. After fixing a witness $W = w$ (lines 7-8), the responsibility score is calculated (line 10). Domain normalization on line 12 is optional, and in Monte Carlo MeanRESP the divisor is the number of instances of x' sampled, not the (potentially infinite) size of the domain. In RESP-UC (Algorithm 10), if condition 2B holds from Definition 1 (lines 4-9), then we check for condition 2A (lines 10-12).

7.3 Semantics of Causal Variables as Explanans

We have presented several related methods for identifying causal variables given an event in the context of a stochastic planner, policy approximator, or classifier. However, not all causal variables are equally well-suited for use as explanans. This is not only due to individual and systemic human preferences, covered at length by Miller [2019] and in depth with respect to our system in §8.3, but also due to the explanans and the counterfactual scenarios themselves representing fundamentally different semantics. Most importantly, we need to highlight that explanations generated using our framework, or any of the others we refer to or compare against, do not and cannot explain action *outcomes*. That is, they cannot provide reasons for why an action with stochastic outcomes resulted in a particular outcome. They can, however, provide reasons for why a particular action was or was not *chosen*.

When explaining the nature of such a partial policy, there are two broad classes of counterfactual scenarios. The first references a fixed world model and a counterfactual scenario. In an MDP, this corresponds to fixed transition and reward functions and thus a fixed policy, and the counterfactual scenario corresponds to the agent being in a different state. Explanations generated in this way will be based on reasons related to the state factors

Table 2. Comparison of method applicability

Method	F	R	T	V	Causal?
Elizalde et al. [2009]	Yes	-	-	-	No
Russell and Santos [2019]	Yes	-	-	-	No
Khan et al. [2009]	-	Yes	-	-	No
Juozapaitis et al. [2019]	-	Yes	-	-	No
Bertram and Wei [2018]	-	Yes	-	-	No
Wang et al. [2016]	-	-	Yes	-	No
Madumal et al. [2020]	Yes	Yes	-	-	Yes
MeanRESP	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

of the MDP and their possible alternate values. Generally, such explanations will rely on reasoning similar to "I took action a because X is true in my current *state*."

The second class references a fixed scenario and a counterfactual model of the world. In an MDP, this corresponds to altering the transition or reward functions while fixing the state factors. Explanations generated using this class of counterfactual scenarios will provide reasons for action selection based on alternate possible worlds. Generally, they will rely on reasoning similar to "I took action a because X is true about my current *world*." When using a policy approximator such as a neural network, this type of counterfactual query is not possible as agent's world model is not exposed in an interpretable manner. Technically, the calculations can be run, but it is unlikely that the results can be translated fruitfully back into terms a human user could understand.

Given these choices and the potential for further differentiation, such as between transition and reward variables, it is not at all clear how to best define potential explanans, X. Intuitively, all previous work defines X as being, for example, the set of all state factors or the set of all reward variables. That is, we tend to define X according to some semantic type. While many other methods implement forms of analysis specific to certain subsets of variables, thus requiring such a definition, our framework does not. However, while it may be difficult to justify from first principles, we still find it useful to follow this intuition, albeit with additional flexibility, as our framework allows us to furnish explanations using many definitions for X, as seen in Table 2. Broadly, we have identified four types of explanation in the literature, each focusing on one component of the MDP tuple: *state factors* (F) (Elizalde et al. [2009]; Russell and Santos [2019]), *rewards* (R) (Khan et al. [2009]; Juozapaitis et al. [2019]; Bertram and Wei [2018]), *transitions* (T) (Wang et al. [2016]), and *future states and values* (V) [125]. These papers define metrics, algorithms, and definitions particular to their type, and lead us to define the following.

Definition 5. *Y-type explanations use explanans $X \subset Y$. For example, R-type explanations use reward variables.*

8 Results

In this section, we cover a range of results, both theoretical and empirical, that address questions regarding the practical, effective use of our framework to generate explanations of AI reasoning systems. Theoretically, we focus primarily on run-time bounds, convergence properties, and error rates of MeanRESP and Monte Carlo MeanRESP. Empirically, we cover some comparisons with Shapley values, provide some insight into effective measures of similarity between the top k lists of explanans, and study an array of user preferences, as well as more basic results that confirm earlier theoretical claims. Though our primary example is of an autonomous vehicle, we also present some results on other types of domains, including both discrete and continuous planning domains as well as some classification systems. This set of experiments represents a much more complete picture of system performance than has previously been available for any other MDP explanation system, and overall, our results suggest that MeanRESP and its variants are an extremely competitive option for automatic explanation generation for a wide variety of AI systems, and especially model-based planners, such as MDPs.

8.1 Theoretical Analysis

Here, we cover several properties of Monte Carlo MeanRESP, as well as provide some preliminary run-time and memory bounds for some of the algorithms presented earlier. We also briefly discuss the tightness of these bounds in practice.

8.1.1 On the Performance of Monte Carlo MeanRESP: Errors and Convergence.

We are most interested in the correctness, one-sidedness of errors, error rate, and sample efficiency, and below we present several propositions exploring these topics. The first concerns the adherence of MeanRESP to the definition of weak cause given in Definition 1.

Proposition 2.1. *Given a set of variables X and an event ϕ, X is a weak cause of ϕ according to Definition 1 if and only if the responsibility score, ρ, output by MeanRESP (Algorithm 8) is greater than zero.*

Proof: If $\rho > 0$, the condition on line 21 in Algorithm 8 must be true. This condition is True when there exist assignments $X = x'$ and $W = w$ such that $\phi(x', w)$ is False. The existence of such assignments satisfies condition 2A from Definition 1. Line 21 is only executed when b is True, corresponding to the variable assignment $X = x$ satisfying condition 2B from Definition 1 w.r.t. event ϕ and all alternative contingency sets $W = w'$. Condition 1 from Definition 1 is always trivially met as the event ϕ represents an existing partial policy. Thus, $\rho > 0$ iff there exist $X = x'$ and $W = w$ that satisfy Definition 1. □

Proposition 2.2. *The false positive rate of Monte Carlo MeanRESP is 0.*

Proof: Estimates for ρ^*, denoted ρ, are initialized to 0. As shown in the proof of Proposition 2.1, the only way to increment ρ is to satisfy Definition 1. Therefore, it is not possible to attain a positive ρ estimate without being a weak cause, and thus Monte Carlo MeanRESP has a false positive rate of 0. □

We also see that the error rate in Monte Carlo MeanRESP is bounded.

Proposition 2.3. *Let ρ^* be the true responsibility score for the set of potential causal variables X. The expected false negative rate of Monte Carlo MeanRESP, ε, after n samples is $(1 - (\rho^*|F \setminus X|))^n \leq \varepsilon \leq (1 - \rho^*)^n$.*

Proof: ρ^* represents the number of times a contingency set W exists such that X satisfies Definition 1, divided by $|W|$. Thus, the probability of not classifying X as a weak cause (i.e., a false negative) given 1 sample will be at least $1 - (\rho^*|F \setminus X|)$ and at most $1 - \rho^*$ since $1 \leq |W| \leq |F \setminus X|$. If sample contingency sets are drawn independently, then the false negative rate using n samples is at least $(1 - (\rho^*|F \setminus X|))^n$ and at most $(1 - \rho^*)^n$. If using domain normalization, as in Algorithm 8, then the same proof works with $\rho^* = \rho^*|\mathcal{D}|$. □

This proposition essentially shows us that when responsibility is high, expected sample efficiency is high (the expected error rate upper bound is low). This makes sense since we expect sets of variables with high responsibility score to be more easily identifiable as weak causes. When $|F \setminus X|$ is large, the maximum size of possible contingency sets $|W|$ is relatively also large, and thus in some cases the probability of picking some assignment $W = w$ that satisfies Definition 1 increases, decreasing the lower bound on the expected false negative rate. Moreover, we may also establish probabilistic bounds on the error of our responsibility score estimates themselves (rather than just the false negative rate) as a function of the number of samples.

Proposition 2.4. *Let n be the number of samples examined by Monte Carlo MeanRESP, ρ^* be the true responsibility score for the set of potential causal variables X, and $k = |F \setminus X|$. Then the probability that the estimated ρ deviates by $\sqrt{\epsilon\rho^*}$ or more is at most $2e^{-\epsilon n/3k}$. That is, $P(|\rho - \rho^*| \geq \sqrt{\epsilon\rho^*}) \leq 2e^{-\epsilon n/3k}$.*

Proof: According to the Chernoff bound, we can write

$$P(|\frac{\rho n}{k} - \frac{\rho^* n}{k}| \geq \delta \frac{\rho^* n}{k}) \leq 2e^{-\delta^2 \rho^* n/3k}. \tag{14}$$

Setting $\delta = \sqrt{\frac{\epsilon}{\rho^*}}$, we get

$$P(|\frac{\rho n}{k} - \frac{\rho^* n}{k}| \geq \frac{\sqrt{\epsilon \rho^*} n}{k}) \leq 2e^{-\epsilon n/3k}. \tag{15}$$

This is equivalent to

$$P(|\rho - \rho^*| \geq \sqrt{\epsilon \rho^*}) \leq 2e^{-\epsilon n/3k}. \tag{16}$$

□

Largely for similar reasons as outlined regarding Proposition 2.3, the accuracy of estimates for ρ^* is sensitive to the size of possible contingency sets, as the exact solution requires enumerating all such sets. Finally, we show a simple relation between Shapley values and responsibility scores.

Proposition 2.5. *The exact Shapley value for a singleton set always upper bounds the exact responsibility score for the same set.*

Proof: In Equation (12), $0 \leq \frac{\phi(x_p)}{1+\beta} \leq 1$. Thus, since the term $(\phi(x_p) - \phi(x_m))$ is just the Shapley value, then ρ^* is always less than or equal to the Shapley value. □

8.1.2 Preliminary Run Time and Memory Bounds.

Table 3 shows some preliminary bounds on resource use. Here, $|S|$ and $|A|$ are the sizes of the state and action spaces, respectively. $|S^0|$ is the size of event layer of the LCG representing the MDP. $|S^0| \leq |S||A|$, and every variable in $|S^0|$ represents a binary value of True or False indicating whether or not a policy may perform an action in a particular state[7]. Thus, as S^0 may represent any possible counterfactual, partial policy, including full policies, $|\mathcal{D}(S^0)| \leq 2^{|S||A|}$. Below, we also use the fact that a power set of Q, $\mathcal{P}(Q)$, contains $2^{|Q|}$ elements, on average a set drawn from $\mathcal{P}(Q)$ contains $|Q|/2$ elements, and the average domain size of all elements of the power set is upper bounded by $\frac{|\mathcal{D}(Q)|}{2^{|Q|/2}}$, as each ground element q has a $\frac{1}{2}$ chance of existing in any given element of the power set.

Intermediate layers of the LCG may have different structure depending on design choices, so our bounds for Algorithm 2 compared to Algorithm 1 are much looser. $\mathcal{V}$ denotes all other endogenous variables, not including those in layer S^0. For an MDP, this may include up to the entire transition function $|T| = |S|^2|A|$ and the entire reward function $|R| = |S|^2|A|$, thus upper bounded [8] by $|\mathcal{V}| \leq 2|S|^2|A|$.

We let $|X|$ be the number of variables that are checked at any point for weak causality. All variables in an MDP are tied to specific states. Thus, the path-finding-between-variables operation in Algorithm 3 can be converted to a smaller, path-finding-between-states problem. Using BFS on a fully connected MDP results in $|S|^3$ operations.

[7]This exact definition could change given a stochastic policy. For example, one may want to know whether actions have above or below some probability of being selected rather than being strictly required or prohibited. The following results hold for any such binary categorization of action probabilities.

[8]In this paper we use $R(s, a)$ to denote the reward function's dependence on both the current action and current state. Other popular notations include $R(s, a, s')$ and $R(s)$, which offer more or less fine-grain control of the reward signal, but do not fundamentally affect the conclusions of this paper. Here we write $|R| = |S|^2|A|$ as it is the worst case.

Table 3. Worst-case resource bounds. We assume nothing about the structure of the MDP, and thus the bounds appear completely intractable. In practice, we find that simple approximations create tractable problems.

Alg.	Res.	Complexity	Bottleneck
1	time	$2^{2\|S\|\|A\|} + 2^{7\|S\|\|A\|/2} + k(\text{Alg. } 2)$	Enumerating causal / contingency sets
	space	$3\|S\|\|A\|2^{\|S\|\|A\|}$	Storing R, R^-
2	time	$\|\mathcal{D}(\mathcal{V})\|^2(2^{3\|\mathcal{V}\|/2} + 2^{3\|\mathcal{V}\|})$	Enumerating causal / contingency sets
	space	$3\|\mathcal{V}\|2^{\|\mathcal{V}\|}$	Storing R
3	time	$\|S\|\lfloor e(\|S\|-2)!\rfloor(\|X\|\|S\|^3 + \|X\|^2 + \|\mathcal{V}\|)$	Finding / checking all paths
	space	$\|S\|$	Storing one path in the LCG
4	time	$\|\mathcal{C}_\mathcal{W}\|(\log(\|\mathcal{C}_\mathcal{W}\|) + \|\mathcal{C}_\mathcal{W}\|\|X\|)$	Subset checks
	space	$\|\mathcal{C}_\mathcal{W}\|\|\mathcal{V}\|$	Storing weak causal sets
6	time	$\|S\|^2(h+1)$	Connectivity checks + Label comparisons
	space	$\|S\|^2$	Storine transition / reachability matrix
8	time	$2^{\|\mathcal{V}\|/2}\|\mathcal{D}(\mathcal{V})\|(2^{3\|\mathcal{V}\|/4} + \|\mathcal{D}(\mathcal{V})\|)$	Enumerating causal / contingency sets
	space	$\|\mathcal{V}\|2^{\|\mathcal{V}\|-1}$	Storing weak causal sets
8_{MC}	time	$\kappa\eta\upsilon\beta_{UB}(2^{\beta_{UB}} + \chi)$	Enumerating causal / contingency sets
	space	$\kappa\|\mathcal{V}\|$	Storing weak causal sets

Checking all $\lfloor e(|S|-2)!\rfloor$ paths for all variables in X takes $|X||S|\lfloor e(|S|-2)!\rfloor$ operations[9] Also recall that $\mathcal{C}_\mathcal{W}$ is the set of all weak causes, from which actual causes may be determined.

In practice, worst-case bounds are very loose. In Algorithms 1 and 2, we check all assignments of X, and $X \subseteq \mathcal{V}$. However, usually $|X| \ll |\mathcal{V}|$, especially after reducing the LCG. Bounds for Algorithm 8 are also poor estimates of in-practice cost, since it uses short-circuiting. Some bounds' tightness depends on the connectivity of the MDP. For example, in Algorithm 6, the bounds assume fully-connected MDPs, but most MDPs are sparse and thus the number of edges $E \ll |S|^2$. Moreover, if h is small compared to the width of the MDP, run time will decrease since nodes labeled ∞ are handled in linear time. There are other possible improvements since theoretically every explanation can be pre-computed, but this is impractical due to the number of possible explanations. Notably, constructing LCGs for each state, regardless of how ϕ and X are specified, and computing connectivity and reachability allows reductions and causal model approximations to be applied quickly online, given X and ϕ. Empirical run-time results are presented in §8.2.4.

8.2 Empirical Analysis

Our empirical analysis covers several topics, including run-time, analysis of metrics, comparison of weak cause definitions and Shapley values, and two user studies comparing our framework to state-of-the-art MDP explanation methods. However, we begin with a case study highlighting the generality and flexibility of our framework.

8.2.1 Case Study: Explanation Diversity.

The purpose of this study is to show how (1) our approach can handle *semantically different* types of causal queries, corresponding to different conceptions of MDP explanation in the literature, and (2) formal definitions of causality identify sensible explanans. Here, we qualitatively examine the correctness of causal attribution in the following simplified MDP domain. Consider a robot navigating the environment depicted in Figure 6. The agent knows: its (x, y) location, time to failure, c, and if its location is normal, ideal for repairs, or hazardous, t. Thus,

[9]Given a complete graph (worst case) of size n with vertices s and t, there are $\sum_{1\leq k\leq n-1} \frac{(n-2)!}{(n-1-k)!}$ total simple paths from s to t. Here, k ranges over path lengths as measured in edges. This sum can be written as $(n-2)!(1 + \frac{1}{1!} + \frac{1}{2!} + \ldots + \frac{1}{(n-2)!})$, which can ultimately be simplified to $\lfloor (n-2)!e \rfloor$.

