Evaluating Plan-Property Dependencies: A Web-Based Platform and User Study

Rebecca Eifler,[1] **Martim Brandao,**[2] **Amanda Coles,**[2] **Jeremy Frank,**[3] **Jörg Hoffmann**[1,4]

[1]Saarland University, Saarland Informatics Campus, Germany
[2]King's College London, UK
[3]NASA Ames Research Center, Mountain View, CA, USA
[4]German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany
{eifler, hoffmann}@cs.uni-saarland.de, {amanda.coles, martim.brandao}@kcl.ac.uk, Jeremy.D.Frank@nasa.gov

Abstract

The trade-offs between different desirable plan properties – e. g. PDDL temporal plan preferences – are often difficult to understand. Recent work addresses this by iterative planning with explanations elucidating the dependencies between such plan properties. Users can ask questions of the form "Why does the plan not satisfy property p?", which are answered by "Because then we would have to forego q". It has been shown that such dependencies can be computed reasonably efficiently. But is this form of explanation actually useful for users? We run a large crowd-worker user study ($N = 100$ in each of 3 domains) evaluating that question. To enable such a study in the first place, we contribute a Web-based platform for iterative planning with explanations, running in standard browsers. Comparing users with vs. without access to the explanations, we find that the explanations enable users to identify better trade-offs between the plan properties, indicating an improved understanding of the planning task.

1 Introduction

Explainable AI planning (XAIP) is a growing sub-area of planning (e. g.(Göbelbecker et al. 2010; Seegebarth et al. 2012; Fox, Long, and Magazzeni 2017; Chakraborti et al. 2017; Behnke et al. 2019; Sreedharan et al. 2019b,a; Chakraborti and Kambhampati 2019; Krarup et al. 2019; Sreedharan, Srivastava, and Kambhampati 2020)). We refer the reader to (Chakraborti et al. 2019a) for a survey.

In this work, we are concerned with a particular form of XAIP, proposed recently by Eifler et al. (2020a; 2020b) (henceforth *Eif20*), that addresses dependencies between desirable *plan properties*. The targeted context are scenarios where these properties are partially conflicting and where one or multiple users need to make up their mind on the best trade-off. For example, a property initially perceived to be important may be re-evaluated if it turns out to be a bottleneck excluding many other properties. In such a setting, users need to understand the conflicts to converge to a utility function or otherwise acceptable trade-off. Hence an *iterative planning* process as proposed by Smith (2012) is adequate, where users iteratively refine example plans π.

Eif20's explanation framework supports user questions about a given plan π in such iterative planning: Why does π not satisfy a property I care about? What alternatives are there? Specifically, Eif20 assume a fixed set P of plan properties expressed as LTLf formulas (De Giacomo, De Masellis, and Montali 2014). They compute *dependencies* between those, in the form of plan-space entailments: given $p, q \in P$, p entails $\neg q$ in plan space if all plans that satisfy p do not satisfy q. A user question "Why does π not satisfy p?" is then answered by "Because if it did we would have to forego q".

Eif20 show that the set of all plan-property dependencies can be computed reasonably efficiently. But are the resulting explanations actually useful to users? We contribute a large user study evaluating that question, in terms of test-person performance in several case studies on iterative planning.

The ideal user study would be run with real-life experts. However, experts are notoriously hard to come by in basic XAIP research, for various reasons including the abstract and non-applied nature of the evaluated technology. Previous XAIP user studies hence resorted to university students or crowd workers. Here we do the latter as it facilitates large user numbers N. We specifically use Prolific (https://www.prolific.co/) (Palan and Schitter 2018), which is more suited for longer user studies, with complex tasks, than for example Amazon Mechanical Turk.

To enable this user study in the first place, we also contribute a Web-based platform for iterative planning with explanations [1]. The platform runs in standard browsers and is therefore ideally accessible for crowd working studies.

We run the study on three different planning domains, including a new domain "Parent's Afternoon" that encodes the everyday problem of family logistics, familiar to layperson users. In each domain, we carefully design a use case encoding preference trade-offs complex enough to render the plan-property dependencies non-trivial to understand, while easy enough to be solved within the limited time span crowd workers are willing to invest (less than an hour).

