Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)
Survey Track

# The Emerging Landscape of Explainable Automated Planning & Decision Making

**Tathagata Chakraborti**[1*], **Sarath Sreedharan**[2*] and **Subbarao Kambhampati**[2]
[1]IBM Research AI
[2]Arizona State University
tchakra2@ibm.com, {ssreedh3, rao}@asu.edu

## Abstract

In this paper, we provide a comprehensive outline of the different threads of work in Explainable AI Planning (XAIP) that has emerged as a focus area in the last couple of years, and contrast that with earlier efforts in the field in terms of techniques, target users, and delivery mechanisms. We hope that the survey will provide guidance to new researchers in automated planning towards the role of explanations in the effective design of human-in-the-loop systems, as well as provide the established researcher with some perspective on the evolution of the exciting world of explainable planning.

## 1 Introduction

As AI techniques mature, issues of interfacing with users has emerged as one of the important challenges facing the AI community. Primary among these challenges is for AI-based systems to be able to explain their reasoning to humans in the loop [31]. This is necessary both for collaborative interactions where humans and AI systems solve problems together, as well as in establishing trust with end users in general. Among the work in this direction in the broader AI community, in this survey, we focus on how the automated planning community in particular has responded to this challenge.

One of the recent developments towards this end is the establishment of the Explainable AI Planning (XAIP) Workshop[1] at the International Conference on Automated Planning and Scheduling (ICAPS), the premier conference in the field. The agenda of the workshop states:

> *While XAI at large is primarily concerned with black-box learning-based approaches, model-based approaches are well suited – arguably better suited – for an explanation, and Explainable AI Planning (XAIP) can play an important role in helping users interface with AI technologies in complex decision-making procedures.*

In general, this is true for sequential decision making tasks for a variety of reasons. The complexity of automated planning and decision making, and consequently the role of explainability in it, raises many more challenges than function approximation tasks (e.g. classification) as was originally focused on [30] by the XAI Program from DARPA. This includes dealing with complex constraints over problems intractable to the human's inferential capabilities, differences in human expectations and mental models, to proving provenance of various artifacts of a system's decision making process over long term interactions even as the world evolves around it. Furthermore, these typically deal with reasoning tasks where we tend to seek explanations anyway in human-human interactions, as opposed to perception tasks where we largely do not ask for explanations.

Thus, the original DARPA XAI Project [30], which served as a great catalyst towards advancing research in explainable AI, has also seen evolution [31] from a core focus in classification tasks to the broader sense of human-AI collaboration. Recent surveys on the topic [3] also recognize this lacuna in explainability of artificial intelligence in decision-making tasks. As the issue of explainability becomes front and center in AI, the importance of long term decision making cannot be avoided [82]. This is highlighted by the emergence of XAI-subcommunities within planning, multi-agents, and other communities at premier AI conferences, including the Explainable AI (XAI) Workshop[2] at the International Joint Conference on Artificial Intelligence (IJCAI) and the Explainable Transparent Autonomous Agents and Multi-Agent Systems (EXTRAAMAS) Workshop[3] at the International Conference on Autonomous Agents and Multiagent Systems (AAMAS), which in addition to the XAIP Workshop mentioned above, has captured the imagination of this emerging field of inquiry.

**Survey Scope and Outline.** In this survey, we highlight the role of explanations in the many unique dimensions of a decision making problem, and provide a comprehensive survey of recent work in this direction. In particular, we will focus on automated planning as a subfield of decision making problems but we will point to work in the broader area wherever necessary to highlight themes of explainable planning in general. To this end, we will start with a brief overview of the different kinds of users associated with an automated decision making task and the considerations for an explanation in each case. We then introduce various aspects of a planning

*Equal contribution.
[1]https://kcl-planning.github.io/XAIP-Workshops/
[2]https://sites.google.com/view/xai2019
[3]https://extraamas.ehealth.hevs.ch

task formally and delve into a survey of existing works that tackle the explanation problem in one or more of these dimensions, while comparing and contrasting the properties of such explanations. Finally, we will conclude with a summary of emerging trends in XAIP research.

**Out of scope.** In the survey, we focus exclusively on explanations of a plan as a solution of a given planning problem. We will not cover meta planning problems such as *goal reasoning* [66; 17; 60], or open world considerations in the explanation of plans that fail [32]. We will also not cover novel behaviors in pursuit of explainability: e.g. the generation of *explicable plans* [87] that conform to user expectations and are thus not required to be explained, or the design of environments to facilitate the same [46]. For a detailed treatise of the same, we refer the reader to [11]. Other topics excluded are execution time considerations, such as in [49].

## 2 The Many Faces of XAIP

The primary considerations in the design of explainable systems is the consideration of the persona of the explainee. This is true for explaianble AI in general [89] but also acknowledged to be crucial to the XAIP scene as well [48].

- *End user:* This is the person who interacts with the system in the form of a user. For a planning system, this may be the human teammate in a human-robot team [13] who is impacted by, or is a direct stakeholder in the plans of the robot, or user collaborating with an automated planner in a decision support setting [29].
- *Domain Designer:* This is the person involved in the acquisition of the model that the system works with: e.g. the designer of goal-oriented conversation systems [69].
- *Algorithm Designer:* The final persona is that of the developer of the algorithms themselves: e.g. in the context of automated planning systems, this could be someone working on informed search. The domain designer is distinct from the algorithm designer and may even not have any overlap in expertise (e.g. [69]).

## 3 The Decision Making Problem

A sequential decision making or planning problem $\Pi$ is defined in terms of a transition function $\delta_\Pi : A \times S \to S \times \mathbb{R}$, where $A$ is the set of capabilities available to the agent, $S$ is the set of states it can be in, and the real number denotes the cost of making the transition. $\delta$ thus describes how an agent behaves in the world in terms of its capabilities – what those capabilities depend on and how they change the state of the world. The planning algorithm $\mathbb{A}$ solves $\Pi$ subject to a desired property $\tau$ to produce a plan or policy $\pi$, i.e. $\mathbb{A} : \Pi \times \tau \mapsto \pi$. Here, $\tau$ may represent different properties such as soundness, optimality, and so on. We refer to the optimal or best plan, that optimizes $\tau$, as $\pi^*$.