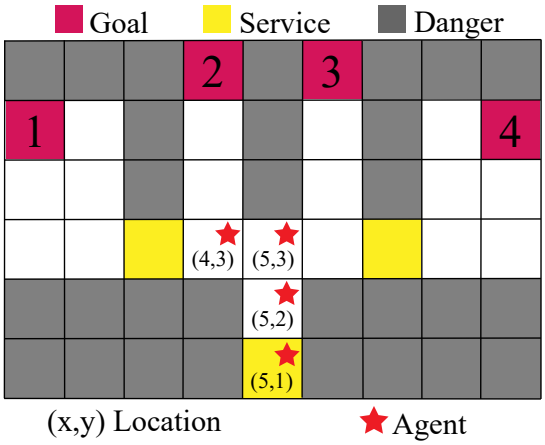


Fig. 6. Example domain.

Table 4. List of scenarios.

ID	Y	s_0 (x, y, t, c)	R_1	R_2	R_3	R_4	R_C	*Action*
1	F	(5,1), R, 5	80	-50	-40	100	2	Repair
2	F	(5,2), N, 4	80	-50	-40	100	2	Up
3	F	(5,3), N, 3	80	-50	-40	100	2	Left
4	F	(4,3), N, 2	80	90	-40	100	-1	Left
5	R	(5,3), N, 3	80	-50	-40	100	-1	Left
6	R	(5,3), N, 3	80	-50	-40	100	2	Left
7	T	(5,3), N, 3	80	80	70	100	-1	Left
8	T	(5,3), N, 3	80	-50	-40	100	-1	Left
9	V	(5,3), N, 3	80	-50	-40	100	-1	Left

the state factors are $x \in \{1, \ldots, 9\}$, $y \in \{1, \ldots, 6\}$, $c \in \{0, \ldots, 5\}$, $t \in \{\text{Normal(N)}, \text{Repair(R)}, \text{Danger(D)}\}$. The actions are $A = \{\text{Up}, \text{Left}, \text{Right}, \text{Repair}\}$. If the agent breaks down and cannot be repaired or visits a hazardous state, it gets a reward of -10. Repairing has a reward of R_C, and reaching the ith goal state yields reward R_i. All other state-action pairs have a reward of -1. Last, transitions are deterministic with two exceptions. When taking action Left at $(5, 3)$, the agent transitions to $(4, 3)$ (with probability $T_L = 0.6$) or $(4, 4)$ (with probability $1 - T_L = 0.4$). When taking action Right at $(5, 3)$, the agent transitions to $(6, 3)$ (with probability $T_R = 0.01$) or $(6, 4)$ (with probability $1 - T_R = 0.99$). For all examples we considered event $\phi = \pi_{s_0}$, and for explanation type Y we set $X = Y$ and apply MeanRESP. Table 4 shows the value of all state factors s_0, rewards, and actions for each scenario. Below, we contextualize MeanRESP's output.

Scenario 1: The cause for action Repair is $t = \text{R}$ ($\rho = 0.64$). Since $R_C > 0$, it is optimal to repair as it prevents failure later and is better than the default reward of -1.

Scenarios 2 and 3: There are two causes of action Up in scenario 2: $t = \text{N}$ ($\rho = 0.50$) and $(x, y) = (5,2)$ ($\rho = 0.26$). Scenario 3 has the same causes, but $\pi(s_0) = \text{Left}$ and (x, y) has $\rho = 0.60$. This is due to the topology at (5,3) compared to (5,2). In both cases, t is a cause since if $t = \text{R}$, $\pi(s_0) = \text{Repair}$.

Scenario 4: The causes for action Left are $c = 2$ ($\rho = 0.85$), $(x, y) = (4,3)$ ($\rho = 0.68$), and $t = \text{N}$ ($\rho = 0.60$). Note that unlike in previous scenarios, time to failure is both a cause and has the highest responsibility score. If the agent were to go directly to goal 2, it will break down at (4,5).

Scenario 5: There are two causes of action Left: R_1 ($\rho = 0.30$) and R_3 ($\rho = 0.55$). Since we bound $\mathcal{R} = [-100, 100]$, no values for R_2 or R_4 can change the outcome. R_4 is already the maximum, and R_2 alone is not a cause due to its relatively weak effect on the expected value of that subtree (transition variables are in $\mathcal{U}$, not $\mathcal{V}$).

Scenario 6: Here, only R_3 ($\rho = 0.55$) is causal. Since $R_C > 0$, the agent exploits this by repeatedly taking service at (3,3). Thus, R_1 alone cannot affect the agent's policy since goals 1 and 4 will never be visited. This is a good example of the proposed approach identifying a possible poorly specified objective.

Scenarios 7 and 8: Since $R_1 = R_2$ in scenario 7, the only cause of the action Left is T_R ($\rho = 0.66$). However, in scenario 8, both T_R ($\rho = 0.66$) and T_L ($\rho = 0.56$) are causes.

Scenario 9: Using MeanRESP with beam search, we find that the most influential set of trajectories lead to goal 1. That is, its value contributes most to the expected value, even though the most likely (p=0.6) outcome of taking action Left at (5,3) reaches goal 2.

Summary: These scenarios show how different sets of explanans provide *semantically distinct* insights into variables' effects on actions. This underscores the utility of flexibility in generating explanations since the 'best' explanans may be unknown pre-deployment. Because existing methods only analyze one component of the MDP, they cannot produce most of these explanations (Table 2).

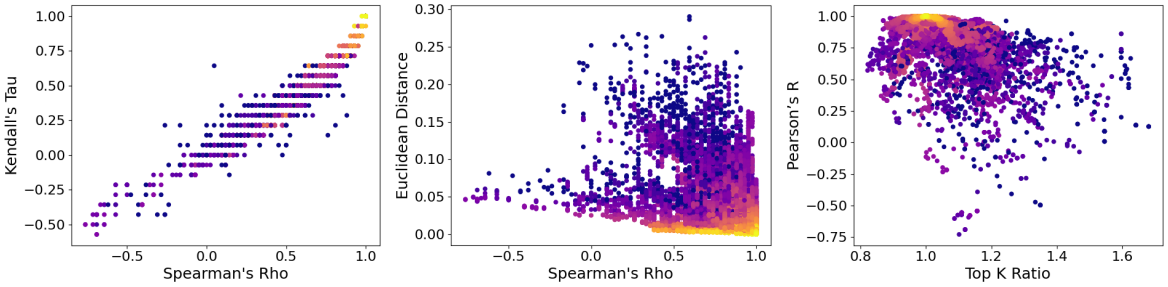
8.2.2 *Evaluating Measures for Comparing Approximate Explanations.*

To better understand how different measures of similarity between two top k lists of explanans might capture different qualitative judgements or distinctions from users, we compare their evolution over time as Monte Carlo MeanRESP takes additional samples. In our experiments, 60 states are sampled from the Lunar Lander[10] MDP, from OpenAI Gym [16], and then both exact and Monte Carlo MeanRESP are run, the latter for a total of 5,000 samples. After each sample, the top k explanans along with their rank and responsibility score are recorded. The proposed measures are then run 5,000 times for each state, measuring the difference between the top k explanans after the ith sample in Monte Carlo MeanRESP and the top k explanans according to exact MeanRESP.

In particular, we are not just interested in the behavior of individual measures, but rather, whether or not any of them capture unique qualities of the top k results that others do not, suggesting increased utility when used in combination. To highlight this, we plot all metrics against each other, bilaterally across several plots. Figure 7 illustrates some of these comparisons. In each sub-figure, two measures are plotted against each other, each comparing the result of Monte Carlo MeanRESP after every sample to the result of exact MeanRESP. Each point in the scatter plots represents a pair of measures comparing an approximate explanation from some state against the exact explanation for that state. The colors correspond to sample order, with dark blue representing samples near the beginning, and light yellow representing samples near 5,000. Because Monte Carlo MeanRESP converges so quickly, the linear color map shown here actually operates on $\log(\lfloor \frac{n}{20} \rfloor)$, where n is the sample number, rather than a direct linear map.

We can see from these figures that some measures, such as the two popular rank correlation measures in Figure 7a, are very similar, likely to the point of being redundant with respect to identifying meaningful changes in explanation quality. However, this is not the case for the other two pairs of measures. For example, in Figure 7b, Euclidean distance rapidly converges to 0, while Spearman's ρ remains well below 1 for some time. This suggests that the exact responsibility scores are quite close in magnitude and thus their order can be changed by small deviations. Depending on the model, this may indicate either an option to terminate earlier than expected if responsibility values are more important, or a need to increase the number of samples if ordinal relationships between causes are more important. Moreover, some pairs of metrics, such as those in Figure 7c, appear to measure orthogonal, equally-scaled phenomena, converging at roughly the same rate but not necessarily in a correlated manner. Using such a pair of metrics may offer a much more robust condition for control decisions

[10] https://gymnasium.farama.org/environments/box2d/lunar_lander/



(a) Kendall's τ v. Spearman's ρ (b) ℓ^2-norm v. Spearman's ρ (c) Pearson's R v. Top k Resp.

Fig. 7. Selection of measure comparisons. Quantization in sub-figure (a) is due to Kendall's τ operating on small lists.

like early termination. For additional experimental details and a comprehensive illustration of all comparisons, please see Appendix B.

These qualitative results suggest that there do not exist single measures that concisely capture the evolution of approximate top k causal sets over time and open several new directions for research, which we propose as interesting open problems. First, more research is needed to understand if and how these measures, others proposed [26], or combinations thereof, might be used to automatically reject obviously bad approximation settings that yield results "far" from ground truth. This would allow automated parameter tuning, and vastly increase the efficiency and impact of user studies by reducing wasted user study resources on clearly flawed systems. Second, such measures may also detect favorable conditions for early termination of Monte Carlo MeanRESP or similar algorithms, creating a more flexible type of "anytime explanation" system. To the best of our knowledge, no such system yet exists. Last, although some of these measures appear in other user-centric applications, such as recommender systems, they are primarily used to analyze data already labeled by users. Here, we propose understanding essentially the opposite direction: performing user evaluation of approximate explanations to understand if, when, and how these measures capture differences that are important to humans. That is, given the difference between two explanations according to one or more measure, can we understand if they are satisfactory for a user?

8.2.3 *MeanRESP and Shapley Values.*

While our framework was originally conceived to explain agent behavior that was the result of planning using MDPs, it may in principle also be applied to black box policy approximators and even function approximators designed for prediction or classification. Given this potential application, it is important to understand how, if at all, our framework differs from those built upon the concept of Shapley values, which is commonly applied in such domains for the purpose of explanation. To this end, we design several experiments to compare the two approaches.

In these experiments, summarized in Figure 8, we find weak causal sets for 60 randomly selected states in four environments: Lunar Lander, Taxi[11], BlackJack[12], and a version of HighwayEnv[13] (highway-fast-v0; KinematicObservation). To compare with a Shapley-value method we implement a representative method [142], and focus on identifying causal variables from the set of state factors. Policies were learned via either value iteration or deep Q-learning, please see Appendix B for more experimental details.

Given the Shapley values and responsibility scores generated for each method, we compare their similarity. Figures 8a, 8b, and 8c show the average Pearson, Kendall, and Spearman correlations, respectively, between the

[11]https://gymnasium.farama.org/environments/toy_text/taxi/
[12]https://gymnasium.farama.org/environments/toy_text/blackjack/
[13]https://github.com/Farama-Foundation/HighwayEnv

outputs of each method. Note that there is no "better" or "best" score. These graphs show only the correlation between outputs of approximate attribution algorithms. Somewhat surprisingly, we can see that the results from different responsibility methods are sometimes less similar than those from Shapley-value methods, and this is true both for ordinal and real-valued correlation metrics.

However, if we look at Figure 8d, we can see that, predictably, OC and UC are more similar when a scale-sensitive measure like Euclidean distance is used, which makes sense given expressions (12) and (13). Perhaps most importantly, it is clear from Figure 8e that the set difference can be substantial even for relatively small, seemingly simple domains. That is, choosing the top k items as explanans, will, on average, yield substantially different results. Here, some of the difference between domains can be explained by our need to set different values of k due to the varying complexity of the domains. For example, the BlackJack domain has only three features, so we use $k = 1$, while $k = 2$ for Taxi, $k = 4$ for Lunar Lander, and $k = 7$ for HighwayEnv. Nevertheless, these differences are not small, suggesting an average difference in causal set contents of approximately 10-20% across all domains.

These results have mixed implications. On the one hand, it appears that there are likely many acceptable technical solutions and definitions for establishing cause or attribution that remain relatively consistent across many domains. On the other hand, when we consider the necessity in most cases of choosing a subset of causes to present in an explanation (either setting k or choosing explanans from within the top k directly, corresponding to Step 3 from the general procedure outlined in §4), these differences may make these choices more difficult in that a larger number of tradeoffs must be considered. For more results on how different definitions of cause affect potential similarity measures as functions of the number of samples taken, please see Appendix B.

8.2.4 Run-Time Analysis and Approximation.

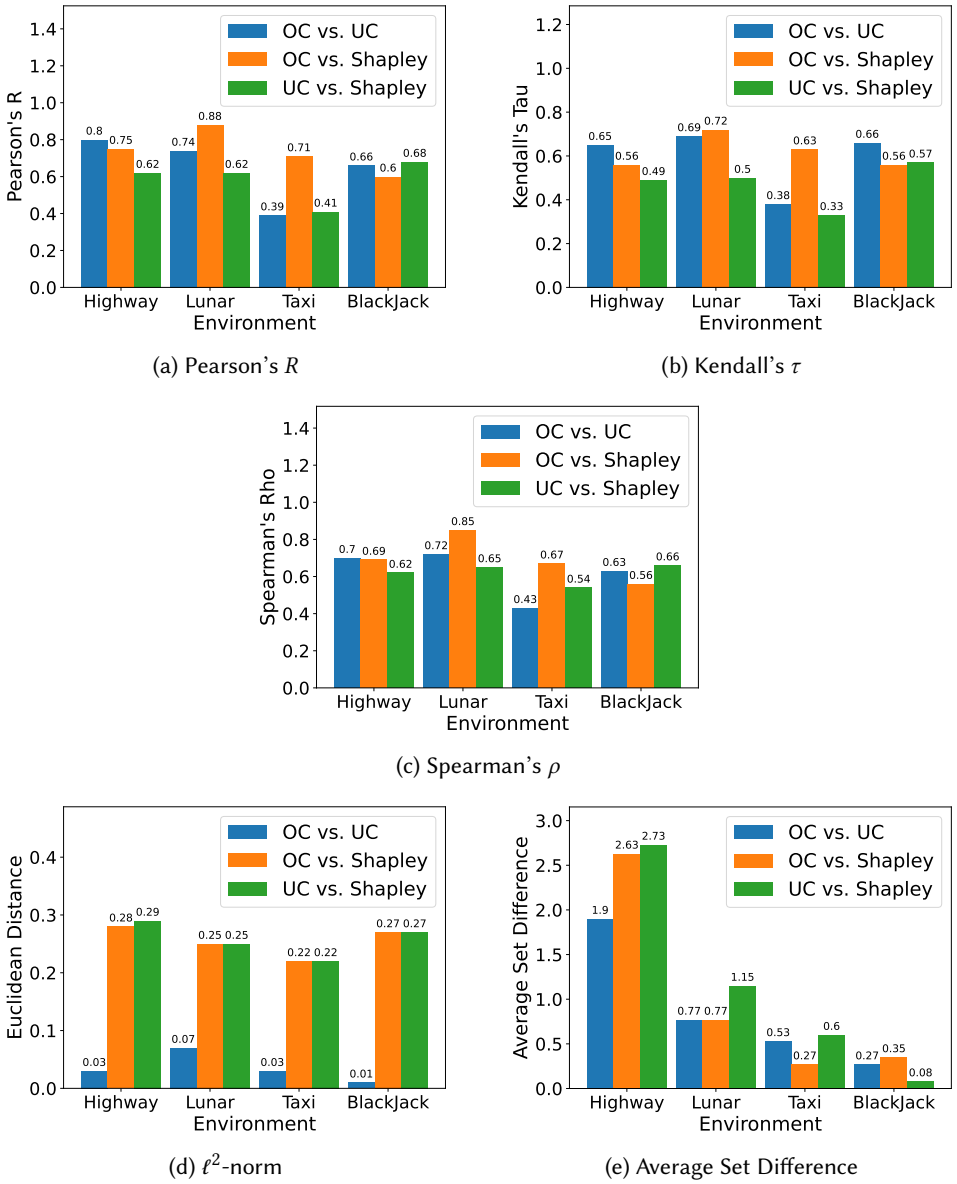
A major concern for the practical application of any method based on causal or counterfactual analysis is whether it remains tractable as the problem size increases. Here, we show that the exact versions of MeanRESP indeed suffer from the curse of dimensionality in enumerating possible counterfactual scenarios. However, Monte Carlo MeanRESP retains remarkable efficiency *and accuracy* even as problem size increases.

Figures 9a and 9b show how the running time of both methods scales as a function of the domain size of the variables and the number of features present, respectively. Lines plotted represent mean run times over 60 trials, and the 95% confidence interval falls within the width of the lines on the plots. All data was generated using different versions of the Lunar Lander domain. For more details on the experimental setup, please see Appendix B.