We run our user study with $N = 100$ test persons on each domain, split into two equal-size groups having vs. not having access to Eif20's explanation facilities. Our results show that users with access to explanations tend to identify better trade-offs between the plan properties, indicating an improved understanding of the planning task.

[1] The source code and user study data are available at: https://github.com/XPP-explainable-planning

2 Planning Context and Iterative Planning

Our investigation is placed in the context of oversubscription planning (Smith 2004; Domshlak and Mirkis 2015). An OSP planning task (short: OSP task) τ defines an initial state I and actions A over a set of state variables (or propositions/facts) via a value assignment and precondition/effect pairs respectively. There is an action-cost budget b, i. e., action sequences whose summed up cost exceeds b are not allowed. There is a set G^{hard} of hard goals (state-variable values) that must be achieved; and a set G^{soft} of soft goals that are of interest but are not mandatory.

In contrast to standard OSP frameworks, we do not define a utility over G^{soft}. Instead, G^{soft} represents a set of plan properties, specifically LTLf plan-preference formulas compiled into (soft-)goal facts (Baier and McIlraith 2006; Edelkamp 2006; Eifler et al. 2020b). The explanation facility by Eif20 that we evaluate in our user study identifies dependencies between these plan properties.

Eif20 identify dependencies between $X, Y \subseteq G^{\text{soft}}$, where $\bigwedge_{g \in X} g \Rightarrow \neg \bigwedge_{g \in Y} g$. Such a dependency holds if all action sequences in τ with cost $\leq b$, that achieve G^{hard} as well as all $g \in X$, do not achieve at least one $g \in Y$. The strongest such dependencies correspond exactly to **minimal unsolvable goal subsets (MUGS)** $X \cup Y = G \subseteq G^{\text{soft}}$ where G cannot be achieved but every $G' \subsetneq G$ can. Eif20's algorithms compute all MUGS, as an offline process that prepares the answers to all possible user questions.

We implement an iterative planning process where each iteration proceeds as follows: 1) the user selects a subset $G_{\text{enf}} \subseteq G^{\text{soft}}$ to be enforced; 2) a new plan that satisfies $G^{\text{hard}} \cup G_{\text{enf}}$ is computed; 3) the user can optionally ask questions about that plan; 4) the user selects a new G_{enf}.

Eif20's method facilitates step 3) Specifically, there are three possible situations, with corresponding explanations:

(A) Question: "Why does π not satisfy p?"

Situation: G_{enf} solvable, current plan π satisfies $G_{\text{sat}} \subseteq G^{\text{soft}}$ with $G_{\text{enf}} \subseteq G_{\text{sat}}, p \notin G_{\text{sat}}$; $G_{\text{enf}} \cup \{p\}$ unsolvable.

Answer: List of sets G_i where $G_i \subseteq G_{\text{sat}}$ and there exists a MUGS G with $p \cup G_i = G$.

Example: "Why can I not go shopping before sports?" – "Because then you cannot bring your kid to the music school."

(B) Question: "Why does π not satisfy p?"

Situation: G_{enf} solvable, current plan π satisfies $G_{\text{sat}} \subseteq G^{\text{soft}}$ with $G_{\text{enf}} \subseteq G_{\text{sat}}$, $p \notin G_{\text{sat}}$; but $G_{\text{enf}} \cup \{p\}$ is solvable (i. e., p was not enforced but could be).

Answer: "Actually yes, we can satisfy p in addition to the currently enforced properties."

Example: "Why can I not bring my friend to the sport center?" – "Actually yes, you can do that in addition to bringing your kid to the music school."

(C) Question: "Why is there no plan?"

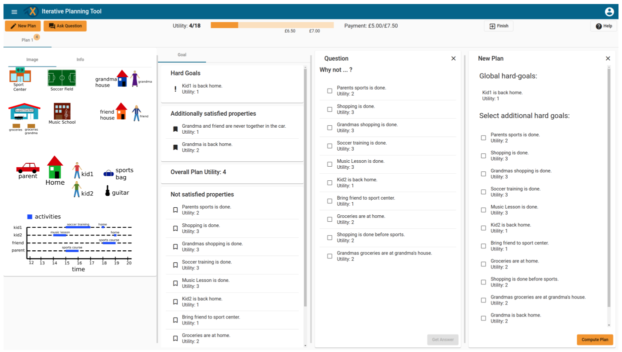
Situation: G_{enf} unsolvable.