- **Plan** $\pi = \langle a_i, a_2, \ldots, a_n \rangle, a_i \in A$ that transforms the current state $I \in S$ of the agent to its goal $G \in S$, i.e. $\delta_\Pi(\pi, I) = \langle G, \sum_{a_i \in \pi} c_i \rangle$. The second term in the output denotes the plan cost $c(\pi)$. A plan is thus a sequence of actions or a course of action that an agent can take to achieve its goal, given a description of the world.

| | Algorithm-based Explanations | Model-based Explanations | |
|---|---|---|---|
| | | Inference Resolution | Model Reconciliation |
| End User | ✗ | ✓ | ✓ |
| Domain Designer | ✗ | ✓ | n/a |
| Algorithm Designer | ✓ | ✗ | ✗ |

- **Policy** $\pi : s \mapsto a, a \in A, \forall\, s \in S$ provides a mapping from any state $s$ of the agent to the desired action $a$ to be taken in that state. A policy tells the agent how to behave in any given state of the world, as opposed to a plan which prescribes a sequence of actions given the current state to reach a specified goal state.

While specific decision making tasks have more nuanced definitions characterizing what forms states and actions can take, how the transition function is defined, etc. for the purposes of this survey, this abstraction should be enough for the general audience to grasp the salient features of a decision making task and relevant XAIP concepts.

### 3.1 The Explanation Process

The explanation process of a planning problem proceeds as follows, with a question from the explainee about the current solution of a given planning problem, and the explainer (the XAIP system) coming up with an explanation for it:

Q. *"Why $\pi$?"* or *"Why not $\pi'$?"*

Here, $\pi'$ is a *foil* [55] and may be either stated explicitly, implicitly, or even partially (leading to a set of foils) in the questions.[4] Examples of foils would be:

- *"Why $a \notin \pi$?"* is a partial foil where all plans with action $a$ in them are the foils.
- The original question *"Why $\pi$?"* where the implicit foil is *"as opposed to all other plans $\pi'$"*.

A. An explanation $\mathcal{E}$ ensures that the explainee can compute $\mathbb{A} : \Pi \times \tau \overset{\mathcal{E}}{\mapsto} \pi$ and verify that either:

$\mathbb{A} : \Pi \times \tau \overset{\mathcal{E}}{\not\mapsto} \pi'$; or

$\mathbb{A} : \Pi \times \tau \overset{\mathcal{E}}{\mapsto} \pi'$ but $\pi \equiv \pi'$ or $\pi > \pi'$ (the criterion for comparison may be cost, preferences, etc.).

The point of an explanation is thus to establish the property $\tau$ of $\pi$ given a problem $\Pi$ and a algorithm $\mathbb{A}$. The Q&A continues until the explainee is satisfied.

### 3.2 Explanation Artifacts: Algorithm/Model/Plan

From the definition of the decision making task, there are many components at play here which can contribute to an explanation. The system can explain the steps made in $\mathbb{A}$ while solving a problem to the debugger / algorithm designer. It can explain artifacts of the problem description $\Pi$ that led to the decision: these are model-based algorithm-agnostic explanations and are more useful to end users. It can also communicate characteristics of $\pi$ as an explanation.

[4] These questions are shorthand for more specific questions [20; 24] in explanatory dialogue, such as: *"What goal is action $a$ supporting in the plan $\pi$?"*. These generally lead to partial plans and foils that can be compiled into the more generic questions above.

It is interesting to note that this sort of a distinction can be seen in the literature on explainable machine learning as well. For example, LIME [59] interfaces with the explainee at the level of outputs only, i.e. the classification choices made (corresponding to plans computed in our setting) – it is also algorithm dependent since it reveals (albeit simplified) details of the learned model to the user. Approaches like [62], on the other hand, are purely algorithm dependent requiring the explainee to visualize the internal representations learned by the algorithm at hand. Other works such as [18] provide algorithm independent explanations in terms of the input data and black box learners, similar to model-based explanations in our case that use the input problem definition as the basis of an explanation and not the inference engine.

### 3.3 Properties of Explanations

**Social, Selective, and Contrastive.** Looking at how humans explain their decisions to each other can provide great insight on the desired properties of an explanation. Miller in [55] provides an insightful survey of lessons learned from social sciences and how they can impact the informed design of explainable AI systems. He outlines three key properties for consideration: *social* in being able to model the expectations of the explainee, *selective* in being able to select explanations among several competing hypothesis, and *contrastive* in being able to differentiate properties of two competing hypothesis. The contrastive property in particular has received a lot of attention [34; 54] in the XAIP community.

**Local versus Global Explanations.** Another consideration is whether an explanation is geared towards a particular decision (local), e.g. LIME [59], or they are for the entire model (global), e.g. TCAV [42] – for a planning problem this distinction can manifest in many ways: whether the explanation is for a given plan versus if it is for the model in general.

**Abstractions.** One final approach we want to highlight is the use of abstractions: this is especially useful if the model of decision-making is too complex for the explainee and a simplified model can provide more useful feedback [59].

## 4 Algorithm-based Explanations

We first look at attempts to explain the underlying planning algorithm. This is quite useful for debugging: e.g. [51] provides an interactive visualization of the search tree for a given problem. Another case is where the explanation methods are particularly tailored for specific algorithms. Such explanatory methods have become quite common in explaining decisions generated by deep reinforcement learning: [27] uses perturbation based saliency maps for explaining a policy learned by asynchronous advantage actor-critic algorithms, [36] uses more selective saliency maps for dueling deep Q-network, and [44] learns finite-state representations (Moore machine) that can represent RL policies learned by RNNs.

A closely related thread of research involves the development of algorithms with built-in capability for the generating explanations. Examples include RL techniques that work with decomposed reward functions to allow for explanations in those terms ([37; 2]) and the use of multi-objective methods to support qualitative attributes [75; 76].