8.3 User Studies

Explanations are inherently multi-agent and human-centered. That is, all explanations have the common purpose of communicating information from one agent to another, and frequently the effectiveness of the explanation depends on the internal or hidden states of both agents. Moreover, the recipient is nearly always a human. Therefore, automated metrics alone, even those that consider information beyond purely an explanation's contents, are not sufficient for understanding whether explanations generated using our proposed framework, or any other framework, will meet their deployment needs. Given the diversity of potential applications, it is natural that there may be a variety of target effects of an explanation depending on the system. To understand whether our system achieves these effects we ran two user studies, SOTA and CONTEXT. While not designed using Hoffman et al. [2018] as a specific reference, our studies align primarily with two of the four main categories of evaluation they outline: goodness and satisfaction, although our experiments also include some measures beyond these criteria. There are many other possible studies measuring more specifically a user's internal model of the planning algorithm or their ability to perform tasks cooperatively with the agent being explained, where appropriate, but we leave those for future work. To the best of our knowledge, we present the largest user studies to-date measuring both absolute and relative performance of algorithms for explaining MDPs.

In particular, we investigate the following hypotheses:



(a) Pearson's R

(b) Kendall's τ

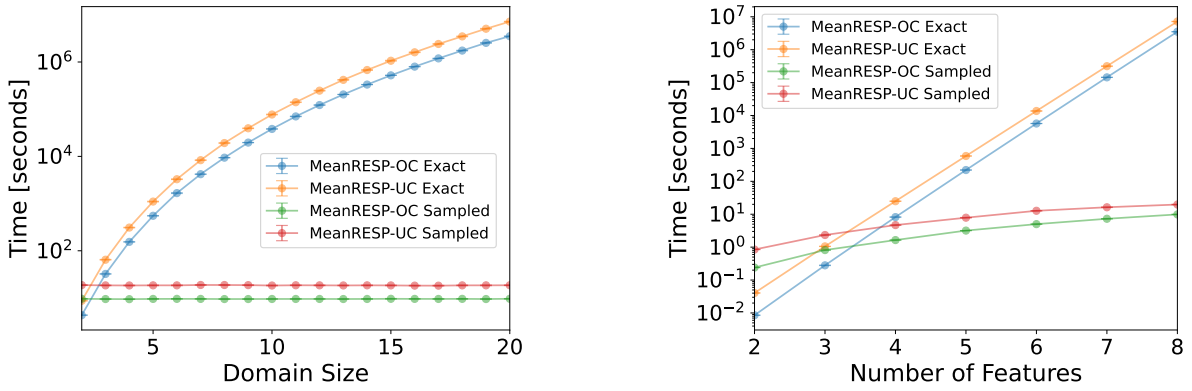
(c) Spearman's ρ

(d) ℓ^2-norm

(e) Average Set Difference

Fig. 8. Comparison of different weak cause (responsibility) definitions and Shapley value attribution. In sub-figures (a)-(c) we show correlation measures, thus a higher value indicates more similarity. In sub-figures (d) and (e) we show difference measures, thus a lower value indicates more similarity.

- **H1:** Users prefer explanations generated using causal reasoning to those generated using heuristics. (SOTA)
- **H2:** Users prefer explanations supported by explanans representing specific types of information. (SOTA)
- **H3:** User preferences for explanations composed of specific types of information depend on the context within which they receive the explanation. (CONTEXT)



(a) Run time as a function of the domain size.

(b) Run time as a function of feature number.

Fig. 9. Timing results for exact and Monte Carlo MeanRESP for original (OC) and updated (UC) definitions. Note the log scale on both vertical axes and 95% CI error bars.

- **H4:** Users differentiate meaningfully between concepts of trust, understanding, and necessity, in the context of an explanation. (CONTEXT)
- **H5:** User preferences for explanation methods or explanan types correlate with demographic or lifestyle indicators. (SOTA and CONTEXT)

8.3.1 Study Descriptions and Administration Overview.

Roughly 200 participants, recruited via the crowd-sourcing platform Prolific[14], participated in each study. Participants were fluent in English, aged between 18 and 65, and consisted of roughly 50% men and 50% women. Both studies had a similar high-level structure: participants were asked for some basic demographic information, followed by study-specific questions, and finally questions about their patterns of use and attitude towards different forms of AI technologies, which we put at the end of the study to avoid biasing participants prior to their evaluation of the explanations in the middle portion of the studies. Both studies were conducted using the same virtual driving domain [79], see Figure 10. Participants were shown short clips of simulated driving scenarios in a randomized order, where a car drives on a highway and changes speeds and lanes based on a policy from an MDP solved offline. After watching each clip, participants are shown one or more automatically generated explanations referring to the behavior of the vehicle shown in the clip, and asked questions about the explanation(s). Clip order and explanation order (for questions involving multiple explanations) were randomized.

The predominant method for communicating automatically generated explanations to humans, Step 4 from the procedure in §4, is via text. While we use natural language templates for conducting our studies, we make no claims about the relative effectiveness of this method compared to other options. To present as little bias towards different explanations as possible, every explanation was presented using the same basic template: "The car <took action> because <explanan 1>, ..., <explanan N>." Each action and explanan in the MDP was mapped to a custom phrase, signifying both what the explanan represented and its value, or the nature of the action. Although many of the explanans generated by other methods *do not* assert causality, at least not theoretically, we still use the conjunction "because" in order to maintain consistency. For a more comprehensive description of administration details, specific questions, and participant demographics for both studies, please see Appendix A.

[14]www.prolific.co

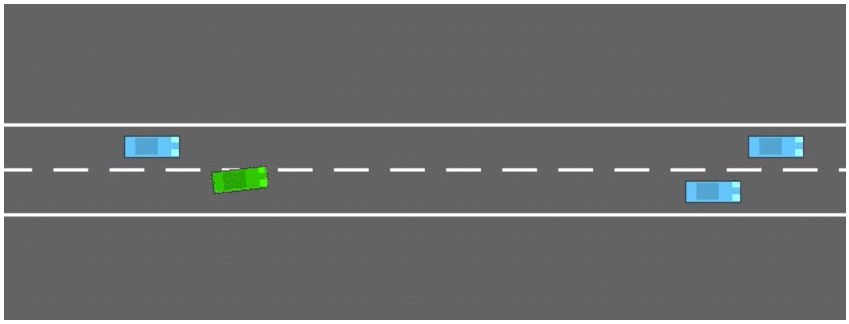


Fig. 10. A screen capture from an example scenario in which the ego car (green) makes a left lane change.

8.3.2 *The SOTA Study.*

Design. Following demographic questions, participants were shown three clips, one at a time. After each clip, showing the agent (car) taking a particular action, such as the left lane change shown in Fig. 10, participants were shown 7 different explanations in a randomized order and asked to rank them relative to each other to produce a strict preference ordering. Specifically, they were asked to "rank the following explanations according to how well you feel they would help someone understand the behavior of the vehicle". Each explanation was generated using a different method for automatic explanation, including three baselines[15]: (F-type) [31], (R-type) [71], and (T-type) [154], as well as all four types of explanation generated by our proposed method. We find remarkably strong evidence in support of **H1** and **H2**. Below we review the results in detail.

H1: Preferences for Causal Explanations. Figure 11 summarizes our findings on user preferences for explanation methods. The most important observation is that, for every explanation type (F, R, T), users prefer the explanations generated via causal reasoning over those generated via heuristic methods. We applied the Mann-Whitney U-test [96] to each pair of generation methods (21 in total), using an initial α-value of 0.5, and a Bonferroni corrected α-value of 0.0024 [12]. We detected the following preference ordering with p-values below 0.0001.

$$1)\ \text{Prop-}F \sim \text{Prop-}V \succ \text{Elizalde} \succ \text{Prop-}R \sim \text{Prop-}T \succ \text{Khan} \succ \text{Wang}$$

Here, $A \succ B$ denotes a strict preference for A over B, and $\sim$ denotes preference equality. We believe the overall preference for causal explanations is due to their consistent relevance across all scenarios. For example, although both heuristic and causal F-type methods have access to the same potential explanans, and occasionally produce the same explanations, there are some cases where the heuristic methods do not produce sensible explanations, such as the following, where the heuristic method fails to provide both relevant and complete information to explain the event in this case:

"The car changed lanes to the right because the car was in the left lane".

In contrast, the method based on causal reasoning more reliably produces explanations that capture more completely the underlying reasons for the observed behavior:

"The car changed lanes to the right because the car was in the left lane, the estimated time to collision in the left lane was 2 seconds, and the right lane was empty".

[15]Some of the state-of-the-art methods we compare against [71, 154] were originally designed to explain certain sub-types of MDPs or POMDPs. In order to run our experiments, we modified these methods as little as possible from their original implementations.

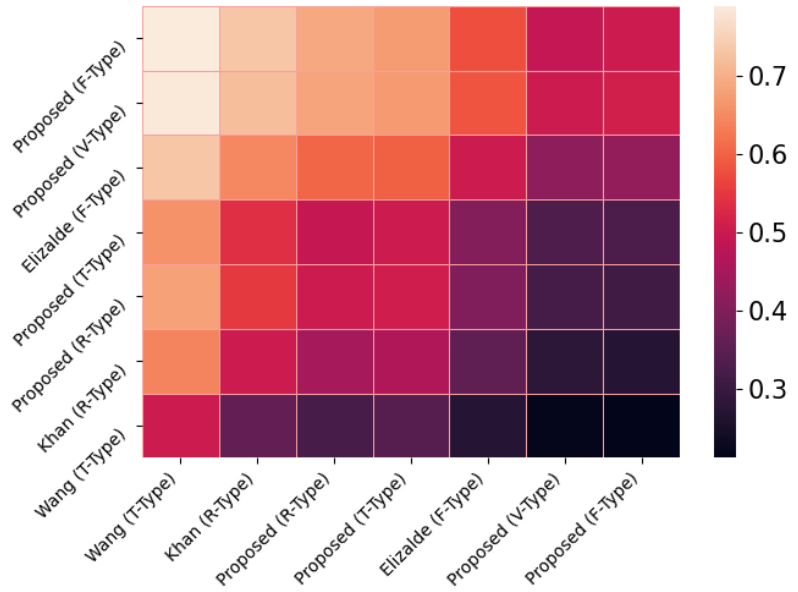


Fig. 11. Preference likelihoods for MDP explanation methods. The color of cell (*row*, *column*) indicates the probability that explanations generated using method *row* are preferred to explanations generated from method *column*.

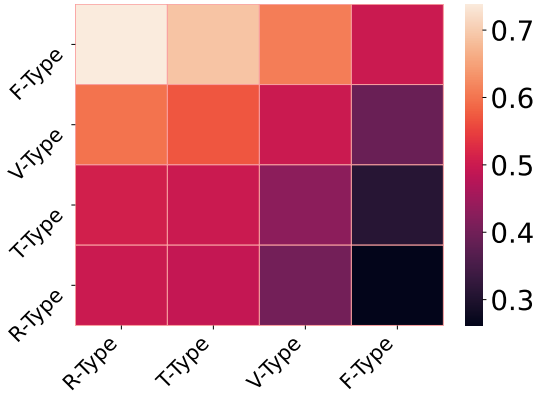


Fig. 12. Preference likelihoods for MDP explanations based on explanan type.

H2: Preferences for Explanan Types. Figure 12 shows a similar analysis with respect to different explanan types (F, R, T, V), where we can see that F-type explanations are, on average, preferred over R-type and T-type. For this analysis if there were multiple explanations based on the same explanans we used the most preferred rank. For example, if a user preferred $A \succ B \succ C$, where A and C were F-type and B was R-type, then we would record F-type as being ranked highest, followed by R-type. We apply the same pair-wise Mann-Whitney analysis as before, now with a total of 6 pairs and a Bonferroni corrected α-value of 0.0083. Following this analysis, we obtain the following preference ordering: F-type $\succ$ V-type $\succ$ R-type $\sim$ T-type with p-value 0.00001.

8.3.3 *The CONTEXT Study.*

As is clear from the results of the previous study, given identical scenarios and identical methods for identifying multiple explanans, humans may exhibit strong preferences over explanations supported by semantically different explanans. This is not surprising given the number of different preferences for explanations already captured in the literature [104]. However, one piece crucially missing from our understanding of these preferences is whether or not they are context dependent [152], and if so, to what degree. Given the generality of our framework, which allows us to pick explanans from virtually any set of variables within a model, we have in a sense bypassed one of the typical gating or filtering mechanisms humans are theorized to use when simulating or constructing hypothetical or counterfactual worlds [68]. This begets a new problem (Step 3 from the procedure outlined in §4) which is to pick only the most preferred explanans for use in the explanation.

Given the possibility for context-dependence and the relatively wide scope of possible contextual variables, we consider the following family of hypotheses. There are factors, such as inherent task risk, level of complexity, power differential, amount of independence, amount of cooperation, etc. that affect relative preferences for types of information (or reasons) referenced in an explanation. Here, we choose to focus on just one small subset of such hypotheses, the context of an autonomous driving scenario.

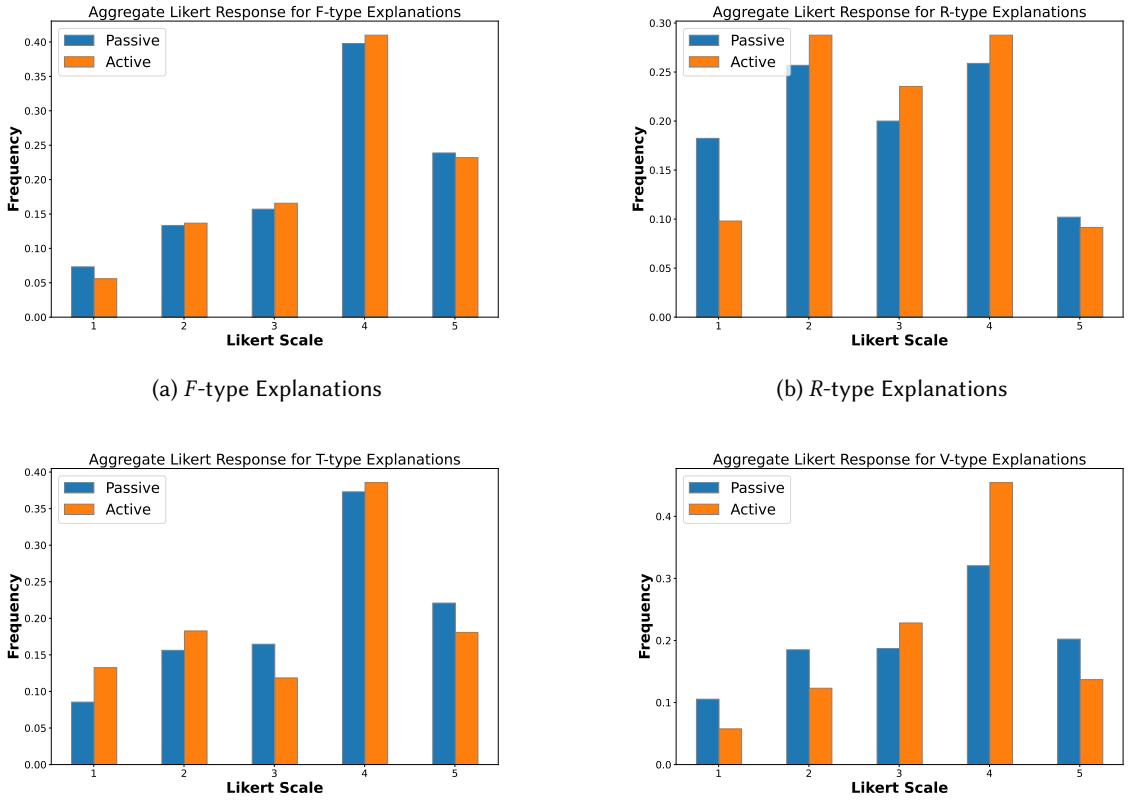
Design. This study split participants into two groups. Group A were told that they were a passenger, with no ability to change the behavior of the car, communicate with the car, or intervene in the navigation or driving process in any way. Group B were told that they were to play the role of a driver. They were not currently in control of the vehicle, but if they wished they could signal their intent to take over control from the autonomous vehicle at any point by hitting a designated key. This design was intended to create two groups of users who were (possibly) interested in different forms of information about the car's operation and thus would exhibit different evaluations of the explanations presented subsequently. For the exact prompts, please see Appendix A.

Following demographic questions, participants were shown several clips, one at a time. After each clip, they were shown a single explanation and asked to rate on a 5-point Likert scale how the explanation affected their trust of the system, understanding of the system, and the necessity of the explanation itself given what was occurring in the simulated scenario. All explanations were generated using our proposed method, and were equally distributed between different types of explanans (F-, R-, T-, and V-type). We find evidence in support of **H3** for V-type explanations, and **H4**, but not **H5**. Below we review the results in detail.