Answer: List of MUGS G where $G \subseteq G_{\text{enf}}$.

Example: "Because you cannot bring your friend to the sports center, and bring your kid to the music school, and bring grandma to the supermarket."

3 Web-Based Iterative Planning Tool

We implemented a Web-based tool for iterative planning with eXplanation through Plan Properties, short **XPP**. XPP features an interface for end-users doing iterative planning but also supports developers in user-study design.



Hard Goals

! Kid1 is back home.
Utility: 1

Additionally satisfied properties

Grandma and friend are never together in the car.
Utility: 1

Grandma is back home.
Utility: 2

Overall Plan Utility: 4

Not satisfied properties

Parents sports is done.
Utility: 2

Shopping is done.
Utility: 3

Grandmas shopping is done.
Utility: 3

Soccer training is done.
Utility: 3

Music Lesson is done.
Utility: 3

Kid2 is back home.
Utility: 1

Bring friend to sport center.
Utility: 1

Groceries are at home.
Utility: 2

Question

Why not ... ?

□ Parents sports is done.
Utility: 2

□ Shopping is done.
Utility: 3

□ Grandmas shopping is done.
Utility: 3

□ Soccer training is done.
Utility: 3

□ Music Lesson is done.
Utility: 3

□ Kid2 is back home.
Utility: 1

□ Bring friend to sport center.
Utility: 1

□ Groceries are at home.
Utility: 2

□ Shopping is done before sports.
Utility: 2

□ Grandmas groceries are at grandma's house.
Utility: 2

New Plan

Global hard-goals:

Kid1 is back home.
Utility: 1

Select additional hard goals:

□ Parents sports is done.
Utility: 2

□ Shopping is done.
Utility: 3

□ Grandmas shopping is done.
Utility: 3

□ Soccer training is done.
Utility: 3

□ Music Lesson is done.
Utility: 3

□ Kid2 is back home.
Utility: 1

□ Bring friend to sport center.
Utility: 1

□ Groceries are at home.
Utility: 2

□ Shopping is done before sports.
Utility: 2

□ Grandmas groceries are at grandma's house.
Utility: 2

□ Grandma is back home.
Utility: 2

Figure 1: Screenshot of the XPP tool from the user's (test persons's) perspective.

As Figure 1 shows, XPP's interface is divided into 4 columns. In the first column, the user has access to the image and textual description of the OSP task. The second column lists G_{enf} of the currently selected plan. The third and fourth columns contain the interfaces for asking questions, and for selecting G_{enf} for the next iteration. For layperson users, the plan properties are depicted in natural language.

4 Planning Domains and OSP Tasks

To cover different sources of conflicts between soft-goal preferences, we implemented user studies in three different domains. [2] We introduce a new domain, called Parents Afternoon, that encodes family logistics familiar to lay users: driving children to sports events, shopping, etc. The source of conflicts are pick-up/drop-off/opening times. Our two other domains, Transport and Rovers, are variations of standard benchmarks (IPC NoMystery and Rovers). Transport encodes transportation of packages on a road map with fuel consumption, so that soft goals compete for the same consumed resource. Rovers encodes data collection and transmission on Mars, constrained by both resource consumption and timing constraints for data uploads. Note that competition for resources is a ubiquitous source of conflict – consider money for instance – so this structure also is natural and should be familiar to lay users to a certain degree.

The complexity of the domain instances, i. e., the OSP tasks, requires careful attention. The task and plan-property dependencies must be sufficiently complex to be interesting, yet must be feasible for users in crowd-sourcing (who otherwise tend to give up quickly, resulting in unusable data). Given the exponential nature of the underlying structures

[2] Other XAIP user studies use one domain (Chakraborti et al. 2019b; Chakraborti and Kambhampati 2019; Sreedharan et al. 2019a, 2020; Lindsay et al. 2020; Das, Banerjee, and Chernova 2021), two domains (Sreedharan et al. 2019b; Sreedharan, Srivastava, and Kambhampati 2020) or four domains (Krarup et al. 2021).