## 5 Model-based Explanations

Majority of works in XAIP look at algorithm-agnostic methods for generating explanations since properties of a solution can be evaluated independently of the method used to come up with it, given the model of the decision making task. As opposed to debugging settings where the algorithm has to be investigated in more detail, end users typically care about model-based algorithm-agnostic explanations more so that services [7] can be built around it.[5] Approaches in this category deal with two considerations: 1) the inferential capability; and/or 2) the mental model of the user. When both of these are aligned, there is no need to explain.

### 5.1 Inference Reconciliation

Users have considerably less computational ability (let's say $\mathbb{A}^H$) than a planner. In this situation:

$$\mathbb{A} : \Pi \times \tau \mapsto \pi \text{ and } \mathbb{A}^H : \Pi \times \tau \not\mapsto \pi$$

An explanation here is supposed to reconcile the inferential power of the user and the planner:

$$\mathbb{A}^H : \Pi \times \tau \xrightarrow{\mathcal{E}} \pi$$

In order to help the inference process of the user, there are usually two broad approaches (not necessarily exclusive): (a) Allow the user to raise specific questions about a plan and engage in explanatory dialogue; and (b) leverage abstraction techniques to allow the user to better understand the plan.

**Investigatory Dialogue.** With a few exceptions, most of the methods that engage in explanatory dialogue look at queries contrasting the given plan with a foil (implicit or explicit). Authors in [88] discuss how such approaches can have interesting applications in cyber-physical systems (CPSs), while in [69] authors showed how such questions can help in model acquisition tasks as well.

$Q_1$: *"Why is this action in this plan or why $a \in \pi$?"*

Among recent approaches for answering this, [4; 63] use a causal link chain originating at $a$ that can be traced to the goal. There has been a long history of using such information as ways to characterize plans in the context of plan modification and reuse. For example, [80] employs regression from goal to identify initial state conditions relevant to the goal. Similarly [39] looks at using PRIAR style [40] validation annotations to extract "g-features" that includes task effect annotations that support the goal. The explanations here thus take the form of a subset of the model $\mathcal{E} \subseteq \Pi$ that effectively explains the role of the action by pointing out the preconditions of successive actions that are being supported by the plan in question. While these works do not specifically talk about any selection criterion for the explanation content, recent work [8] has shown how such information can be minimized.

This type of explanatory dialogue has also been investigated in the context of agents designed around the belief-desire-intention model, as described in [1; 83; 84]. These

[5] Such domain-independent services and off-the-shelf software – like VAL [35] that has often been used as a subroutine in XAIP techniques – can prove to be very useful in driving adoption for planning and decision-making algorithms in general.

works explain why a particular action was chosen in the context of a specific agent execution trace. Explanations in these works generally consist of the preconditions of the queried action that were met, along with information about previous actions that are relevant to the execution. These explanations also include a contrastive aspect since for choice nodes in the agent program, the explanation includes why the current executed branch was preferred over the alternatives.

$Q_2$: *"Why not this other plan $\pi'$?"*

This is the case where a contrastive foil is explicitly considered. Authors in [7; 45] assume that the foils specified by the user can be best understood as constraints on the plans they are expecting: e.g. a certain action or sequence of actions to be included/excluded. The explanation is then to identify an exemplary plan or policy that satisfies those constraints thus demonstrating how the computed plan is better. Authors in [20], on the other hand, expect the user queries to be expressed in terms of plan properties which are user-defined binary properties that apply to all valid plans for the problem. The explanation then takes the form of other plan properties that are entailed by those properties. This is computed using oversubscription planning with plan properties as goals.

Such methods have also been investigated in the context of RL to train alternate policies based on user questions and contrast them against the original one by comparing the expected outcomes of each [81]. Similarly, authors in [57] generate contrastive explanations by using generative models to create counterfactual states that shows the closest state where a foil action would have been chosen by the agent.

$Q_3$: *"Why is this policy optimal, i.e. $\pi(s) = a \wedge \pi(s) \neq a'$?"*

Such questions are pursued particularly in the context of MDPs: authors in [41] phrase explanations in terms of the frequency with which the current action would lead the agent to high-value states, while authors in [19] looked at such questions in a specific application context with explanations that show how the action allows for the execution of more desirable actions later. The latter additionally employs a case-based explanation technique to provide historical precedents about the results of the actions. Authors in [37] answer questions over $a$ being preferred over $a'$ by illustrating how the actions affect the total value in terms of various human-understandable components of the reward function. The approach in [76] justifies a policy in a multi-objective MDP by presenting alternative policies where one of the objectives is going to be higher than in the current policy and showing how improving that objective had led to an overall worse policy.

Among these works, [41; 19; 37; 20] aim for minimal explanations as a means of selection. Moreover, [37] and [20] could be considered social as they at least specifically try to frame explanations in human understandable terms.

$Q_4$: *"Why is $\Pi$ not solvable?"*

There are several ways to surface to the user the constraints in the problem that are leading to unsolvability.

**Excuses.** One approach would be to transform the given problem to a new one so that the updated problem is now solvable and provide the model fix as an explanation of why the original problem was unsolvable.

$$\Pi \mapsto \Pi' \text{ so that } \mathbb{A} : \Pi' \times \tau \not\mapsto \phi$$
$$\mathcal{E} \leftarrow \Pi \Delta \Pi'$$

These are called excuses [25]: here the authors identify a set of static initial facts to update by framing it as a planning problem. It is possible to impose selection strategies in this framework by associating costs to the various excuses.

**Abstractions.** An alternative transformation on the problem would be to find a simpler version of the given problem which is still unsolvable and highlight the problems there.

$$\mathcal{E} \leftarrow Abs(\Pi) \text{ so that } \mathbb{A} : \mathcal{E} \times \tau \mapsto \phi$$

These are called model abstractions and have been used in [21; 72; 74] to reduce the computational burden on the user. The approach in [74] also leverages temporal abstractions in the form of intermediate subgoals to illustrate why possible foils fail. Use of abstractions is, of course, not confined to explanations of unsolvability: recent work [50] used abstract models defined over simpler user-defined features to generate explanations for reinforcement learning problems in terms of action influence. The method discussed in [39] also allows for causal link explanations for abstract tasks, such as in HTN planning [63]. The use of plan properties by [20], interpretable state representations in [81], and subsets of state factors in [41] are more examples of the use of abstraction schemes to simplify the explanation process.