H3: Context-dependent Preferences for Explanan Types. Figure 13 shows the distribution of Likert scores related to trust, understanding, and necessity on all scenarios, conditioned on the type of explanation presented and on whether the user was given the passenger prompt (passive user) or the driver prompt (active user). To each of the four pairs of distributions, we applied the two-sample Kolmogorov-Smirnov test [73, 139], and thus have a Bonferroni corrected α-value of 0.0125. With p-value 0.00530, we find that for V-type explanations, passive and active users exhibited a difference in their evaluations. We also saw some effect for R-type explanations (p-value 0.06026), but we cannot make additional conclusions at this time.

Interestingly, both F-type and V-type explanations make use of the same underlying types of variables (state factors), but differ in how they present them. F-type explanations reference state factor variables and values that are currently true in the scenario, and V-type explanations reference state factor variable values that may or may not be true at the current time. For more example explanations, please see Appendix A.

At a high-level, these results seem to parallel the psychological theory of agency. Identification of agentive entities is theorized to be a key capability for both humans and some animals [38], and one of the primary ways we identify other agents is through observing their actions [19]. Moreover, it has been shown that robots that do not look human can still elicit agency attribution in humans [109]. Such agents are called instrumental agents [37]. So-called communicative agents have a further ability to express their intent via explicit communication [37]. An interesting hypothesis for future consideration is whether V-type explanations elicit context-dependent



(a) F-type Explanations

(b) R-type Explanations

(c) T-type Explanations

(d) V-type Explanations

Fig. 13. Aggregate Likert distributions for different explanan types.

preferences from users due to their increased ability to communicate intent and therefore signal agency. In this case, the causal grounding of the explanations may serve to maintain rationality of the action with respect to the goal, which we know to be another key factor in agency attribution [91].

Another notable psychological theory, prominence-interpretation theory [34], breaks trust-affecting events into two stages, an observation stage, based on the prominence of the feature or event with respect to the user, and an interpretation stage, where the user assesses the affect of their observation on their trust of the system. Whether active users react differently to V-type explanations due to differences in prominence or interpretation is thus an open question, although we hypothesize, given the focused nature of the study, that interpretation plays a larger role. Other studies have looked at the impact of proactive explanations, but do not compare directly to post-hoc explanations [164]. Thus, the utility of proactive explanations to positively affect trust in general is still largely unexplored. We should note also that there is a large volume of existing work related to context-dependent trust and the antecedents of trust under many different conditions. However, to the best of our knowledge, this is the first such study to consider these phenomena with respect to different explanations.

H4: Trust, Necessity, and Understanding. Tables 5 and 6 summarize the results of our correlation tests. Our primary finding is that ratings of increased understanding, increased trust, and the necessity of the explanation

Table 5. Correlation of understanding, trust, and necessity across all user types.

Type	Trust v. Und.	Trust v. Nec.	Und. v. Nec.
F	0.6682	0.6643	0.7296
R	0.6086	0.6084	0.6082
T	0.7247	0.6985	0.7480
V	0.6748	0.5256	0.6202
ALL	0.6796	0.6405	0.6931

Table 6. Correlation of understanding, trust, and necessity across all explanation types

User Type	Trust v. Und.	Trust v. Nec.	Und. v. Nec.
Active	0.7000	0.6298	0.7128
Passive	0.6619	0.6497	0.6754

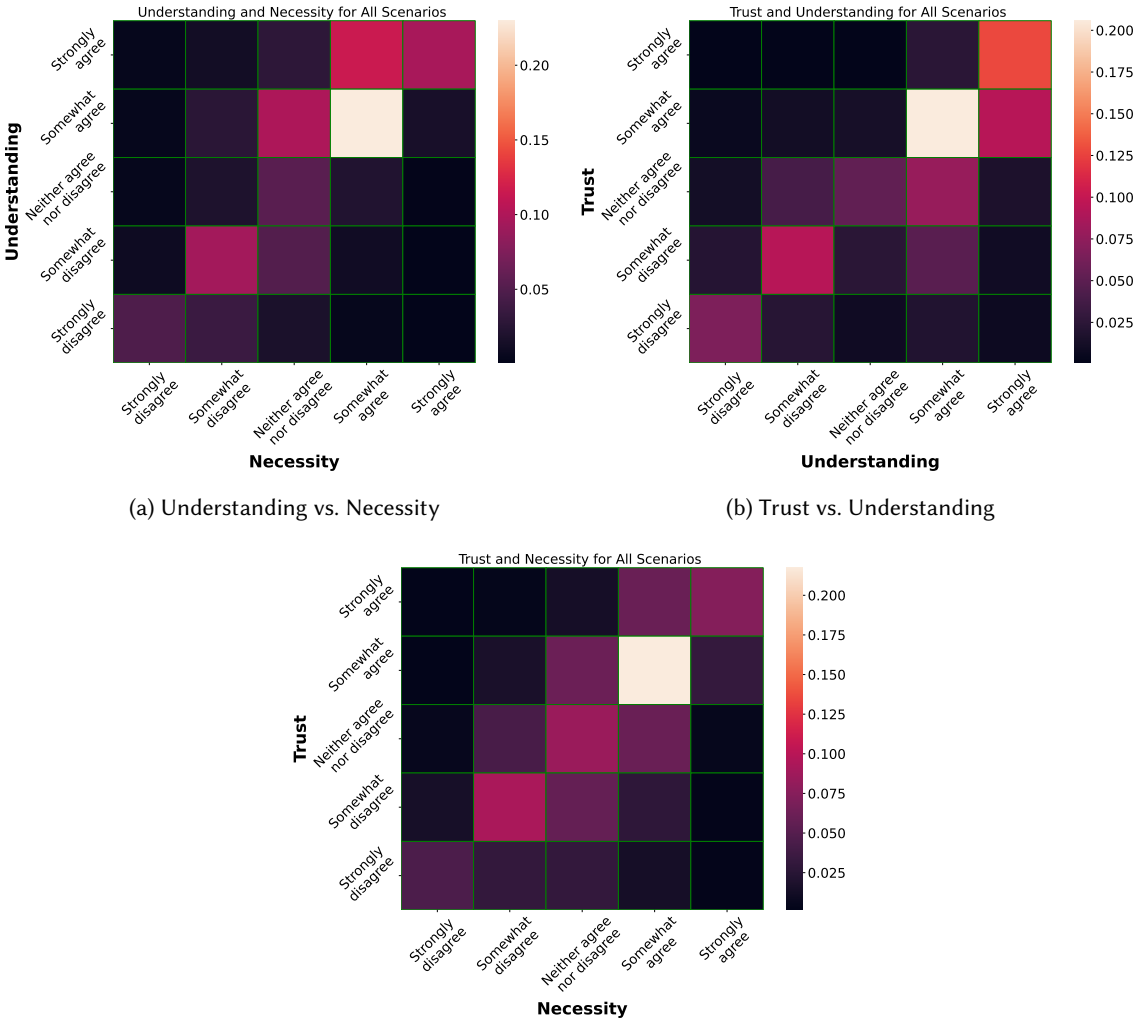
are highly correlated, with the minimum and maximum r-values of 0.5034 and 0.7729 for all combinations of user and explanation types, respectively (see Appendix B for Figures). Moreover, the mode response across all questions was "somewhat agree", indicating a positive effect on trust and understanding given the explanations, and a modest desire (necessity) for the explanations provided. These results are expected, given the previously established connections between desire for explanations and their typical effects on understanding and trust.

Our second and more interesting finding is that explanations that increase understanding may not always be deemed necessary and may not always increase trust to the same degree that they do understanding. That is, in Figure 14, we can see Figures 14a and 14b show strong correlation but have a slope that clearly differs from 1, indicating that response magnitudes frequently differed, while Figure 14c shows data representing the null hypothesis, that measures of trust and necessity for different explanations scale equally.

We see these results as additional support for a number of conclusions established within a large body of work on the antecedents of trust [69]. Principally, these include the importance of understanding, predictability, and competence in the formation of trust [77, 99], and that these results hold for both passive and active users [160]. There have also been results suggesting that different types of explanations may impact different beliefs about trust [155]. Trustor models of trustee motivations may also drive development of trust [49], and there is an important distinction made between a decrease in trust due to lack of competence (forgiven more easily) versus a lack of benevolence or honesty (not as easily forgiven) [117]. Although the types of explanations in these studies do not match exactly with Definition 5, our framework and results offer a promising initial direction to study more advanced hypotheses about the effect of explanations on trust under a variety of conditions, beyond the established wisdom that they have a generally positive impact.

H5: Demographic and Lifestyle Non-Impact. We found no instance in which we could reject the null hypothesis with respect to **H5**. That is, the results presented with respect to **H1-H4** are consistent across genders, age groups, and rates of technology use. We also, somewhat surprisingly, found these results to be consistent regardless of the frequency with which participants operated motor vehicles. We also tried several unsupervised clustering methods to check for more complex correlations between user demographics and preferences for different types of explanations, including principal component analysis (PCA) [123] and t-distributed stochastic neighbor embedding (TSNE) [59], but did not find anything significant. This should give practitioners confidence that these results will hold in many settings.

These results primarily index particularized trust [133]. That is, trust established between specific entities. This is generally accepted to be distinct from generalized trust [25], which is considered to be an individual's natural predisposition to trust, and which has been repeatedly established to be contingent on many demographic



(a) Understanding vs. Necessity

(b) Trust vs. Understanding

(c) Trust vs. Understanding

Fig. 14. Distributions of Likert responses on the effect of explanations.

factors including age [161], gender [153], religion [118], and socioeconomic class [47]. We should also note for completeness that sociologists have developed many understandings and models of trust in addition to those proposed by psychologists, and the particularized-generalized trust dichotomy is not universally accepted [133].

9 Discussion and Conclusion

9.1 Study Limitations

Overall, the results, especially for **H1** and **H2**, are exceptionally strong. However, we should address some key limitations of this study. First, self-reporting, and specifically self-reporting of trust, has known weaknesses as an experimental practice [72]. Second, the simulations lacked significant realism. While more immersive driving

simulators exist, recruiting a similar number of participants to do in-person studies was not feasible from a cost perspective. It is possible some effects were not detectable in this study that would be under more realistic autonomous driving conditions.

9.2 Inference and Conceptual Limitations

Because they do not encompass repeated interaction with our system, our experiments offer limited evidence for or against popular models for the dynamics of trust formation between humans and automated systems from psychology. While there are key differences among the most established, there is some level of consensus that 1) trust is established over time, and 2) reasons for trust in a specific interaction can come from several qualitatively different sources. For example, one model, advanced by Marsh and Dibben [2003] identifies dispositional, situational, and learned trust as distinct sources. These represent one's natural propensity to trust, the context of an interaction (including both environment variables and transient mood or attitude variables), and past experiences with similar agents. See Hoff and Bashir [2015] for a more extensive discussion.

Others theorize that the underlying reasons for trust shift over time. In particular, it has been proposed that between humans trust is initially justified by the predictability of the trustee, is later based on the trustee's dependability, and is finally based on faith in the trustee [126]. Zuboff [1988] and Muir, Muir [1987, 1994] have proposed a similar evolution of trust between humans and technology, beginning with experience, followed by understanding, and finally, faith. However, later studies have contended that this order is actually reversed for some human-automation trust relationships [110, 115, 93].

Lee and Moray [1992] proposed reasons supported by performance, process, and purpose as the foundation of trust, representing different types of information. For example, performance estimates may be established from direct observation, process understanding can be developed through familiarity with the underlying mechanisms, and purpose may be imputed from the system's intended use. This may explain why users with no experience but an understanding of the purpose of a system initially trust based on faith [60]. This model, although not directly mapping onto the different types of explanations outlined in Definition 5, is a promising theoretical meeting point.

These models are not mutually exclusive, and regardless of the specifics, they all suggest a complex set of dynamics governing trust formation. While studies like the ones we present make some progress on understanding the effectiveness of causal explanations on trust and understanding in an isolated interaction, to confidently support or reject any of the theories summarized above, longitudinal studies are likely required. Fortunately, embodied agents such as autonomous vehicles, or other robotic systems operating in the open world, afford a rich experimental domain. Previous work has identified that automated information acquisition, information analysis, decision selection, and action implementation play an important role in the dynamics of trust in automation [122], and such systems perform all of these functions at different levels of natural transparency, in addition to eliciting some of the strongest effects [69].

Moreover, there are many different definitions of trust [138], and it is likely that study participants will vary in their theoretical construction of 'trust'. To what extent this variance affects experimental results that rely on self reporting is, to the best of our knowledge, unknown. In practice, it may be possible to avoid forming consensus on abstract notions of trust by disentangling it from certain observable behavior that is more straightforward to label. For example,Ajzen [1980] developed a framework in which beliefs and perceptions (available information) inform attitudes, which, in turn, affect intentions and finally behaviors. As this framework distinguishes between beliefs, attitudes, intentions, and behavior, it can model the influence of trust on reliance. In this model, trust is used as a heuristic to aid in establishing an appropriate level of reliance on a system with which we may lack experience. Given trust is the belief and reliance is the action, trust is thus an antecedent of reliance; additionally, reliance as the behavior can sometimes be observed directly. However, in order to gain experience and grow

trust, some form of reliance must initially exist. Lee and See [2004] discuss this apparent contradiction and many other key central ideas relevant to trust in automation research in their excellent survey paper.

Others from philosophy have argued that trust and reliance may be distinguished since trust may be betrayed while reliance may only be disappointed [8]. For example, reliance on a clock does not produce feelings of betrayal should it break [100], and thus it is the violation of trust alone which elicits a feeling of betrayal [50]. This raises a question for future research on trust in any form in automated systems. As many academic conceptions of trust require competing objectives, without competing objectives, often captured as different theoretical constructions of rational action, there is no need for trust. Moreover, trust requires that an agent may *choose* to prioritize one objective (its own) over another. Robots occupy a strange middle ground: they clearly have objectives and pursue them rationally to the extent they are capable, and thus are obviously agents. However, they are designed specifically for their ability to aid people. Thus, their objectives are necessarily aligned with our own to the extent that we can model and program accurately[16]. They have no ability to choose based on emotion or other motivations. Their success or lack thereof in pursuit of their objective can be attributed purely to their level of competence, and thus reliance seems to be the more appropriate construct to study, whether influenced by attitudes of trust or not.

One might try to differentiate between a bug in the software due to programmer incompetence, a flaw in the design or model due to developer oversight, or an aspect of the behavior that is intentionally malicious. In this case, trust may be placed in the human agents responsible for designing, developing, and deploying the system, but this distinction ought to be made clearly, and at the moment is often left implicit or unquestioned.

9.3 Conclusion and Future Work

We present a novel framework for causal analysis of MDPs using SCMs, motivated by generating explanations of MDP agent behavior. Principally, this framework provides (1) a theoretical foundation for explainable sequential decision making, and (2) simultaneous support for causal queries using different decision problem components, which has previously not been possible. Beyond the proposed framework, this paper presented a set of theoretical and empirical analyses regarding several important properties of approximate MeanRESP related to convergence rates, error rates, similarity and distinction from other methods, and quality of approximation. We also presented two different user studies investigating several hypotheses, the most striking of which is users' overwhelming preference for explanations generated using causal analysis compared to heuristic methods.

We see several promising directions for future work. These include extending this framework to other decision-making models, running longer or more realistic studies of autonomous driving, and using metric information online to produce anytime explanation systems for models which are much larger than the ones tested in this paper. Additionally, further research is needed to understand context-dependent preferences for different types of explanations — particularly by exploring additional contextual variables such as risk level, degree of cooperation, and task complexity.

Acknowledgments

This work was supported in part by the United States National Science Foundation grants #1954782, #2205153, and #2321786. Claudia Goldman was partially supported by the David Goldman Data-Driven Innovation Research Center at the Hebrew University Business School and was affiliated with General Motors, Technical Center Israel, when this research was performed.

[16]This is, of course, an important qualification. Here, we are avoiding possible complications where a robot may be attempting to meet the goals of multiple users that may conflict. There is significant research on how systems ought to reason about such conflicts and about the consequences of their actions, including in decision-making models like MDPs [146, 113, 87]

References

[1] I. Ajzen. 1980. *Understanding attitudes and predictiing social behavior*. Prentice-hall.

[2] M. S. Alam and Y. Xie. 2022. Appley: Approximate shapley value for model explainability in linear time. In *IEEE International Conference on Big Data (BigData)*, 95–100.

[3] E. Albini, J. Long, D. Dervovic, and D. Magazzeni. 2022. Counterfactual Shapley additive explanations. In *ACM Conference on Fairness, Accountability, and Transparency*, 1054–1070.

[4] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez, and K. Kochut. 2017. Text summarization techniques: A brief survey. In *arXiv preprint arXiv:1707.02268*.