(state space size, number of MUGS), the transition between these classes of tasks is rapid: the ideal task complexity lies on a *knife edge* between too easy vs. too hard. Our OSP task design balances that knife edge as follows.

We keep the plan properties simple, so that they are easy to understand and remember for laypersons, and it is easy to see whether or not a given plan property is satisfied by the current plan. On the other hand, MUGS of size 2 tend to be easy to identify and remember, so we designed our tasks to mostly feature larger MUGS, incorporating complex dependencies and thus a challenging plan space. Along similar lines, we tried to avoid "bottleneck" plan properties, i. e., plan properties that appear in a large fraction of MUGS.

We fine-tuned the task size and MUGS complexity based on small test-run studies. We settled on the following instances. The Transport instance has 9 locations, 2 trucks and 5 packages. There are 15 plan properties reflecting the delivery of packages, use or non-use of road connections, location visits, and ordering relations between packages. There are 37 MUGS. The Parents Afternoon instance has 6 locations and 4 persons, items and activities. We defined 13 plan properties, reflecting achieved activities and ordering relations between those. The instance has 25 MUGS. The Rovers instance has 1 rover, 4 locations and 10 tasks. We designed 14 plan properties pertaining to task achievement and the order of data uploads. There are 102 MUGS.

5 User Study Design

In application scenarios of iterative planning, users are working to understand conflicts between preferences and thus converge to an acceptable trade-off. In a user study, however, test persons will not have an intrinsic motivation to do so. Hence we give them an objective to pursue, namely additive-reward (utility) maximization. This is canonical as it is easy to understand for layperson users. We assign a fixed utility to each plan property. [3]

We link this objective to payment via a bonus growing with the utility achieved, thus providing a strong incentive to find good plans (some prior work, e. g. (Chakraborti et al. 2019b), has followed similar schemes). The basic compensation for participating in the user study is $5£$, the maximum achievable bonus payment is $2.50£$.

To evaluate the effect of Eif20's explanation facility, we divided test persons randomly into groups with vs. without that facility, i. e., with vs. without the option to ask questions. We refer to these two groups as Q+ and Q− respectively.

We ran Transport first, then Parents Afternoon, then Rovers. To maximize familiarity of test persons with the XPP tool and iterative planning, we re-invited test persons to also address the remaining domains. We waited with domain i until the user study on domain $i-1$ was completed, so as to maximize the number of re-invited test persons. We fixed each person's assignment to the Q−/Q+ group across all domains. This serves to obtain consistent streams of test persons becoming increasingly familiar with either of the two tool variants. Also, distributing re-invited users across both variants would have resulted in too many different subgroups for a meaningful analysis.

We used the test person recruitment facilities of Prolific (Palan and Schitter 2018). We applied several filters on test persons to obtain meaningful results. First, we required fluency in English and that at least 50% of each test person's previous submissions in Prolific must have been accepted by the respective study organizers. Second, we filtered out test persons who did not meaningfully process the user study. Finally, in the Q+ group, we filtered out those test persons who did not actually use the explanation facility, i. e., who did not ask any questions. [4]

For each domain, we kept running the study until we had 50 test-person runs for each of Q+ and Q−, to a total of $N = 100$. We ended up with 87 (108) test persons in the Q− (Q+) group. About 75% of the test persons participated in two domains, and 50% of the test persons participated in all three domains.

In addition to user performance measurements, we included a questionnaire for subjective measures, asking users to rate task difficulty, their satisfaction with the achieved utility, and the helpfulness of the explanations, on a Likert scale from 1 to 7. We also included free-text questions, targeted at qualitatively assessing the presentation and usefulness of explanations in the proposed setting.

Each experiment, i. e. each test-person run addressing the OSP task from one of our domains, proceeded according to the following workflow: a textual domain and tool description; familiarization with the tool through an introductory instance; planning for evaluation instance; questionnaire.