**Certificates.** Finally, authors in [22] look at a different way to approach the unsolvability issue by creating inductive certificates for the initial states that captures all reachable states. They have also investigated axiomatic systems that can generate proofs for task unsolvability [23]. Such certificates (represented, for example, as a binary decision diagram) can be quite complicated and are not meant to be consumed by end users, but provide useful debugging information to domain designers, algorithm designers, and AI assistants.

## 5.2 Model Reconciliation

One of the recurring themes in human-machine interaction is the "mental models" of users [6] – users of software systems often come with their own preconceived notions and expectations of the system that may ore may not be borne out by the ground truth. For a planning system, this means that even if it is making the best plans it could, the human-in-the-loop is evaluating those plans with a different model, i.e. their mental model of the problem, and may not agree to its quality. Differences in models between the user and the machine appear in many settings, such as in drifting world models over long terms interactions [5], search and rescue settings where there are internal and external agents with different views into the world [13], in intelligent tutoring systems between the student and the instructor [28], in smart rooms with distributed sensors [8], and so on. This model difference, along with inferential limitations of the human, is thus the root cause of the need for explanations from the end user persona.

In [15], the original work on this topic, we posit that explanations can no longer be a "soliloquy" in the agent's own model but must instead consider and explain in terms of these model differences. The process of explanations is then one of *reconciliation* of the systems model and the human mental

model so that both can agree on the property $\tau$ of the decision being made. Thus, if $\Pi^H$ is the mental model of the user, the model reconciliation process requires that:

Given: $\mathbb{A} : \Pi \times \tau \mapsto \pi$

$\Pi^H + \mathcal{E} \rightarrow \hat{\Pi}^H$ such that $\mathbb{A} : \hat{\Pi}^H \times \tau \mapsto \pi$.

In the original work, the mental model was assumed to be known and reconciliation was achieved through a search in the space of models induced by the difference between the system model and the mental model, until a model is found where $\tau$ holds. The difference between this intermediate model and the mental model is provided as an explanation.

**Social.** Such explanations are inherently social in being able to explicitly capture the effect of expectations in the explanation process. In user studies conducted in [12], it was shown that participants were indeed able to identify the correct $\tau$ based on an explanation. Note that, in the model reconciliation framework, the mental model is just a version of the decision making problem at hand which the agent believes the user is operating under. This may be a graph, a planning problem, or even a logic program. The notion of model reconciliation is agnostic to the actual representation.

**Contrastive.** The contrastive nature of these explanations comes from how the model update preserves $\tau$ of the given plan as opposed to the foil, which may be implicitly [15] or explicitly [72] provided. This is also closely tied with the selection process of those model updates.

**Selective.** In [15], the explanation content was selected based on minimality of model update: $\min |\Pi \Delta \Pi^H|$. The minimal explanation is not unique and it was shown in [86] how users attribute different value to theoretically equivalent model updates, thereby motivating further research on how to select among several competing explanations for the user.

**Model Reconciliation Expansion Pack**

The last couple of years have seen extensive work on this topic, primarily focused on relaxing the assumptions made on the mental model in the original model reconciliation work, and expanding the scope of problems addressed by it (such as in the explanation of logic programs [65; 79]).

**Model Uncertainty.** One of the primary directions of work has been in considering uncertainty about the mental model. In [71], authors show how to reconcile with a set of possible mental models $\{\Pi_i^H\}$ and also demonstrate how the same framework can be used to explain to multiple users in the loop. In [72], on the other hand, the authors estimate the mental model from the provided foil.

**Inference Reconciliation.** The original work on model reconciliation assumed an user with identical inferential capability (optimal or sound as the case may be) to the planner. However, as we saw previously, much of XAIP has been about dealing with the computational limits of users. Model reconciliation approaches have started adapting to this [72; 74] by identifying from the given foil the simplest abstraction of their model to explain in. [74] provides further inferential assistance in the form of unmet subgoals.

**Unsolvability.** An important aspect of human-planner interaction, where inferential limitations play an outsized part, is the case of unsolvability. An interesting case of this is recently explored in [69] where the domain acquisition problem has been cast into the model reconciliation framework, reusing [74] to help out the domain designer persona when they cannot figure out why their domain has no solutions or the solutions do not match their expectation.

**Model-free Model Reconciliation.** So far, model reconciliation has considered the mental model explicitly. This may not be necessary. At the end of the day, the explanation includes information regarding the agent model and what it include and do not include. The mental model only helps the system to filter what new information is relevant to the user. Thus an alternative would be to predict how model information can affect the expectation of the user [70] by learning a labeling model that takes a state-action-state tuple, a subset of information about the system's model and whether the user after receiving the information would find this tuple explicable. The learned model then drives the search to determine what information should be exposed to the user.

**Lies and Deception.** In the original work on model reconciliation, $\mathcal{E}$ was always constrained to be consistent with the ground truth $\Pi$. In [10; 9] we showed how this can be relaxed to hijack the model reconciliation process into producing false explanations, opening up intriguing avenues of research on the ethics of mental modeling in planning. This can be an issue in model-free model reconciliation as well, in that explanations no longer need to be faithful to the original model but instead be whatever the explainee finds satisfying.

## 6 Plan-based Explanations

Finally, we look at the role of plans in explanatory dialogue. Works like [52; 67] have explored explanations in the form of a plan that explains a set of observations, while methods like [78] have looked at ways to generate the most likely explanation for why a plan failed. Beyond inferential support in human-AI interaction, the qualitative structure of plans has also been used for plan-reuse and validation [38].