[5] E. Altman. 2000. Applications of Markov decision processes in communication networks: A survey. Tech. rep. INRIA.

[6] Y. Amitai, G. Avni, and O. Amir. 2022. Interactive explanations of agent behavior. In *ICAPS 2022 Workshop on Explainable AI Planning*.

[7] S. Anjomshoae, A. Najjar, D. Calvaresi, and K. Främling. 2019. Explainable agents and robots: Results from a systematic literature review. In *18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 1078–1088.

[8] A. Baier. 2014. Trust and antitrust. In *Feminist Social Thought*. Routledge, 604–629.

[9] R. Bellman. 1952. On the theory of dynamic programming. *National Academy of Sciences of the United States of America*, 38, 8, 716.

[10] L. Bertossi, J. Li, M. Schleich, D. Suciu, and Z. Vagena. 2020. Causality-based explanation of classification outcomes. In *arXiv preprint arXiv:2003.06868*.

[11] J. Bertram and P. Wei. 2018. Explainable deterministic MDPs. In *arXiv preprint arXiv:1806.03492*.

[12] C. Bonferroni. 1936. Teoria statistica delle classi e calcolo delle probabilita. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze*, 8, 3–62.

[13] S. Bongers, P. Forré, J. Peters, and J. M. Mooij. 2021. Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49, 5, 2885–2915.

[14] R. J. Boucherie and N. M. Van Dijk. 2017. *Markov decision processes in practice*. Vol. 248. Springer.

[15] T. Brázdil, K. Chatterjee, M. Chmelık, A. Fellner, and J. Křetınskỳ. 2015. Counterexample explanation by learning small strategies in Markov decision processes. In *27th International Conference on Computer Aided Verification (CAV)*, 158–177.

[16] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. 2016. OpenAI Gym. *ArXiv*, abs/1606.01540.

[17] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock. 2021. Explainable machine learning in credit risk management. *Computational Economics*, 57, 203–216.

[18] R. Bustin and C. V. Goldman. 2024. Structure and reduction of MCTS for explainable-AI. In *arXiv preprint arXiv:2408.05488*.

[19] S. Carey and E. Spelke. 1994. Domain-specific knowledge and conceptual change. *Mapping the mind: Domain specificity in cognition and culture*, 169, 200.

[20] T. Chakraborti, A. Kulkarni, S. Sreedharan, D. E. Smith, and S. Kambhampati. 2019. Explicability? legibility? predictability? transparency? privacy? security? the emerging landscape of interpretable agent behavior. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 86–96.

[21] T. Chakraborti, S. Sreedharan, and S. Kambhampati. 2020. The emerging landscape of explainable automated planning & decision making. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 4803–4811.

[22] T. Chakraborti, S. Sreedharan, Y. Zhang, and S. Kambhampati. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In *arXiv preprint arXiv:1701.08317*.

[23] J. Y. Chen, S. G. Lakhmani, K. Stowers, A. R. Selkowitz, J. L. Wright, and M. Barnes. 2018. Situation awareness-based agent transparency and human-autonomy teaming effectiveness. *Theoretical Issues in Ergonomics Science*, 19, 3, 259–282.

[24] H. Chockler and J. Y. Halpern. 2004. Responsibility and blame: A structural-model approach. *Journal of Artificial Intelligence Research*, 22, 93–115.

[25] L. L. Couch and W. H. Jones. 1997. Measuring levels of trust. *Journal of research in personality*, 31, 3, 319–336.

[26] S. Dasgupta, N. Frost, and M. Moshkovitz. 2022. Framework for evaluating faithfulness of local explanations. In *International Conference on Machine Learning*, 4794–4815.

[27] E. David Wong. 1993. Understanding the generative capacity of analogies as a tool for explanation. *Journal of Research in Science Teaching*, 30, 10, 1259–1272.

[28] A. D. Dragan, K. C. Lee, and S. S. Srinivasa. 2013. Legibility and predictability of robot motion. In *International Conference on Human-Robot Interaction (HRI)*, 301–308.

[29] R. Eifler, M. Steinmetz, A. Torralba, and J. Hoffmann. 2021. Plan-space explanation via plan-property dependencies: Faster algorithms & more powerful properties. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 4091–4097.

[30] T. Eiter and T. Lukasiewicz. 2006. Causes and explanations in the structural-model approach: Tractable cases. *Artificial Intelligence*, 170, 6-7, 542–580.

[31] F. Elizalde, E. Sucar, J. Noguez, and A. Reyes. 2009. Generating explanations based on Markov decision processes. In *Mexican International Conference on Artificial Intelligence*.

[32] R. Fagin, R. Kumar, and D. Sivakumar. 2003. Comparing top k lists. *SIAM Journal on Discrete Mathematics*, 17, 1, 134–160.
[33] M. Finkelstein, L. Liu, Y. Kolumbus, D. C. Parkes, J. S. Rosenschein, S. Keren, et al. 2022. Explainable reinforcement learning via model transforms. *Advances in Neural Information Processing Systems*, 35, 34039–34051.
[34] B. J. Fogg. 2003. Prominence-interpretation theory: Explaining how people assess credibility online. In *CHI'03 Extended Abstracts on Human Factors in Computing Systems*, 722–723.
[35] M. Fox, D. Long, and D. Magazzeni. 2017. Explainable planning. In *arXiv preprint arXiv:1709.10256*.
[36] D. A. Freedman. 2007. Statistical models for causation. In *The SAGE Handbook of Social Science Methodology*, 127–146.
[37] G. Gergely. 2010. *Kinds of agents: The origins of understanding instrumental and communicative agency*. Wiley Online Library, 76–105.
[38] G. Gergely and G. Csibra. 2003. Teleological reasoning in infancy: The naıve theory of rational action. *Trends in Cognitive Sciences*, 7, 7, 287–292.
[39] L. Giordano and C. Schwind. 2004. Conditional logic of actions and causation. *Artificial intelligence*, 157, 1-2, 239–279.
[40] C. V. Goldman and M. Baltaxe. 2021. Why are you predicting this class? In *2021 IEEE Intelligent Vehicles Symposium (IV)*, 415–420.
[41] C. V. Goldman, M. Baltaxe, D. Chakraborty, J. Arinez, and C. E. Diaz. 2023. Interpreting learning models in manufacturing processes: Towards explainable AI methods to improve trust in classifier predictions. *Journal of Industrial Information Integration*, 33.
[42] A. Grastien, C. Benn, and S. Thiébaux. 2021. Computing plans that signal normative compliance. In *AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, 509–518.
[43] J. Halpern. 2015. A modification of the Halpern-Pearl definition of causality. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 3022–3033.
[44] J. Y. Halpern and J. Pearl. 2001. Causes and explanations: A structural-model approach — Part I: Causes. In *Seventeenth Conference on Uncertainty in Artificial Intelligence*, 194–202.
[45] J. Y. Halpern and J. Pearl. 2005. Causes and explanations: A structural-model approach. Part I: Causes. *The British Journal for the Philosophy of Science*, 52, 3.
[46] J. Y. Halpern and J. Pearl. 2005. Causes and explanations: A structural-model approach. Part II: Explanations. *The British Journal for the Philosophy of Science*, 56, 4, 889–911.
[47] T. Hamamura. 2012. Social class predicts generalized trust but only in wealthy societies. *Journal of Cross-Cultural Psychology*, 43, 3, 498–509.
[48] L. Hammond, J. Fox, T. Everitt, R. Carey, A. Abate, and M. Wooldridge. 2023. Reasoning about causality in games. In *arXiv preprint arXiv:2301.02324*.
[49] R. Hardin. 2002. *Trust and trustworthiness*. Russell Sage Foundation.
[50] K. Hawley. 2014. Trust, distrust and commitment. *Noûs*, 48, 1, 1–20.
[51] B. Hayes and J. A. Shah. 2017. Improving robot controller transparency through autonomous policy explanation. In *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 303–312.
[52] J. He, S. L. Baxter, J. Xu, J. Xu, X. Zhou, and K. Zhang. 2019. The practical implementation of artificial intelligence technologies in medicine. *Nature Medicine*, 25, 1, 30–36.
[53] L. He, N. Aouf, and B. Song. 2021. Explainable deep reinforcement learning for UAV autonomous path planning. *Aerospace Science and Technology*, 118, 107052.
[54] F. Heider. 1958. *The psychology of interpersonal relations*. Wiley, New York.
[55] G. Hesslow. 1988. The problem of causal selection. In *Contemporary Science and Natural Explanation: Commonsense Conceptions of Causality*, 11–32.
[56] D. J. Hilton. 1990. Conversational processes and causal explanation. *Psychological Bulletin*, 107, 1, 65.
[57] D. J. Hilton. 1996. Mental models and causal explanation: Judgements of probable cause and explanatory relevance. *Thinking & Reasoning*, 2, 4, 273–308.
[58] D. J. Hilton and B. R. Slugoski. 1986. Knowledge-based causal attribution: The abnormal conditions focus model. *Psychological Review*, 93, 1, 75.
[59] G. E. Hinton and S. Roweis. 2002. Stochastic neighbor embedding. *Advances in Neural Information Processing Systems*, 15.
[60] J.-M. Hoc. 2000. From human–machine interaction to human–machine cooperation. *Ergonomics*, 43, 7, 833–843.
[61] K. A. Hoff and M. Bashir. 2015. Trust in automation: Integrating empirical evidence on factors that influence trust. *Human Factors*, 57, 3, 407–434.
[62] R. R. Hoffman, S. T. Mueller, G. Klein, and J. Litman. 2018. Metrics for explainable AI: Challenges and prospects. In *arXiv preprint arXiv:1812.04608*.
[63] T. Huber, K. Weitz, E. André, and O. Amir. 2021. Local and global explanations of agent behavior: Integrating strategy summaries with saliency maps. *Artificial Intelligence*, 301.
[64] Y. Iwasaki and H. A. Simon. 1986. Causality in device behavior. *Artificial Intelligence*, 29, 1, 3–32.
[65] B. Jacobs, A. Kissinger, and F. Zanasi. 2019. Causal inference by string diagram surgery. In *International Conference on Foundations of Software Science and Computation Structures*, 313–329.

[66] A. Jacovi, A. Marasović, T. Miller, and Y. Goldberg. 2021. Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in AI. In *ACM Cconference on Fairness, Accountability, and Transparency*, 624–635.

[67] Z. Juozapaitis, A. Koul, A. Fern, M. Erwig, and F. Doshi-Velez. 2019. Explainable reinforcement learning via reward decomposition. In *IJCAI/ECAI Workshop on Explainable Artificial Intelligence*.

[68] D. Kahneman and A. Tversky. 1981. *The simulation heuristic*. National Technical Information Service.

[69] A. D. Kaplan, T. T. Kessler, J. C. Brill, and P. Hancock. 2023. Trust in artificial intelligence: Meta-analytic findings. *Human Factors*, 65, 2, 337–359.

[70] A.-H. Karimi, B. Schölkopf, and I. Valera. 2021. Algorithmic recourse: From counterfactual explanations to interventions. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, 353–362.

[71] O. Khan, P. Poupart, and J. Black. 2009. Minimal sufficient explanations for factored Markov decision processes. In *International Conference on Automated Planning and Scheduling (ICAPS)*.

[72] S. C. Kohn, E. J. de Visser, E. Wiese, Y.-C. Lee, and T. H. Shaw. 2021. Measurement of trust in automation: A narrative review and reference guide. *Frontiers in Psychology*, 12, 604977.

[73] A. N. Kolmogorov. 1933. Sulla determinazione empirica di una legge didistribuzione. *Giorn Dell'inst Ital Degli Att*, 4, 89–91.

[74] B. Krarup, M. Cashmore, D. Magazzeni, and T. Miller. 2019. Model-based contrastive explanations for explainable planning. In *ICAPS Workshop on Explainable Planning*.

[75] I. E. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler. 2020. Problems with Shapley-value-based explanations as feature importance measures. In *International Conference on Machine Learning*, 5491–5500.

[76] J. Lee and N. Moray. 1992. Trust, control strategies and allocation of function in human-machine systems. *Ergonomics*, 35, 10, 1243–1270.

[77] J. D. Lee and N. Moray. 1994. Trust, self-confidence, and operators' adaptation to automation. *International Journal of Human-Computer Studies*, 40, 1, 153–184.

[78] J. D. Lee and K. A. See. 2004. Trust in automation: Designing for appropriate reliance. *Human Factors*, 46, 1, 50–80.

[79] E. Leurent. 2018. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env. (2018).

[80] D. Lewis. 1974. Causation. *The Journal of Philosophy*, 70, 17, 556–567.

[81] L. Li, T. J. Walsh, and M. L. Littman. 2006. Towards a unified theory of state abstraction for MDPs. In *International Symposium on Artificial Intelligence and Mathematics*.

[82] M. P. Linegang, H. A. Stoner, M. J. Patterson, B. D. Seppelt, J. D. Hoffman, Z. B. Crittendon, and J. D. Lee. 2006. Human-automation collaboration in dynamic mission planning: A challenge requiring an ecological approach. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50, 23, 2482–2486.

[83] P. Lipton. 1990. Contrastive explanation. *Royal Institute of Philosophy Supplements*, 27, 247–266.

[84] T. Lombrozo. 2010. Causal-explanatory pluralism: How intentions, functions, and mechanisms influence causal ascriptions. *Cognitive Psychology*, 61, 4, 303–332.

[85] T. Lombrozo. 2012. Explanation and abductive inference. In *The Oxford Handbook of Thinking and Reasoning*. K. J. Holyoak and R. G. Morrison, (Eds.), 260–276.

[86] T. Lombrozo. 2006. The structure and function of explanations. *Trends in Cognitive Sciences*, 10, 10, 464–470.

[87] Q. Lu, J. Svegliato, S. B. Nashed, S. Zilberstein, and S. Russell. 2024. Ethically compliant autonomous systems under partial observability. In *International Conference on Robotics and Automation (ICRA)*, 16229–16235.

[88] A. Lucic, H. Haned, and M. de Rijke. 2020. Why does my model fail? Contrastive local explanations for retail forecasting. In *ACM Conference on Fairness, Accountability, and Transparency*, 90–98.

[89] S. M. Lundberg and S.-I. Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774.

[90] S. M. Lundberg et al. 2018. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering*, 2, 10.

[91] Y. Luo and R. Baillargeon. 2005. Can a self-propelled box have a goal? Psychological reasoning in 5-month-old infants. *Psychological Science*, 16, 8, 601–608.

[92] J. L. Mackie. 1980. *The cement of the universe: A study of causation*. Clarendon Press.

[93] P. Madhavan and D. A. Wiegmann. 2007. Similarities and differences between human–human and human–automation trust: An integrative review. *Theoretical Issues in Ergonomics Science*, 8, 4, 277–301.

[94] P. Madumal, T. Miller, L. Sonenberg, and F. Vetere. 2020. Explainable reinforcement learning through a causal lens. In *AAAI Conference on Artificial Intelligence*, 2493–2500.

[95] S. Makridakis. 2017. The forthcoming artificial intelligence (AI) revolution: Its impact on society and firms. *Futures*, 90, 46–60.

[96] H. B. Mann and D. R. Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, 18, 1, 50–60.

[97] J. Marques-Silva and A. Ignatiev. 2022. Delivering trustworthy AI through formal XAI. In *AAAI Conference on Artificial Intelligence*, 12342–12350.

[98] S. Marsh and M. R. Dibben. 2003. The role of trust in information science and technology. *Annual Review of Information Science and Technology (ARIST)*, 37, 465–98.

[99] A. J. Masalonis and R. Parasuraman. 2003. Effects of situation-specific reliability on trust and usage of automated air traffic control decision aids. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*.

[100] C. McLeod. 2021. Trust. In *The Stanford Encyclopedia of Philosophy*. (Fall 2021 ed.). Metaphysics Research Lab, Stanford University.

[101] J. E. Mercado, M. A. Rupp, J. Y. Chen, M. J. Barnes, D. Barber, and K. Procci. 2016. Intelligent agent transparency in human-agent teaming for Multi-UxV management. *Human Factors*, 58, 3, 401–415.

[102] L. Merrick and A. Taly. 2020. The explanation game: Explaining machine learning models using Shapley values. In *Machine Learning and Knowledge Extraction*, 17–38.

[103] S. Michel, P. Triantafillou, and G. Weikum. 2005. KLEE: A framework for distributed top-k query algorithms. In *International Conference on Very Large Data Bases*.

[104] T. Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267, 1–38.

[105] B. Mittelstadt, C. Russell, and S. Wachter. 2019. Explaining explanations in AI. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency*, 279–288.

[106] S. Miura, A. L. Cohen, and S. Zilberstein. 2021. Maximizing legibility in stochastic environments. In *International Conference on Robot & Human Interactive Communication (RO-MAN)*, 1053–1059.