6 User Study Results

The main focus of our evaluation is the impact of Eif20's explanations on performance, in terms of utility achieved over time. We also give a summary of the questionnaire results. In what follows, we use the *Student's t-test* to evaluate statistical significance of the difference between means, and the *Wilcoxon rank-sum test* for the difference between medians.

The iterative planning process on the evaluation instance took up to 30 minutes (the rest of the time being spent on the other parts of the test-person workflow). Figure 2 shows utility as a function of this processing time.

In all three domains, the mean utility for Q+ is higher than that for Q− across the entire timeline, indicating that Q+ indeed yields a performance advantage over Q−. In Parent's Afternoon, this advantage is statistically significant along the entire timeline of the experiment. In Transport, both groups initially make similar progress, but the Q− utility growth slows down earlier on while Q+ users are still gaining deeper insights and hence better trade-offs. Accordingly, the advantage of Q+ over Q− is statistically significant for $t \geq 15$min. In Rovers, the timeline effect is the other way

[3] Fixed utility is a standard form of oversubscription planning, which could be solved optimally using known algorithms (e. g. (Smith 2004; Domshlak and Mirkis 2015; Katz et al. 2019)). Nevertheless, this setup is meaningful for evaluating Eif20's explanation approach, as test persons in our study will need to understand the dependencies between plan properties to perform well.

[4] Among test persons who used the tool for the first time, the dropout rate was 25%. Among the re-invited users, there were no.

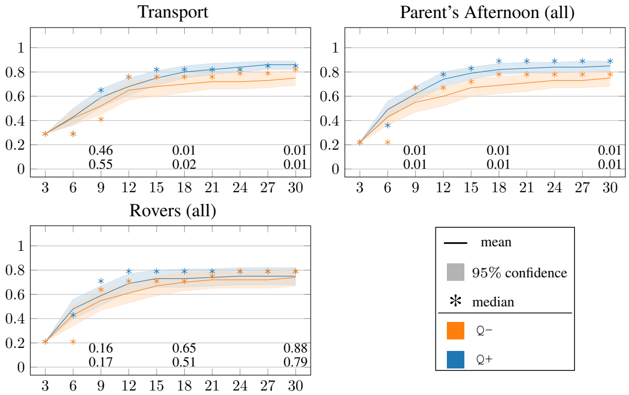


Figure 2: Performance over processing time: x-axis time in min; y-axis maximal achieved plan utility until that time. Numbers shown below the curve are p-values, i. e., the likelihood of the null hypothesis, for mean (top) and median (bottom) after 9/18/30 min processing time.

around, with Q+ users initially making quicker progress but Q− users catching up eventually. Across most of the timeline though, the two curves are closer to each other than in the other two domains, and the advantage of Q+ over Q− is not statistically significant.

Investigating Rovers more deeply, it turns out that tool experience is a major factor here, more than in the other domains, presumably due to the more complex structure (resource consumption *and* time windows) and the larger number of MUGS. Figure 3 assesses this effect in terms of distinguishing between new vs. re-invited users.

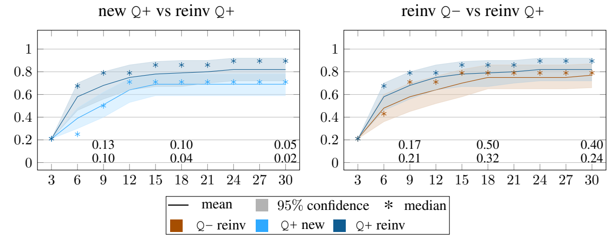


Figure 3: Performance over processing time in Rovers: (left) Q+ new vs. re-invited; (right) re-invited Q+ vs. Q−.

As Figure 3 (left) shows, in Rovers re-invited test persons perform significantly better than new ones a lot of the time. This indicates that tool expertise is important in this domain. Figure 3 (right) evaluates Q+ vs. Q− for re-invited users only. The advantage of Q+ increases clearly compared to the total set of users underlying the data in Figure 2.

Given the smaller sets of test persons in both evaluations in Figure 3 – and the substantial differences between individual crowd workers – the variance is quite high (compare to Figure 2) so statistical significance is found rarely. Nevertheless, in both evaluations, the differences between means and medians are consistent across the entire timeline.