**Plan / Policy Summarization.** With regards to the role of plans in explanatory dialogue, one area we want to highlight in greater detail is that of plan or policy summarization. When the system is generating solutions over long time horizons and over large state spaces, presentation of the plan or policy to the user becomes difficult. One way to approach this issue is through verbalization of plans: e.g. paths taken by a robot [61] along different dimensions of interest such as levels of abstraction, specificity, and locality. Recent work has also attempted at domain-independent methods for plan summarization [8] by using the model reconciliation process with an empty mental model to compute the minimal subset of causal links required to justify each action in a plan.

Abstraction schemes can also simplify the decision structure and allow the user to drill down as required. [77] looks at the possibility of employing state abstraction that project out low importance features. On the other hand, [73] generates temporal abstractions for a given policy by automati-

| Explanation Type | | Social | Contrastive | Selective | Local | Global | Abstraction | User Study |
|---|---|---|---|---|---|---|---|---|
| Algorithm-based explanations | | [37; 2] | | [37; 27; 44; 2; 36] | [37; 27; 51; 44; 2; 36] | | [44] | [27; 51; 2] |
| Model-Based Explanations | Inference Reconciliation | [72; 74; 20; 69; 50; 37; 81; 76; 2] | [63; 4; 72; 74; 20; 69; 25; 41; 19; 22; 50; 37; 7; 84; 83; 76; 81; 57; 1; 2; 88] | [27; 63; 4; 72; 74; 20; 69; 25; 41; 19; 50; 37; 83; 36] | [63; 27; 4; 72; 74; 20; 69; 25; 41; 19; 22; 50; 37; 7; 84; 83; 76; 81; 36; 57; 1; 78; 2; 88] | [74; 20; 69; 25] | [72; 74; 20; 69; 41; 50; 81; 76] | [4; 27; 74; 41; 19; 50; 84; 76; 81; 57; 2] |
| | Model Reconciliation | [15; 72; 74; 69; 71; 70; 13; 68; 8; 79; 10; 56] | [15; 72; 74; 69; 71; 70; 13; 68; 8; 79; 10; 56] | [15; 72; 74; 69; 71; 70; 13; 68; 8; 79; 10; 56] | [15; 72; 71; 70; 13; 68; 8] | [15; 74; 69] | [72; 74; 69] | [12; 74; 70; 13; 10] |
| Plan-based explanations | | [47; 8; 61; 33] | | [85; 44; 73; 77; 33; 47; 43; 8; 61; 78] | [85; 44; 73; 77; 33; 47; 43; 8; 61; 38; 80; 39; 78; 16] | | [85; 44; 73; 77; 33; 61] | [73; 47] |

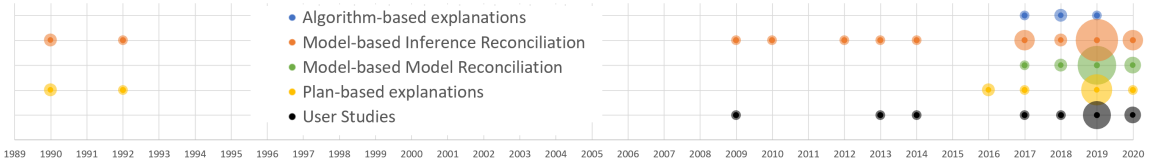


Figure 1: Summary of results: size of a circle is proportional to number of papers in a year, smallest being 1 (2020 is still in progress).

cally extracting subgoals. The approach in [85] takes advantage of both schemes by mapping policies learned through Deep Q-learning methods to a policy for a semi-aggregated MDP that employs both user-specified state aggregation features and temporally extended actions in the form of skills automatically generated from the learned policy. A related work [16], looks at the use of soft decisions trees to create hierarchical representations of policies. A different approach is taken in [47] where users are presented partial plans that they can figure out completions of, based on their knowledge of the task. This is done by using various psychologically feasible computational models.

Another possibility would be to allow the user to ask questions about generated policies: e.g. *"Under what conditions is action $a_i$ performed"?* This was investigated in [33], where both queries and answers were expressed in terms of user-specified features. [43] looked at cases where the user is not just interested in learning details of the model underlying the current decisions but rather how it differs from possible alternatives, by using LTL formulas that are true in a target set of plan traces but are not satisfied by a specified alternate set.

## 7 Concluding Remarks

This concludes our discussion on how the different components of a decision making problem can contribute to an explanation of its solution – the problem definition, the algorithm, and artifacts of the solution itself. We will end with a few remarks on emerging trends in the world of XAIP.

**Explainability and Communication.** In this paper, we focused mostly on the *content* of explanations in the explanatory process – we did not touch on how such content is communicated to the user. Existing work has looked at a variety of modalities including natural language [61], and graphical [8; 29] and mixed-reality interfaces [14] (or a mixture of both [64]). XAIP and user experience (UX) design are inseparable topics [58] going forward as planning and technologies mature and come into contact with end users. In fact, the International Conference on Automated Planning and Scheduling (ICAPS) – the premier conference on planning – is going to oversee an Explaianble AI Planning (XAIP) Workshop with this special theme in 2020 as a joint effort with the organizers of the formerly User Interfaces and Scheduling and Planning (UISP) Workshop at the same venue, hopefully fostering new exciting research in this emerging direction.

**Emerging Landscape.** While the works explored here are mostly after-the-fact explanations, i.e. after a plan has been computed (or no plan has been found), in recent work [13] we demonstrated how the possibility of having to explain its decisions can be folded into an agent's reasoning stage itself. Authors in [75] have also made similar efforts to combining consideration of explanations into the representation of planning problems. This is a well-known phenomenon in human behavior: we are known to make better decisions when we are asked to explain them [53]. By adopting a similar philosophy, we can potentially achieve better, more human-aware, behavior in XAIP-enabled agents as well, thereby opening up a whole new horizon in automated planning techniques.

Early attempts at this, employing search in the space of models [13], had proved computationally prohibitive. However, recent work [68] has shown that achieving such behavior is computationally no harder than its classical planning counterpart! Furthermore, recognizing that plans are not made in vacuum but often in the context of interactions with end users, *can lead to a more efficient planning process with explainable components than without*, for example, in collaborative planning scenarios [29] or in anytime planners that can preserve high-level constraints in partial plans as it plans along [26]. As the XAIP community comes to terms with its own accuracy versus efficiency trade-offs, parallel to similar arguments in the XAI community at large, a whole new world of possibilities open up in imbuing established planning approaches with the latest and best XAIP-components.