[107] S. Miura and S. Zilberstein. 2021. A unifying framework for observer-aware planning and its complexity. In *Uncertainty in Artificial Intelligence*, 610–620.

[108] R. K. Mothilal, A. Sharma, and C. Tan. 2020. Explaining machine learning classifiers through diverse counterfactual explanations. In *ACM Conference on Fairness, Accountability, and Transparency*, 607–617.

[109] J. R. Movellan and J. S. Watson. 2002. The development of gaze following as a Bayesian systems identification problem. In *International Conference on Development and Learning (ICDL)*, 34–40.

[110] B. Muir and N. Moray. 1996. Trust in automation: Part II. Experimental studies of trust and human intervention in automated systems. *Ergonomics*, 37, 1905–1922.

[111] B. M. Muir. 1987. Trust between humans and machines, and the design of decision aids. *International Journal of Man-Machine Studies*, 27, 5-6, 527–539.

[112] B. M. Muir. 1994. Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics*, 37, 11, 1905–1922.

[113] S. Nashed, J. Svegliato, and S. Zilberstein. 2021. Ethically compliant planning within moral communities. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 188–198.

[114] S. B. Nashed, S. Mahmud, C. V. Goldman, and S. Zilberstein. 2023. Causal explanations for sequential decision making under uncertainty. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2307–2309.

[115] C. Nass and Y. Moon. 2000. Machines and mindlessness: Social responses to computers. *Journal of Social Issues*, 56, 1, 81–103.

[116] S. Nisioi, S. Štajner, S. P. Ponzetto, and L. P. Dinu. 2017. Exploring neural text simplification models. In *55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 85–91.

[117] B. Nooteboom. 2002. *Trust: Forms, foundations, functions, failures and figures*. Edward Elgar Publishing.

[118] D. V. Olson and M. Li. 2015. Does a nation's religious composition affect generalized trust? The role of religious heterogeneity and the percent religious. *Journal for the Scientific Study of Religion*, 54, 4, 756–773.

[119] J. Otsuka and H. Saigo. 2022. On the equivalence of causal models: A category-theoretic approach. In *arXiv preprint arXiv:2201.06981*.

[120] C. Panigutti, A. Perotti, and D. Pedreschi. 2020. Doctor XAI: An ontology-based approach to black-box sequential data classification explanations. In *ACM Conference on Fairness, Accountability, and Transparency*, 629–639.

[121] R. Parasuraman and V. Riley. 1997. Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39, 2, 230–253.

[122] R. Parasuraman, T. B. Sheridan, and C. D. Wickens. 2000. A model for types and levels of human interaction with automation. *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans*, 30, 3, 286–297.

[123] K. Pearson. 1901. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2, 11, 559–572.

[124] J. Petch, S. Di, and W. Nelson. 2022. Opening the black box: The promise and limitations of explainable machine learning in cardiology. *Canadian Journal of Cardiology*, 38, 2.

[125] H. Pouget, H. Chockler, Y. Sun, and D. Kroening. 2020. Ranking policy decisions. In *arXiv preprint arXiv:2008.13607*.

[126] J. K. Rempel, J. G. Holmes, and M. P. Zanna. 1985. Trust in close relationships. *Journal of Personality and Social Psychology*, 49, 1, 95.

[127] R. Roscher, B. Bohn, M. F. Duarte, and J. Garcke. 2020. Explainable machine learning for scientific insights and discoveries. *IEEE Access*, 8, 42200–42216.

[128] B. Rozemberczki, L. Watson, P. Bayer, H.-T. Yang, O. Kiss, S. Nilsson, and R. Sarkar. 2022. The Shapley value in machine learning. In *arXiv preprint arXiv:2202.05594*.
[129] J. Russell and E. Santos. 2019. Explaining reward functions in Markov decision processes. In *Thirty-Second International FLAIRS Conference*, 56–61.
[130] W. C. Salmon. 2006. *Four decades of scientific explanation*. University of Pittsburgh Press.
[131] L. Scavuzzo, F. Chen, D. Chételat, M. Gasse, A. Lodi, N. Yorke-Smith, and K. Aardal. 2022. Learning to branch with tree MDPs. *Advances in Neural Information Processing Systems*, 35, 18514–18526.
[132] L. Scharrer, R. Bromme, M. A. Britt, and M. Stadtler. 2012. The seduction of easiness: How science depictions influence laypeople's reliance on their own evaluation of scientific information. *Learning and Instruction*, 22, 3, 231–243.
[133] O. Schilke, M. Reimann, and K. S. Cook. 2021. Trust in social relations. *Annual Review of Sociology*, 47, 239–259.
[134] J. C. Schlimmer. 1987. *Concept Acquisition through Representational Adjustment*. University of California, Irvine.
[135] R. Selvey, A. Grastien, and S. Thiébaux. 2023. Formal explanations of neural network policies for planning. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 5446–5456.
[136] S. Semmler and Z. Rose. 2017. Artificial intelligence: Application today and implications tomorrow. *Duke L. & Tech. Rev.*, 16, 85.
[137] L. S. Shapley. 1953. A value for n-person games. In *Contributions to the Theory of Games*. H. W. Kuhn and A. W. Tucker, (Eds.) Princeton University Press. Chap. 7, 307–317.
[138] T. B. Sheridan. 2019. Individual differences in attributes of trust in automation: Measurement and application to system design. *Frontiers in Psychology*, 10, 1117.
[139] N. V. Smirnov. 1939. On the estimation of the discrepancy between empirical curves of distribution for two independent samples. *Bull. Math. Univ. Moscou*, 2, 2, 3–14.
[140] S. Sreedharan, T. Chakraborti, and S. Kambhampati. 2017. Balancing explicability and explanation in human-aware planning. In *2017 AAAI Fall Symposium Series*, 61–68.
[141] N. Srikanth and J. J. Li. 2020. Elaborative simplification: Content addition and explanation generation in text simplification. In *arXiv preprint arXiv:2010.10035*.
[142] E. Štrumbelj and I. Kononenko. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41, 647–665.
[143] K. Stubbs, P. J. Hinds, and D. Wettergreen. 2007. Autonomy and common ground in human-robot interaction: A field study. *IEEE Intelligent Systems*, 22, 2, 42–50.
[144] R. Sukkerd, R. Simmons, and D. Garlan. 2020. Tradeoff-focused contrastive explanation for MDP planning. In *29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 1041–1048.
[145] M. Sundararajan and A. Najmi. 2020. The many Shapley values for model explanation. In *International Conference on Machine Learning*, 9269–9278.
[146] J. Svegliato, S. B. Nashed, and S. Zilberstein. 2021. Ethically compliant sequential decision making. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 11657–11665.
[147] Y.-M. Tamm, R. Damdinov, and A. Vasilev. 2021. Quality metrics in recommender systems: Do we calculate metrics consistently? In *Proceedings of the 15th ACM Conference on Recommender Systems*, 708–713.
[148] A. V. Taylor, E. Mamantov, and H. Admoni. 2022. Observer-aware legibility for social navigation. In *31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 1115–1122.
[149] L. Trave-Massuyes and R. Pons. 1997. Causal ordering for multiple mode systems. In *11th International Workshop on Qualitative Reasoning*, 203–214.
[150] S. Triantafyllou, A. Singla, and G. Radanovic. 2022. Actual causality and responsibility attribution in decentralized partially observable Markov decision processes. In *arXiv preprint arXiv:2204.00302*.
[151] A. Trotman and V. Kitchen. 2022. Quality metrics for search engine deterministic sort orders. *Information Processing & Management*, 59, 6, 103102.
[152] A. Tversky and I. Simonson. 1993. Context-dependent preferences. *Management Science*, 39, 10, 1179–1189.
[153] F. Wang and T. Yamagishi. 2005. Group-based trust and gender differences in China. *Asian Journal of Social Psychology*, 8, 2, 199–210.
[154] N. Wang, D. V. Pynadath, and S. G. Hill. 2016. The impact of POMDP-generated explanations on trust and performance in human-robot teams. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 997–1005.
[155] W. Wang and I. Benbasat. 2007. Recommendation agents for electronic commerce: Effects of explanation facilities on trusting beliefs. *Journal of Management Information Systems*, 23, 4, 217–246.
[156] D. Watson. 2022. Rational Shapley values. In *ACM Conference on Fairness, Accountability, and Transparency*, 1083–1094.
[157] J. J. Williams, T. Lombrozo, and B. Rehder. 2013. The hazards of explanation: Overgeneralization in the face of exceptions. *Journal of Experimental Psychology: General*, 142, 4, 1006.
[158] J. Woodward. 2005. *Making things happen: A theory of causal explanation*. Oxford university press.

[159] K. H. Wray, S. J. Witwicki, and S. Zilberstein. 2017. Online decision-making for scalable autonomous systems. In *International Joint Conference on Artificial Intelligence, (IJCAI)*, 4768–4774.
[160] J. Xu, K. Le, A. Deitermann, and E. Montague. 2014. How different types of users develop trust in technology: A qualitative analysis of the antecedents of active and passive user trust in a shared technology. *Applied Ergonomics*, 45, 6, 1495–1503.
[161] R. Zeffane. 2018. Do age, work experience and gender affect individuals' propensity to trust others? An exploratory study in the United Arab Emirates. *International Journal of Sociology and Social Policy*, 38, 3/4, 210–223.
[162] Y. Zhang, Q. V. Liao, and R. K. Bellamy. 2020. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *ACM Conference on Fairness, Accountability, and Transparency*, 295–305.
[163] X. Zhong, B. Gallagher, S. Liu, B. Kailkhura, A. Hiszpanski, and T. Y.-J. Han. 2022. Explainable machine learning in materials science. *NPG Computational Materials*, 8, 1, 204.
[164] L. Zhu and T. Williams. 2020. Effects of proactive explanations by robots on human-robot trust. In *12th International Conference on Social Robotics (ICSR)*, 85–95.
[165] S. Zuboff. 1988. *In the age of the smart machine: The future of work and power*. Basic Books, Inc.

A Additional User Study Details

A.1 Common Demographic and Lifestyle Questions

To begin both surveys, we asked three basic demographic questions, shown on the left in Figure 15. At the conclusion of both surveys, we asked the participants about their general use of AI technologies, as well as their overall level of trust in such systems in general, shown on the right in Figure 15.

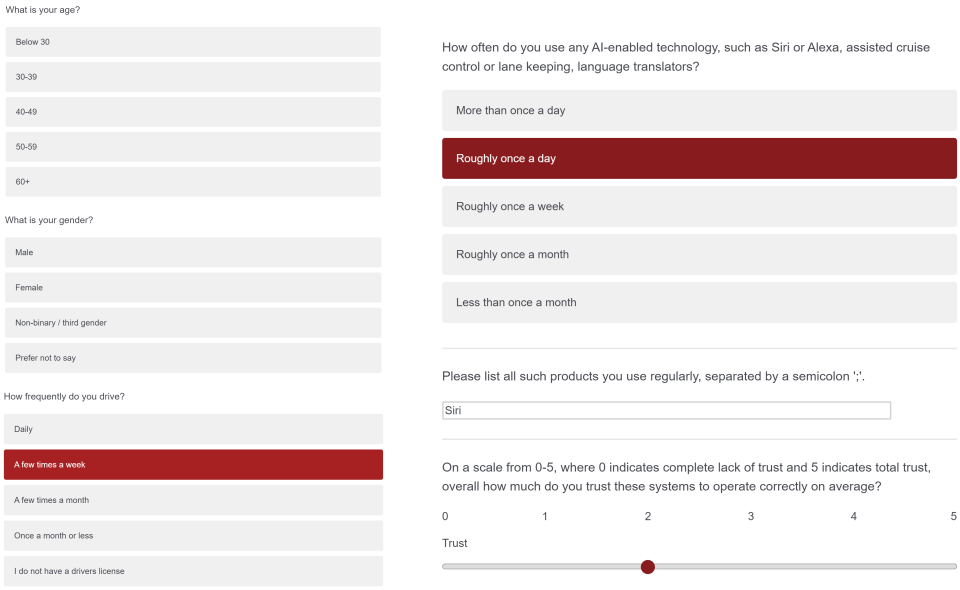


Fig. 15. Demographic, technology use, and attitudes questions common to both surveys.

A.2 Study Demographics

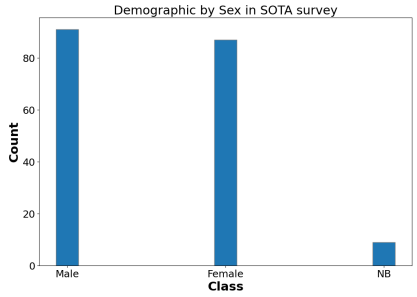
In total, 189 (199) participants from the United States and Canada were recruited via the crowd-sourcing platform Prolific (www.prolific.co) for the SOTA (CONTEXT) study. All participants were fluent in English. Figure 16 summarizes participant demographics by gender (16a), age (16b), driving frequency (16c), and rates of AI technology use (16d) for both studies.

A.3 SOTA Details

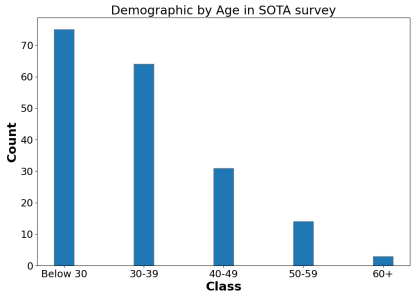
For the SOTA study, after answering the basic demographic questions, every participant was shown the instructions at the top of Figure 17. The participants were then shown 3 clips of simulated driving behavior (not shown here) in a random order. After each clip, participants were asked to rank the explanations using the prompt shown at the bottom of Figure 17.

A.4 CONTEXT Details

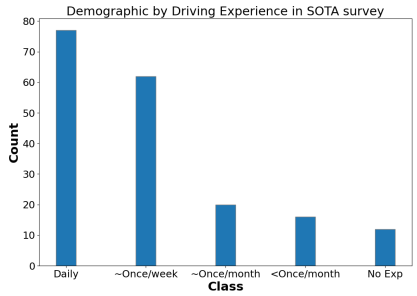
For the CONTEXT study, after answering basic demographic questions, each participant was randomly shown one of the two prompts in Figure 18. If they saw the top prompt, participants were placed in the 'passive' group. If they saw the bottom prompt, they were placed in the 'active' group. After reading the prompt, the participants were shown 8 different clips of simulated driving behavior in a random order and presented with an explanation



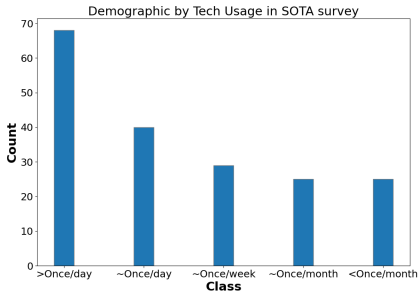
(a) Gender



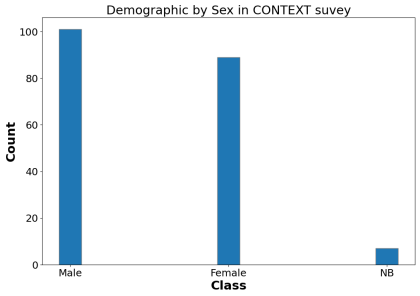
(b) Age



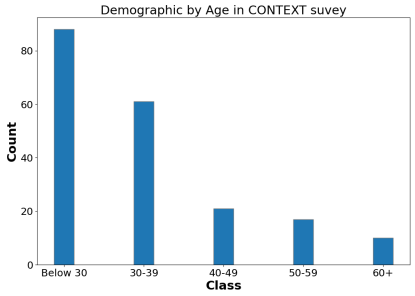
(c) Driving frequency



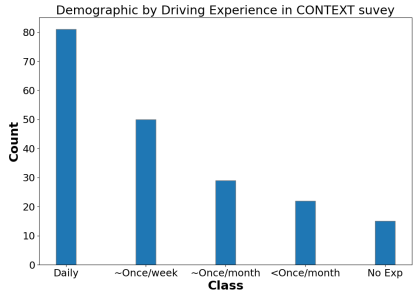
(d) Technology use



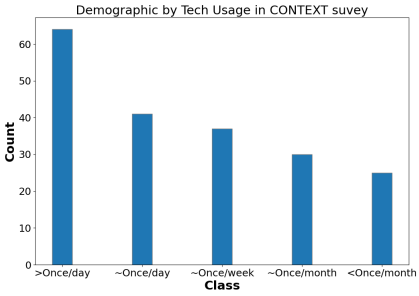
(e) Gender



(f) Age



(g) Driving frequency



(h) Technology use

Fig. 16. Demographic summary for the SOTA and CONTEXT surveys.