Figure 4 gives representative data for the questionnaire results. As Figure 4 (top) shows, subjective user satisfaction tends to be higher for Q+ users. For Rovers, this is again more pronounced when considering reinvited users only. In Figure 4 (bottom), we see that the subjective helpfulness of explanations is rated 6 out of 7 on average for all domains. The distribution of answers is very similar across domains, indicating that the explanations were found similarly useful in each (despite the differences discussed above).

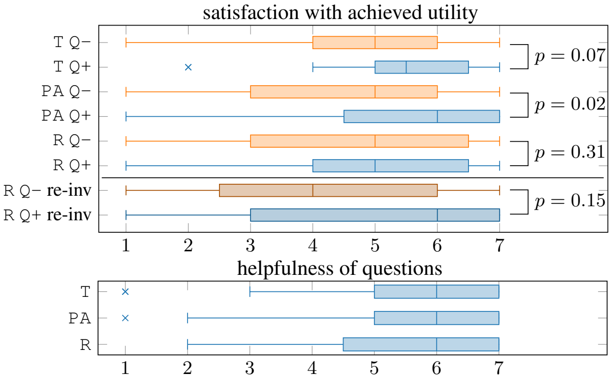


Figure 4: Questionnaire results. Abbreviations: Transport T, Parent's Afternoon PA, Rovers R.

In the free-text questions, study participants were asked to (a) describe their problem solving strategy (Q+ and Q−); (b) criticize the explanation facility (Q+); (c) suggest explanations they would like to have (Q−). The answers were quite diverse, but we identified some trends. A fairly common strategy (a) in both groups (19% of answers) was to start with high-utility properties. The criticisms (b) were extremely diverse, with no major shared themes. In (c), a common theme (16%) were plan-property dependencies, indicating that test persons find this form of explanation natural. Interestingly, another common (c) theme (14%) pertained to planning-model explanations. Users asked whether a package can "be left somewhere that another truck could pick it up", whether a truck can "go back the same way it came", or whether a guitar can be left at the "music lesson".

7 Conclusion

Eif20 introduced a framework for the explanation of conflicts in OSP tasks, and showed how to compute the required information reasonably effectively. An evaluation of whether this form of explanation is useful for users was missing so far. Our work fills this gap. We contribute a web-based platform for iterative planning, and we contribute a large crowd-worker user study. We find that Eif20's explanations tend to enable users to find better trade-offs, allowing us to conclude that the explanations can be useful.

Future work pertains to further developments of Eif20's approach, such as visualization of conflicts, and supporting deeper why questions (why do goals conflict?). Beyond Eif20's approach, our analysis of free-text replies points to a direction that may yet be underdeveloped in XAIP, namely answering questions about the planning task semantics.

Acknowledgments

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-18-1-0245, and by the German Research Foundation (DFG) under grant 389792660 as part of TRR 248 (see https://perspicuous-computing.science).

References

Baier, J. A.; and McIlraith, S. A. 2006. Planning with first-order temporally extended goals using heuristic search. In *Proc. AAAI*, 788–795.

Behnke, G.; Schiller, M. R. G.; Kraus, M.; Bercher, P.; Schmautz, M.; Dorna, M.; Dambier, M.; Minker, W.; Glimm, B.; and Biundo, S. 2019. Alice in DIY wonderland or: Instructing novice users on how to use tools in DIY projects. *AI Communications*, 32(1): 31–57.

Chakraborti, T.; and Kambhampati, S. 2019. (When) can AI bots lie? In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, 53–59.

Chakraborti, T.; Kulkarni, A.; Sreedharan, S.; Smith, D. E.; and Kambhampati, S. 2019a. Explicability? Legibility? Predictability? Transparency? Privacy? Security? The Emerging Landscape of Interpretable Agent Behavior. In *Proc. ICAPS*, 86–96.

Chakraborti, T.; Sreedharan, S.; Grover, S.; and Kambhampati, S. 2019b. Plan Explanations as Model Reconciliation. In *Proceedings of the 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI'19)*, 258–266.

Chakraborti, T.; Sreedharan, S.; Zhang, Y.; and Kambhampati, S. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In *Proc. IJCAI*.