## Acknowledgements

Kambhampati's research is supported in part by ONR grants N00014-16-1-2892, N00014-18-1-2442, N00014-18-1-2840, and N00014-19-1-2119, an AFOSR grant FA9550-18-1-0067, a DARPA SAIL-ON grant W911NF-19-2-0006, NSF grants 1936997 (C-ACCEL), 1844325, and a NASA grant NNX17AD06G.

## References

[1] Tobias Ahlbrecht and Michael Winikoff. Explaining Aggregate Behaviour in Cognitive Agent Simulations Using Explanation. In *EXTRAAMAS Workshop*, 2019.

[2] Andrew Anderson, Jonathan Dodge, Amrita Sadarangani, Zoe Juozapaitis, Evan Newman, Jed Irvine, Souti Chattopadhyay, Alan Fern, and Margaret Burnett. Explaining Reinforcement Learning to Mere Mortals: An Empirical Study. In *IJCAI*, 2019.

[3] Sule Anjomshoae, Amro Najjar, Davide Calvaresi, and Kary Främling. Explainable Agents and Robots: Results from a Systematic Literature Review. In *AAMAS*, 2019.

[4] Pascal Bercher, Susanne Biundo, Thomas Geier, Thilo Hoernle, Florian Nothdurft, Felix Richter, and Bernd Schattenberg. Plan, Repair, Execute, Explain – How Planning Helps to Assemble Your Home Theater. In *ICAPS*, 2014.

[5] Dan Bryce, J Benton, and Michael W Boldt. Maintaining Evolving Domain Models. In *IJCAI*, 2016.

[6] John M Carroll and Judith Reitman Olson. Mental models in human-computer interaction. *Handbook of Human-Computer Interaction*, 1988.

[7] Michael Cashmore, Anna Collins, Benjamin Krarup, Senka Krivic, Daniele Magazzeni, and David Smith. Towards Explainable AI Planning as a Service. In *XAIP Workshop*, 2019.

[8] Tathagata Chakraborti, Kshitij P. Fadnis, Kartik Talamadupula, Mishal Dholakia, Biplav Srivastava, Jeffrey O. Kephart, and Rachel K. E. Bellamy. Planning and Visualization for a Smart Meeting Room Assistant – A Case Study in the Cognitive Environments Laboratory at IBM T.J. Watson Research Center, Yorktown. *AI Communication*, 2019.

[9] Tathagata Chakraborti and Subbarao Kambhampati. (How) Can AI Bots Lie? In *XAIP Workshop*, 2019.

[10] Tathagata Chakraborti and Subbarao Kambhampati. (When) Can AI Bots Lie? In *AIES/AAAI*, 2019.

[11] Tathagata Chakraborti, Anagha Kulkarni, Sarath Sreedharan, David E Smith, and Subbarao Kambhampati. Explicability? Legibility? Predictability? Transparency? Privacy? Security? The Emerging Landscape of Interpretable Agent Behavior. In *ICAPS*, 2019.

[12] Tathagata Chakraborti, Sarath Sreedharan, Sachin Grover, and Subbarao Kambhampati. Plan Explanations as Model Reconciliation – An Empirical Study. In *HRI*, 2019.

[13] Tathagata Chakraborti, Sarath Sreedharan, and Subbarao Kambhampati. Balancing Explanations and Explicability in Human-Aware Planning. In *IJCAI*, 2019.

[14] Tathagata Chakraborti, Sarath Sreedharan, Anagha Kulkarni, and Subbarao Kambhampati. Projection-Aware Task Planning and Execution for Human-in-the-Loop Operation of Robots in a Mixed-Reality Workspace. In *IROS*, 2018.

[15] Tathagata Chakraborti, Sarath Sreedharan, Yu Zhang, and Subbarao Kambhampati. Plan Explanations as Model Reconciliation: Moving Beyond Explanation as Soliloquy. In *IJCAI*, 2017.

[16] Youri Coppens, Kyriakos Efthymiadis, Tom Lenaerts, Ann Nowé, T Miller, R Weber, and D Magazzeni. Distilling Deep Reinforcement Learning Policies in Soft Decision Trees. In *XAI Workshop*, 2019.

[17] Dustin Dannenhauer, Michael W Floyd, Daniele Magazzeni, and David W Aha. Explaining Rebel Behavior in Goal Reasoning Agents. In *XAIP Workshop*, 2018.

[18] Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *IEEE Symposium on Security and Privacy (SP)*, 2016.

[19] Thomas Dodson, Nicholas Mattei, Joshua T. Guerin, and Judy Goldsmith. An English-Language Argumentation Interface for Explanation Generation with Markov Decision Processes in the Domain of Academic Advising. *TiiS*, 2013.

[20] Rebecca Eifler, Michael Cashmore, Jörg Hoffmann, Daniele Magazzeni, and Marcel Steinmetz. A New Approach to Plan-Space Explanation: Analyzing Plan-Property Dependencies in Oversubscription Planning. In *AAAI*, 2020.

[21] Thomas Eiter, Zeynep G Saribatur, and Peter Schüller. Abstraction for Zooming-In to Unsolvability Reasons of Grid-Cell Problems. In *XAI Workshop*, 2019.

[22] Salomé Eriksson, Gabriele Röger, and Malte Helmert. Unsolvability Certificates for Classical Planning. In *ICAPS*, 2017.

[23] Salomé Eriksson, Gabriele Röger, and Malte Helmert. A Proof System for Unsolvable Planning Tasks. In *ICAPS*, 2018.

[24] Maria Fox, Derek Long, and Daniele Magazzeni. Explainable Planning. In *XAI Workshop*, 2017.

[25] Moritz Göbelbecker, Thomas Keller, Patrick Eyerich, Michael Brenner, and Bernhard Nebel. Coming Up with Good Excuses: What to Do When No Plan Can be Found. In *ICAPS*, 2010.