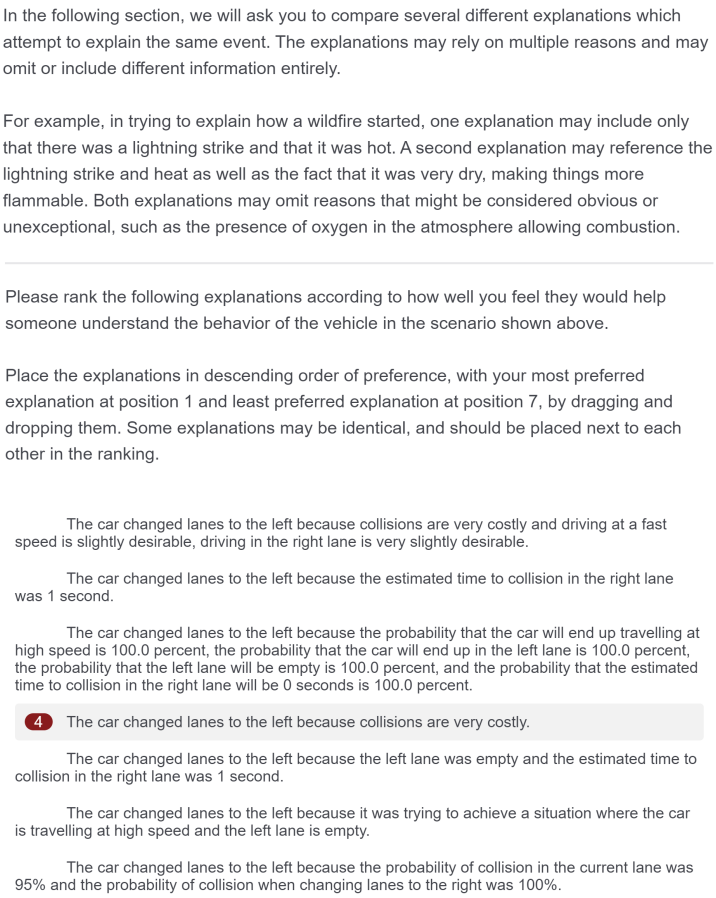


Fig. 17. Instructions (top) and prompt (bottom) for the SOTA survey.

of the behavior shown in the clip which was generated using our framework, all of which are shown in Figure 19. After each clip, the participants were asked the following 3 questions, shown in Figure 20.

In an effort to keep the survey short and thus participant attentiveness and data quality high, we showed each participant a random subset (8) of all (12) possible scenario-explanation pairs. The survey software we used, Qualtrics (www.qualtrics.com), allowed us to balance the questions shown so that all 12 scenario-explanation pairs occurred equally frequently in the data set as a whole.

Last, for all participants in the active group, we asked an additional sequence of questions. For each scenario, we asked whether they would initiate a transfer of control upon seeing different explanations, shown in Figure 22.

A.5 Additional Results

Figure 22 summarizes the results of the question asked in Figure 21. Here we can see that, regardless of the scenario, F-type explanations cause the lowest percentage of transfer of control requests, and R-type explanations seem to produce the highest. T-type and V-type explanations rates seem to vary depending on the specific scenario.

As an example, we also include the partitions of data that lead to the minimum and maximum correlation values. Figure 23a, while still showing a relatively correlated relationship, depicts very uncontroversial V-type

In the following scenarios, you will see a top-down view of cars driving on a highway. The traffic moves from left to right, and you are inside the green car. The green car is aware of the position and speed of other nearby vehicles shown in blue.

Imagine you are a passenger inside the car. It is fully autonomous, without a steering wheel or any other way for you to influence the behavior of the car.

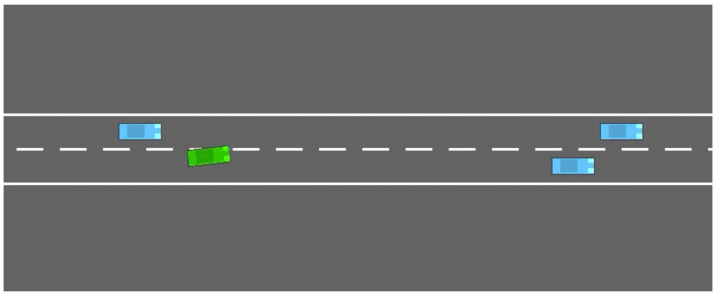
In the following scenarios, you will see a top-down view of cars driving on a highway. The traffic moves from left to right, and you are inside the green car. The green car is aware of the position and speed of other nearby vehicles shown in blue.

Imagine you are sitting in the driver's seat. The car is fully autonomous, but has a steering wheel. You may indicate you wish to intervene and switch back to manual driving by hitting the (i) key. You will not be able to control the car in the clip shown, but your desire to take over control will be registered.

Fig. 18. Passive (top) and active (bottom) prompts in the CONTEXT survey.

This survey will show you several videos, each followed by one or more questions. Please watch the videos before answering the questions. You may watch the videos multiple times if necessary in order to them remember clearly.

Please watch the scenario below.



The car provides you, the passenger, the following explanation for its behavior:

The car changed lanes to the left because it was trying to achieve a situation where the car is driving in the left lane, the car is travelling at high speed, and the left lane is empty.

Assuming only this explanation is provided during the maneuver, please answer the following questions.

Fig. 19. Example clip, explanation, and prompt for the CONTEXT survey.

explanations, with almost all participants responding neutrally or mildly positive. Figure 23b on the other hand shows a highly correlated and highly polarizing reaction to T-type explanations, with roughly the same fraction of participants reporting a strong agreement/disagreement with respect to the explanation's effect on trust and understanding. Moreover, almost no participants responded neutrally to these explanations.

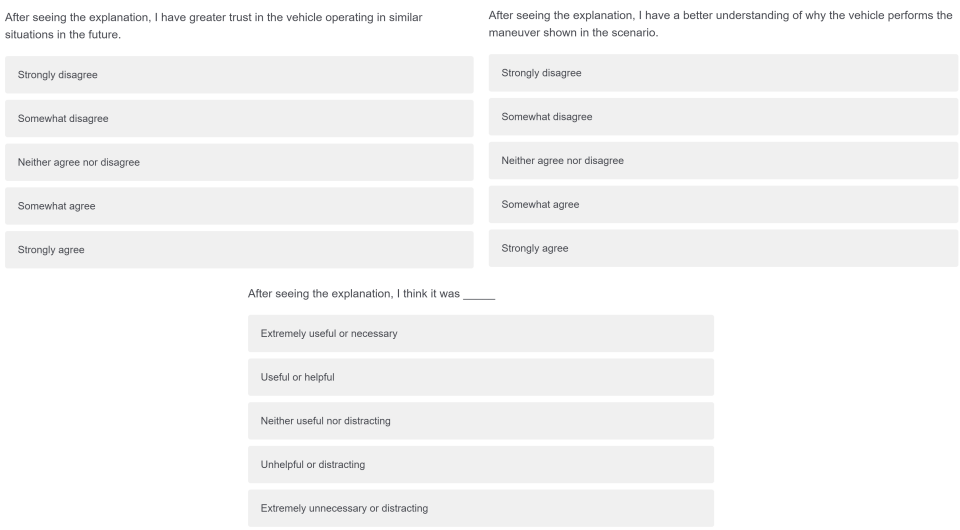


Fig. 20. Questions regarding trust, necessity, and understanding.

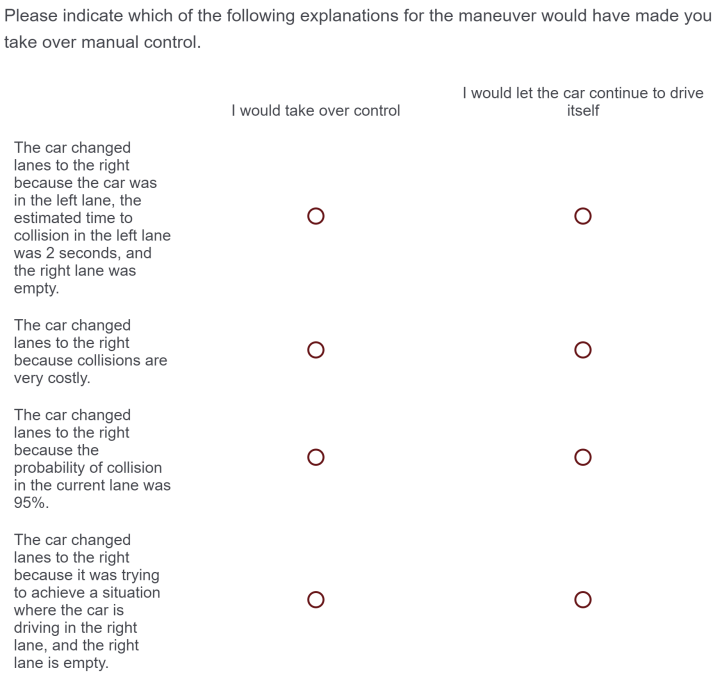
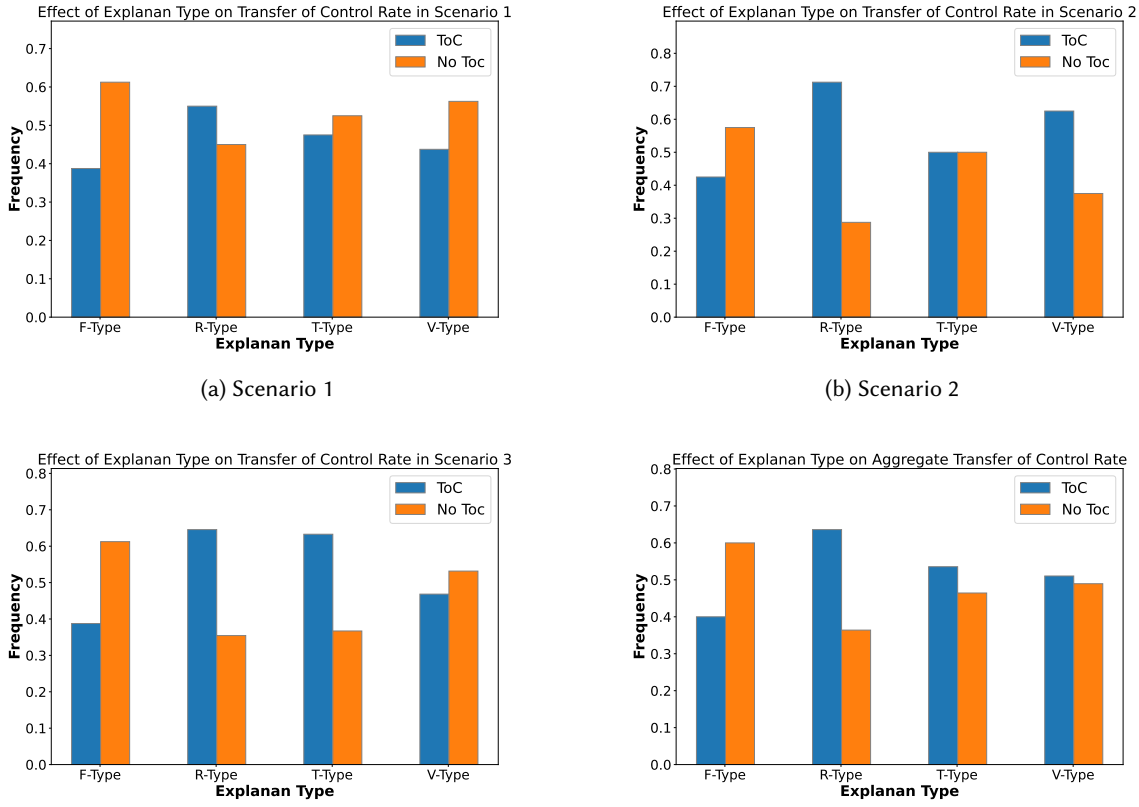


Fig. 21. Transfer of control questions for active drivers only.

B Additional Experiments and Experimental Details

In this section we provide more extensive details regarding the setup of the empirical experiments. We also provide some additional results on convergence rates and similarity measure behavior, and present an additional example.



(a) Scenario 1

(b) Scenario 2

(c) Scenario 3

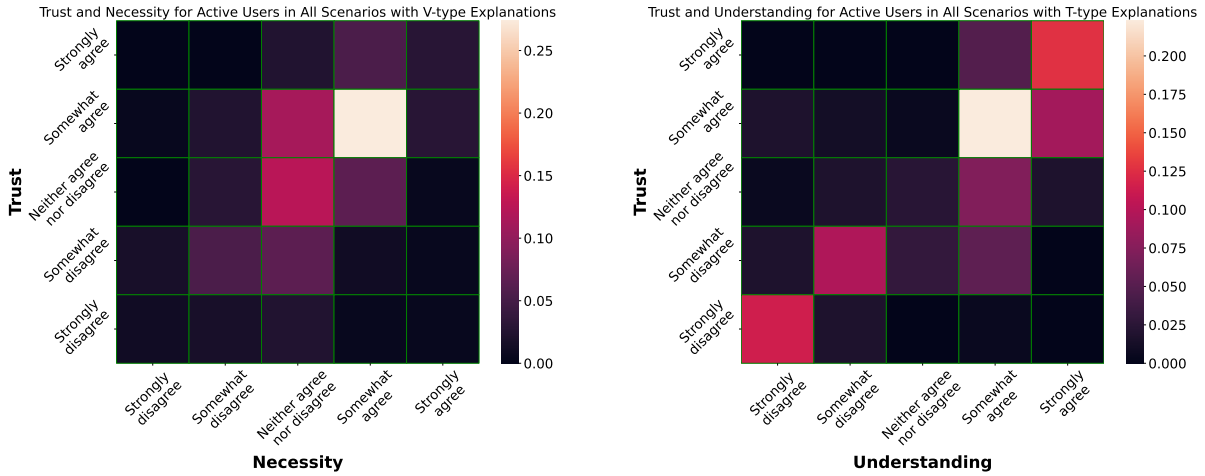
(d) Aggregate

Fig. 22. Transfer of control rates.

B.1 Timing Benchmarks

For timing experiments, we use the Lunar Lander domain, which has up to 8 state factors (features) and 4 actions. To vary the feature size we simply consider a subset of the features as possibly causal. Features that are not chosen stay static throughout the sampling process. Since the original domain is continuous with a minimum and maximum bound on each feature, we discretize the domains linearly prior to sampling, and vary the discretization fidelity from 2 to 20 in increments of 1. The policy being explained is a generated via deep Q-learning using a multi-layer perceptron and stable baseline 3.

Sampling was performed, with replacement, using the occupancy statistics of the policy in order to get a representative picture of explanations likely to be encountered or requested during operation. In this experiment, we only generate singleton ($|X| = 1$) explanations, however we should note that this offers the most favorable scaling for exact MeanRESP compared to Monte Carlo MeanRESP. Because of the non-monotonic behavior of error measures with respect to sample number in Monte Carlo MeanRESP, we run 10,000 samples for every problem, and then find the last point (highest sample number) at which the sum of the absolute difference in responsibility score between all the singletons of the sampled and exact explanation goes below 0.01. This corresponds to the



(a) $r = 0.5034$ (b) $r = 0.7729$

Fig. 23. Minimum and maximum Pearson r (correlation) values.

final, stable convergence point, and using this time as the completion time for Monte Carlo MeanRESP ensures that it never gets 'lucky' by estimating very accurate responsibility score before it has truly converged.

All of our experiments were conducted on a Dell XPS 13 9310 Laptop with an 11th Gen Intel(R) Core(TM) i7-1185G7 3.00GHz processor and 16GB 4267MHz LPDDR4x RAM.