Das, D.; Banerjee, S.; and Chernova, S. 2021. Explainable AI for Robot Failures: Generating Explanations that Improve User Assistance in Fault Recovery. *arXiv preprint arXiv:2101.01625*.

De Giacomo, G.; De Masellis, R.; and Montali, M. 2014. Reasoning on LTL on finite traces: Insensitivity to infiniteness. In *Proc. AAAI*, 1027–1033.

Domshlak, C.; and Mirkis, V. 2015. Deterministic Oversubscription Planning as Heuristic Search: Abstractions and Reformulations. *JAIR*, 52: 97–169.

Edelkamp, S. 2006. On the Compilation of Plan Constraints and Preferences. In *Proc. ICAPS*, 374–377.

Eifler, R.; Cashmore, M.; Hoffmann, J.; Magazzeni, D.; and Steinmetz, M. 2020a. A New Approach to Plan-Space Explanation: Analyzing Plan-Property Dependencies in Oversubscription Planning. In *Proc. AAAI*, 9818–9826.

Eifler, R.; Steinmetz, M.; Torralba, A.; and Hoffmann, J. 2020b. Plan-Space Explanation via Plan-Property Dependencies: Faster Algorithms & More Powerful Properties. In *Proc. IJCAI*, 4091–4097.

Fox, M.; Long, D.; and Magazzeni, D. 2017. Explainable Planning. In *ICAI XAI*.

Göbelbecker, M.; Keller, T.; Eyerich, P.; Brenner, M.; and Nebel, B. 2010. Coming Up With Good Excuses: What to do When no Plan Can be Found. In *Proc. ICAPS*, 81–88.

Katz, M.; Keyder, E.; Winterer, D.; and Pommerening, F. 2019. Oversubscription Planning as Classical Planning with Multiple Cost Functions. In *Proc. ICAPS*, 237–245.

Krarup, B.; Cashmore, M.; Magazzeni, D.; and Miller, T. 2019. Towards Model-Based Contrastive Explanations for Explainable Planning. In *ICAPS XAIP*.

Krarup, B.; Krivic, S.; Magazzeni, D.; Long, D.; Cashmore, M.; and Smith, D. E. 2021. Contrastive explanations of plans through model restrictions. *Journal of Artificial Intelligence Research*, 72: 533–612.

Lindsay, A.; Craenen, B.; Dalzel-Job, S.; Hill, R. L.; and Petrick, R. 2020. Investigating Human Response, Behaviour, and Preference in Joint-Task Interaction. *arXiv preprint arXiv:2011.14016*.

Palan, S.; and Schitter, C. 2018. Prolific.ac A subject pool for online experiments. *Journal of Behavioral and Experimental Finance*, 17: 22–27.

Seegebarth, B.; Müller, F.; Schattenberg, B.; and Biundo, S. 2012. Making Hybrid Plans More Clear to Human Users - A Formal Approach for Generating Sound Explanations. In *Proc. ICAPS*.

Smith, D. 2012. Planning as an Iterative Process. In *Proc. AAAI*, 2180–2185.

Smith, D. E. 2004. Choosing Objectives in Over-Subscription Planning. In *Proc. ICAPS*, 393–401.

Sreedharan, S.; Hernandez, A. O.; Mishra, A. P.; and Kambhampati, S. 2019a. Model-Free Model Reconciliation. In *Proc. IJCAI*, 587–594.

Sreedharan, S.; Kulkarni, A.; Chakraborti, T.; Smith, D. E.; and Kambhampati, S. 2020. A Bayesian Account of Measures of Interpretability in Human-AI Interaction. *arXiv preprint arXiv:2011.10920*.

Sreedharan, S.; Srivastava, S.; and Kambhampati, S. 2020. TLdR: Policy Summarization for Factored SSP Problems Using Temporal Abstractions. In *Proc. ICAPS*, 272–280.

Sreedharan, S.; Srivastava, S.; Smith, D.; and Kambhampati, S. 2019b. Why Cant You Do That HAL? Explaining Unsolvability of Planning Tasks. In *Proc. IJCAI*, 1422–1430.