[26] Antoine Grea, Laëtitia Matignon, and Samir Aknine. How Explainable Plans Can Make Planning Faster. In *XAI Workshop*, 2018.

[27] Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. Visualizing and Understanding Atari Agents. In *ICML*, 2018.

[28] Sachin Grover, Tathagata Chakraborti, and Subbarao Kambhampati. What Can Automated Planning do for Intelligent Tutoring Systems? In *ICAPS Scheduling and Planning Applications Workshop*, 2018.

[29] Sachin Grover, Sailik Sengupta, Tathagata Chakraborti, Aditya Prasad Mishra, and Subbarao Kambhampati. RADAR: Automated Task Planning for Proactive Decision Support. *HCI Journal*, 2020.

[30] David Gunning. Explainable Artificial Intelligence (XAI). *Defense Advanced Research Projects Agency (DARPA)*, 2017.

[31] David Gunning and David W Aha. DARPA's Explainable Artificial Intelligence Program. *AI Magazine*, 2019.

[32] Marc Hanheide, Moritz Göbelbecker, Graham S Horn, Andrzej Pronobis, Kristoffer Sjöö, Alper Aydemir, Patric Jensfelt, Charles Gretton, Richard Dearden, Miroslav Janicek, et al. Robot Task Planning and Explanation in Open and Uncertain Worlds. *Artificial Intelligence*, 2017.

[33] Bradley Hayes and Julie A Shah. Improving Robot Controller Transparency Through Autonomous Policy Explanation. In *HRI*, 2017.

[34] Jörg Hoffmann and Daniele Magazzeni. Explainable AI Planning (XAIP): Overview and the Case of Contrastive Explanation. In *Reasoning Web. Explainable Artificial Intelligence*, 2019. Extended Abstract.

[35] R. Howey, D. Long, and M. Fox. VAL: Automatic Plan Validation, Continuous Effects and Mixed Initiative Planning Using PDDL. In *ICTAI*, 2004.

[36] Tobias Huber and Elisabeth André. Introducing Selective Layer-Wise Relevance Propagation to Dueling Deep Q-learning. In *XAI Workshop*, 2019.

[37] Zoe Juozapaitis, Anurag Koul, Alan Fern, Martin Erwig, and Finale Doshi-Velez. Explainable Reinforcement Learning via Reward Decomposition. In *XAI Workshop*, 2019.

[38] Subbarao Kambhampati. A Classification of Plan Modification Strategies Based on Coverage and Information Requirements. In *AAAI Spring Symposium on Case Based Reasoning*, 1990.

[39] Subbarao Kambhampati. Mapping and Retrieval During Plan Reuse: A Validation Structure Based Approach. In *AAAI*, 1990.

[40] Subbarao Kambhampati and James A Hendler. Flexible reuse of plans via annotation and verification. In *the International Conference on Artificial Intelligence Applications*, 1989.

[41] Omar Zia Khan, Pascal Poupart, and James P Black. Minimal Sufficient Explanations for Factored Markov Decision Processes. In *ICAPS*, 2009.

[42] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In *ICML*, 2018.

[43] Joseph Kim, Christian Muise, Ankit Shah, Shubham Agarwal, and Julie Shah. Bayesian Inference of Linear Temporal Logic Specifications for Contrastive Explanations. In *IJCAI*, 2019.

[44] Anurag Koul, Sam Greydanus, and Alan Fern. Learning Finite State Representations of Recurrent Policy Networks. In *ICLR*, 2018.

[45] Benjamin Krarup, Michael Cashmore, Daniele Magazzeni, and Tim Miller. Model-Based Contrastive Explanations for Explainable Planning. In *XAIP Workshop*, 2019.

[46] Anagha Kulkarni, Sarath Sreedharan, Sarah Keren, Tathagata Chakraborti, David E. Smith, and Subbarao Kambhampati. Design for Interpretability. In *XAIP Workshop*, 2019.

[47] Isaac Lage, Daphna Lifschitz, Finale Doshi-Velez, and Ofra Amir. Exploring Computational User Models for Agent Policy Summarization. In *IJCAI*, 2019.

[48] Pat Langley. Varieties of explainable agency. In *XAIP Workshop*, 2019.

[49] Pat Langley, Ben Meadows, Mohan Sridharan, and Dongkyu Choi. Explainable Agency for Intelligent Autonomous Systems. In *IAAI/AAAI*, 2017.

[50] Prashan Madumal, Tim Miller, Liz Sonenberg, and Frank Vetere. Explainable Reinforcement Learning Through a Causal Lens. In *AAAI*, 2020.

[51] Maurıcio C Magnaguagno, Ramon Fraga Pereira, Martin D Móre, and Felipe Meneguzzi. Web Planner: A Tool to Develop Classical Planning Domains and Visualize Heuristic State-Space Search. In *ICAPS Workshop on User Interfaces in Scheduling and Planning*, 2017.

[52] Ben Leon Meadows, Pat Langley, and Miranda Jane Emery. Seeing Beyond Shadows: Incremental Abductive Reasoning for Plan Understanding. In *AAAI Workshop on Plan, Activity, and Intent Recognition (PAIR)*, 2013.

[53] Hugo Mercier and Dan Sperber. Why do Humans Reason? Arguments for an Argumentative Theory. *Behavioral and Brain Sciences*, 2011.

[54] Tim Miller. Contrastive Explanation: A Structural-Model Approach. *arXiv:1811.03163*, 2018.

[55] Tim Miller. Explanation in Artificial Intelligence: Insights from the Social Sciences. *Artificial Intelligence*, 2019.

[56] Van Nguyen, Stylianos Loukas Vasileiou, Tran Cao Son, and William Yeoh. Conditional Updates of Answer Set Programming and Its Application in Explainable Planning. In *AAMAS*, 2020. Extended Abstract.

[57] Matthew L Olson, Lawrence Neal, Fuxin Li, and Weng-Keen Wong. Counterfactual States for Atari Agents via Generative Deep Learning. In *XAI Workshop*, 2019.