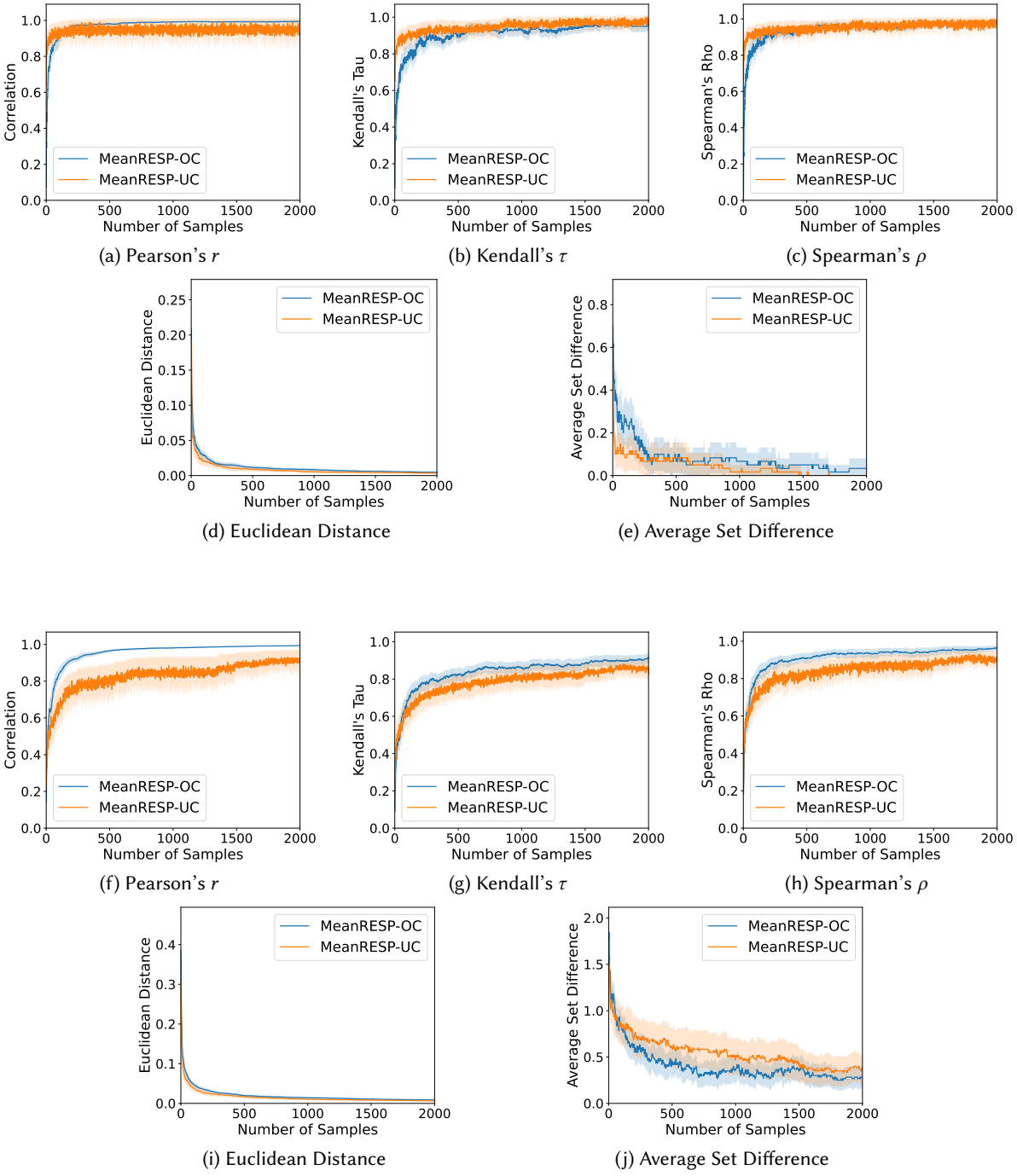
B.2 Experimental Details for Shapley Comparison

Experiments comparing Monte Carlo MeanRESP to Shapley value methods are run on 4 domains: Lunar Lander, Highway, BlackJack, and Taxi. For the Highway and Lunar Lander policies we use deep Q-learning using a multi-layer perceptron and stable baseline 3. Policies for Taxi and Blackjack were computed with value iteration. As in the timing experiments, 60 states were sampled with replacement from each domain proportional to the states's occupancy frequency under the given policies. Empirically, we found that after roughly 2000 samples, there were very few significant changes. Thus, we capture results from Monte Carlo MeanRESP after 1000 samples. The Shapley-value-based method is also sample based, and we used 10,000 samples.

B.3 Error Rates versus Samples

We now show some additional results comparing the behavior of Monte Carlo MeanRESP under alternate definitions of cause (Figures 24 and 25). In these experiments, we use the data from the Shapley comparison experiments and evaluate some of the similarity measures. Mean values over all 60 explanations are shown in bold, while 95% confidence intervals are represented by the shaded regions.

Here, we can see that convergence rates seem more dependent on problem than they do similarity measure, which reinforces the idea that the original and updated definitions of cause may highlight very different explanans and thus may be sensitive to different causal structures within problems. The exceptions to this seem to be Euclidean distance, for which no real difference is observable, and correlation (Pearson's r), which seems to converge slower or relatively equally in the updated version across all the domains.



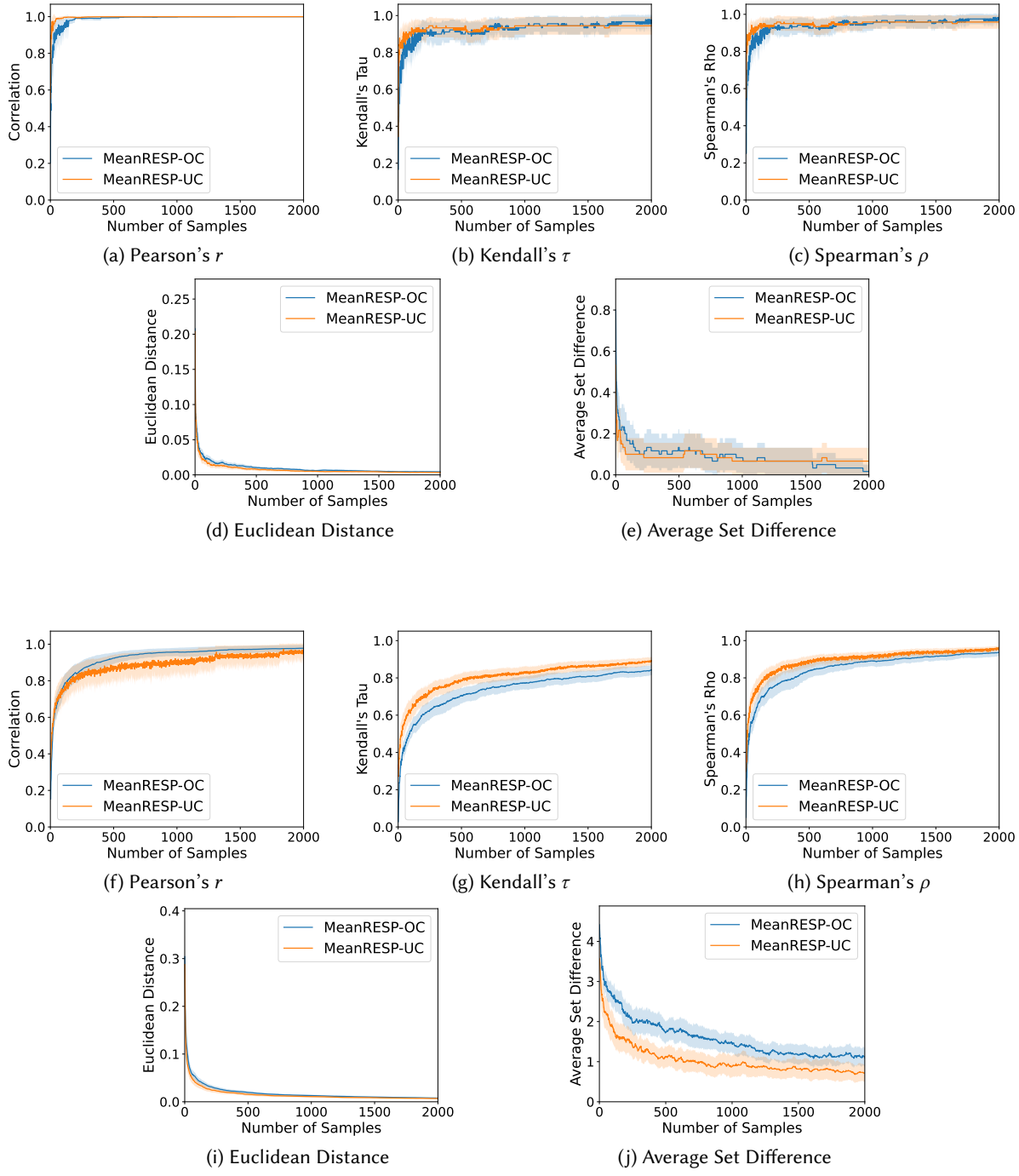
(a) Pearson's r (b) Kendall's τ (c) Spearman's ρ

(d) Euclidean Distance (e) Average Set Difference

(f) Pearson's r (g) Kendall's τ (h) Spearman's ρ

(i) Euclidean Distance (j) Average Set Difference

Fig. 24. Error curves vs. sample numbers for the Taxi domain (top, (a)-(e)), and Lunar Lander domain (bottom, (f)-(j)).



(a) Pearson's r (b) Kendall's τ (c) Spearman's ρ

(d) Euclidean Distance (e) Average Set Difference

(f) Pearson's r (g) Kendall's τ (h) Spearman's ρ

(i) Euclidean Distance (j) Average Set Difference

Fig. 25. Error curves vs. sample numbers for the Blackjack domain (top, (a)-(e)), and Highway domain (bottom, (f)-(j)).

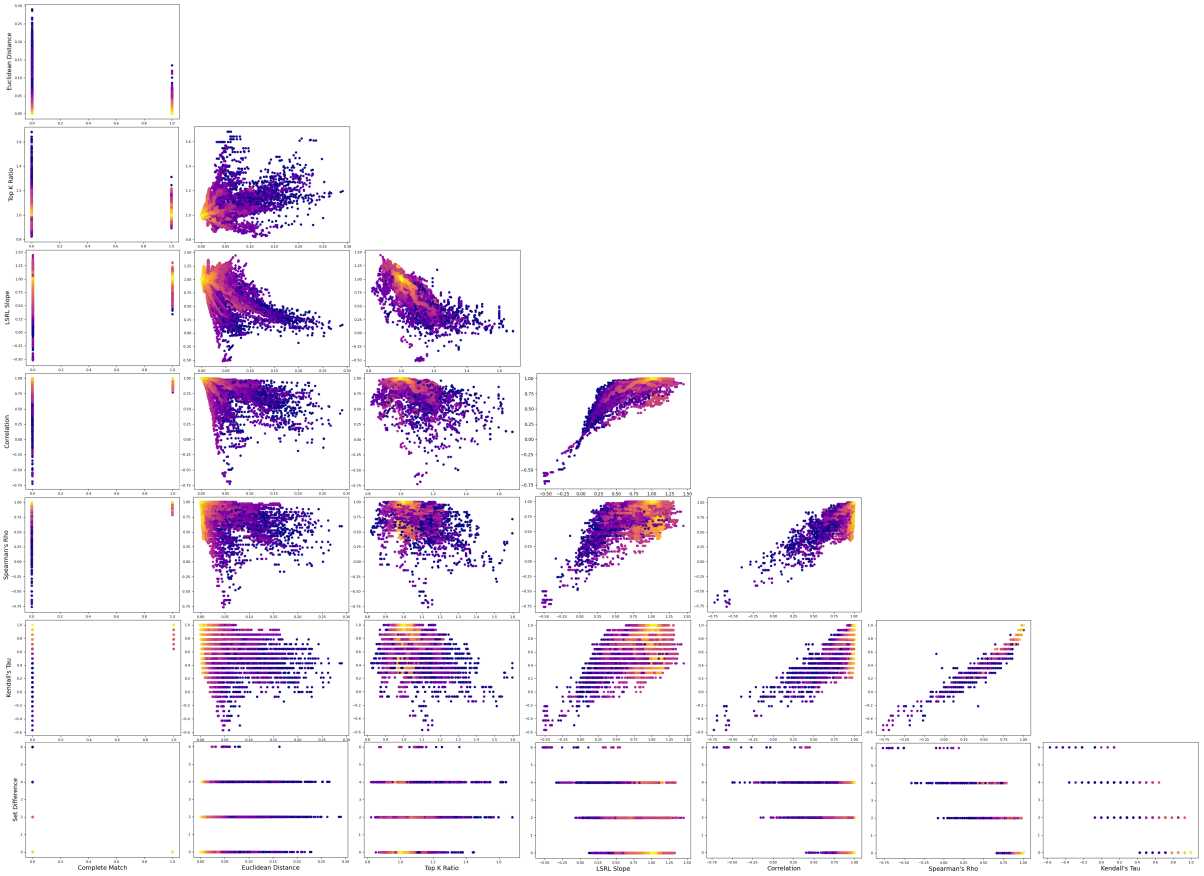
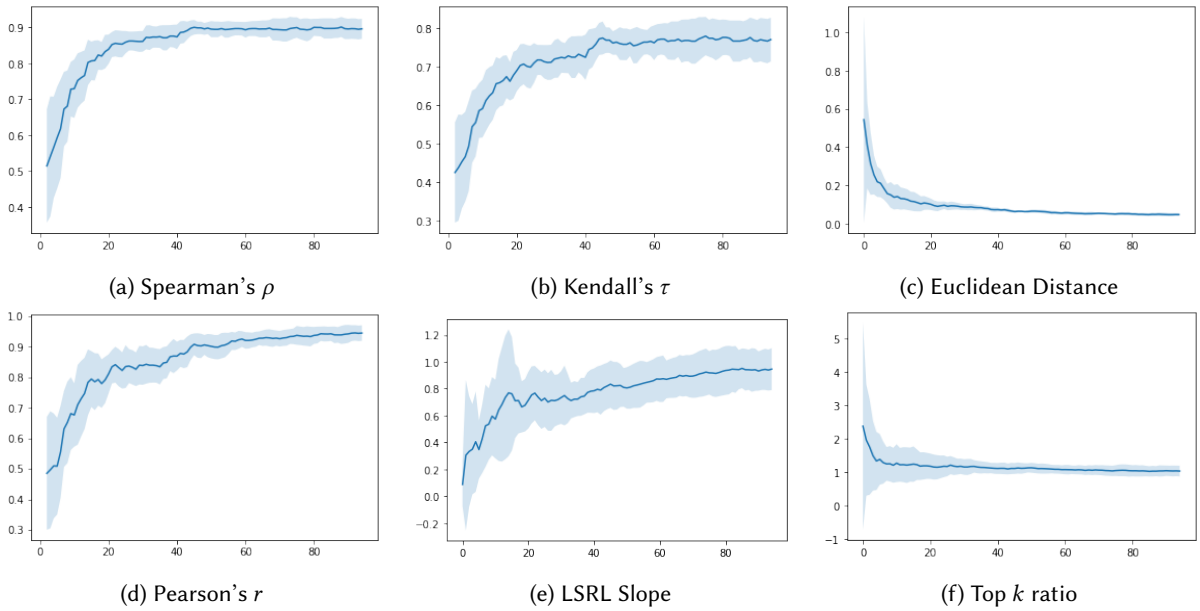


Fig. 26. Measure comparisons for the Lunar Lander domain as a function of sample number. All sub-figures share horizontal and vertical axis labels with the corresponding label at the margin of the figure.

Table 7. The top $k = 5$ causes determined by exact (ground truth) MeanRESP and Monte Carlo MeanRESP after $N = 10, 50, 95$ samples. While responsibility score estimates fluctuate with additional samples, some highly influential variables are easily identified (ranks 1 and 2). Moreover, other weakly influential variables appear frequently (anti-satellite test ban), although not always at the correct rank. This table appears to show that the most influential variables stabilize first, which makes sense given that a higher responsibility score indicates a larger fraction of samples will identify the variable as a cause.

Ground Truth	N = 10	N = 50	N = 95
1. Adoption of the budget resolution	1. Adoption of the budget resolution	1. Adoption of the budget resolution	1. Adoption of the budget resolution
2. Duty-free exports	2. Duty-free exports	2. Duty-free exports	2. Duty-free exports
3. Education spending	3. Anti-satellite test ban	3. Education spending	3. Education spending
4. Anti-satellite test ban	4. Aid to Nicaraguan Contras	4. Superfund Right to Sue	4. Anti-satellite test ban
5. Export administration act South Africa	5. Superfund Right to Sue	5. Anti-satellite test ban	5. Export administration act South Africa



(a) Spearman's ρ (b) Kendall's τ (c) Euclidean Distance

(d) Pearson's r (e) LSRL Slope (f) Top k ratio

Fig. 27. Traces of six measures over time as they compare exact and approximate responsibility estimates. The solid blue line represents the mean (10 runs total), and the blue shaded regions represent one standard deviation. All values were generated for one particular input. Clearly, some measures are more sensitive than others. Moreover, some shifts appear to be detected universally, for example, near 45 samples, while at other points some measures respond to updated estimates while others do not.

B.4 Experimental Details for Metrics for Approximate Explanations

For the metrics experiments, we again use the Lunar Lander domain and a policy trained using deep Q-learning, and we generate explanations for each of the 60 states up to 5000 samples. Here, we let $k = 4$.

In this appendix, we present some additional results of our framework generating explanations of a random forest classifier we trained on the Congressional Voting Records Data Set [134]. The number of instances in the data set is 435. The number of input features is 16 and the number of labels is 2.

B.5 Additional Results for Metrics for Approximate Explanations

Figure 26 shows the complete set of bilateral metric comparisons. Table 7 shows several snapshots of the top k most responsible variables for a given classification outcome in the Congressional Voting Records Data Set for both exact MeanRESP and at several points during the sampling process of Monte Carlo MeanRESP. Figure 27 summarizes the behavior of these measures throughout the sampling process.

Received 15 January 2025; revised 10 May 2025; accepted 15 June 2025