[58] R. G. Freedman, T. Chakraborti, K. Talamadupula, D. Magazzeni and J. D. Frank. User Interfaces and Scheduling and Planning: Workshop Summary and Proposed Challenges. In *AAAI Spring Symposium on Designing the User Experience of Artificial Intelligence*, 2018.

[59] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. In *KDD*, 2016.

[60] Mark Roberts, Isaac Monteath, Raymond Sheh, David Aha, Piyabutra Jampathom, Keith Akins, Eric Sydow, Vikas Shivashankar, and Claude Sammut. What was I planning to do? In *XAIP Workshop*, 2018.

[61] Stephanie Rosenthal, Sai P Selvaraj, and Manuela M Veloso. Verbalization: Narration of Autonomous Robot Experience. In *IJCAI*, 2016.

[62] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. *arXiv:1708.08296*, 2017.

[63] Bastian Seegebarth, Felix Müller, Bernd Schattenberg, and Susanne Biundo. Making Hybrid Plans More Clear to Human Users – A Formal Approach for Generating Sound Explanations. In *ICAPS*, 2012.

[64] Sailik Sengupta, Tathagata Chakraborti, and Subbarao Kambhampati. Ma-radar–a mixed-reality interface for collaborative decision making. *ICAPS Workshop on User Interfaces and Scheduling and Planning*, 2018.

[65] Maayan Shvo, Toryn Q Klassen, and Sheila A McIlraith. Towards the Role of Theory of Mind in Explanation. In *EXTRAAMAS Workshop*, 2020.

[66] David E Smith. Choosing Objectives in Over-Subscription Planning. In *ICAPS*, 2004.

[67] Shirin Sohrabi, Jorge A Baier, and Sheila A McIlraith. Preferred Explanations: Theory and Generation via Planning. In *AAAI*, 2011.

[68] Sarath Sreedharan, Tathagata Chakraborti, Christian Muise, and Subbarao Kambhampati. Expectation-Aware Planning: A Unifying Framework for Synthesizing and Executing Self-Explaining Plans for Human-Aware Planning. In *AAAI*, 2020.

[69] Sarath Sreedharan, Tathagata Chakraborti, Christian Muise, Yasaman Khazaeni, and Subbarao Kambhampati. D3WA+: A Case Study of XAIP in a Model Acquisition Task. In *ICAPS*, 2020.

[70] Sarath Sreedharan, Alberto Olmo Hernandez, Aditya Prasad Mishra, and Subbarao Kambhampati. Model-Free Model Reconciliation. In *IJCAI*, 2019.

[71] Sarath Sreedharan, Subbarao Kambhampati, et al. Handling Model Uncertainty and Multiplicity in Explanations via Model Reconciliation. In *ICAPS*, 2018.

[72] Sarath Sreedharan, Siddharth Srivastava, and Subbarao Kambhampati. Hierarchical Expertise Level Modeling for User Specific Contrastive Explanations. In *IJCAI*, 2018.

[73] Sarath Sreedharan, Siddharth Srivastava, and Subbarao Kambhampati. TLdR: Policy Summarization for Factored SSP Problems Using Temporal Abstractions. In *ICAPS*, 2020.

[74] Sarath Sreedharan, Siddharth Srivastava, David Smith, and Subbarao Kambhampati. Why Can't You Do That HAL? Explaining Unsolvability of Planning Tasks. In *IJCAI*, 2019.

[75] Roykrong Sukkerd, Reid Simmons, and David Garlan. Toward Explainable Multi-Objective Probabilistic Planning. In *ICSE Workshop on Software Engineering for Smart Cyber-Physical Systems (SEsCPS)*, 2018.

[76] Roykrong Sukkerd, Reid Simmons, and David Garlan. Tradeoff-Focused Contrastive Explanation for MDP Planning. *arXiv:2004.12960*, 2020.

[77] Nicholay Topin and Manuela Veloso. Generation of Policy-Level Explanations for Reinforcement Learning. In *AAAI*, 2019.

[78] Gianluca Torta, Roberto Micalizio, and Samuele Sormano. Temporal multiagent plan execution: Explaining what happened. In *EXTRAAMAS Workshop*, 2019.

[79] Stylianos Vasileiou, William Yeoh, and Tran Cao Son. A Preliminary Logic-based Approach for Explanation Generation. In *XAIP Workshop*, 2019.

[80] Manuela M Veloso. Learning by Analogical Reasoning in General Problem Solving. *Doctoral Thesis*, 1992.

[81] J Waa, J van Diggelen, K Bosch, and M Neerincx. Contrastive Explanations for Reinforcement Learning in Terms of Expected Consequences. In *XAI Workshop*, 2018.

[82] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. *Harvard Journal of Law & Technology*, 2017.

[83] Michael Winikoff. Debugging Agent Programs with "Why?" Questions. In *AAMAS*, 2017.

[84] Michael Winikoff, Virginia Dignum, and Frank Dignum. Why Bad Coffee? Explaining Agent Plans with Valuings. In *SafeComp*, 2018.

[85] Tom Zahavy, Nir Ben-Zrihem, and Shie Mannor. Graying the Black Box: Understanding DQNs. In *ICML*, 2016.

[86] Zahra Zahedi, Alberto Olmo, Tathagata Chakraborti, Sarath Sreedharan, and Subbarao Kambhampati. Towards Understanding User Preferences for Explanation Types in Explanation as Model Reconciliation. In *HRI*, 2019. Late Breaking Report.

[87] Yu Zhang, Sarath Sreedharan, Anagha Kulkarni, Tathagata Chakraborti, Hankz Hankui Zhuo, and Subbarao Kambhampati. Plan Explicability and Predictability for Robot Task Planning. In *ICRA*, 2017.

[88] Ellin Zhao and Roykrong Sukkerd. Interactive Explanation for Planning-Based Systems: WIP Abstract. In *ICCPS*, 2019.

[89] Yishan Zhou and David Danks. Different "Intelligibility" for Different Folks. In *AIES/AAAI*, 2020.