Explainability in Multi-Agent Path/Motion Planning: User-study-driven Taxonomy and Requirements

Martim Brandao
King's College London
martim.brandao@kcl.ac.uk

Masoumeh Mansouri
University of Birmingham
m.mansouri@bham.ac.uk

Areeb Mohammed
King's College London
areeb.mohammed@kcl.ac.uk

Paul Luff
King's College London
paul.luff@kcl.ac.uk

Amanda Coles
King's College London
amanda.coles@kcl.ac.uk

ABSTRACT

Multi-Agent Path Finding (MAPF) and Multi-Robot Motion Planning (MRMP) are complex problems to solve, analyze and build algorithms for. Automatically-generated explanations of algorithm output, by improving human understanding of the underlying problems and algorithms, could thus lead to better user experience, developer knowledge, and MAPF/MRMP algorithm designs. Explanations are contextual, however, and thus developers need a good understanding of the questions that can be asked about algorithm output, the kinds of explanations that exist, and the potential users and uses of explanations in MAPF/MRMP applications. In this paper we provide a first step towards establishing a taxonomy of explanations, and a list of requirements for the development of explainable MAPF/MRMP planners. We use interviews and a questionnaire with expert developers and industry practitioners to identify the kinds of questions, explanations, users, uses, and requirements of explanations that should be considered in the design of such explainable planners. Our insights cover a diverse set of applications: warehouse automation, computer games, and mining.

KEYWORDS

multi-agent path finding; multi-robot motion planning; explainable planning; explainable AI

ACM Reference Format:
Martim Brandao, Masoumeh Mansouri, Areeb Mohammed, Paul Luff, and Amanda Coles. 2022. Explainability in Multi-Agent Path/Motion Planning: User-study-driven Taxonomy and Requirements . In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 9 pages.

1 INTRODUCTION

Multi-Agent Path Finding (MAPF) and Multi-Robot Motion Planning (MRMP) algorithms have the goal of computing concurrent trajectories for multiple agents/robots that will not lead to any collisions. For the purposes of this paper, we assume MAPF to be the discrete-trajectory-space case, and MRMP the continuous dynamics-constrained case. These algorithms have applications over multiple industries—from logistics to computer games. Due to interactions between agents, and the sizes of the problems in real-world applications, these problems are extremely complex and high-dimensional. This complexity makes MAPF/MRMP plans overwhelming for humans to interpret, evaluate, or improve—and they depend on multiple factors such as problem properties or choice of algorithm and its parameters. One potential way of alleviating these issues is the use of explanation-generation algorithms [3, 10, 21]. These methods could provide interpretable explanations that help users, developers or other stakeholders improve their understanding of MAPF/MRMP problems and algorithms. This increased understanding could in turn contribute to better algorithms and better human-algorithm interactions.

While several explanation-generation methods have been developed for machine-learning classification [3, 24, 26], AI planning [11, 13] and single-agent path planning [7], only limited explainability work has been done in the context of MAPF/MRMP. Research has shown that explanations depend greatly on the application, the algorithm, the users, and the users' interests [5, 16, 21]. Therefore, in order to build useful explainable MAPF/MRMP algorithms, we need to understand what kinds questions can be asked about the output of these algorithms, and what kinds of explanations can be given to answer those questions depending on the context.

In this paper we report on a user study targeted at unpacking the concept of explanation in MAPF/MRMP problems—in particular, we conducted interviews and a questionnaire study with experts in these algorithms in order to identify the kinds of *questions* that can be asked about MAPF/MRMP algorithm output, the kinds of *explanations* that exist to answer such questions, the potential *uses and users* of explanations, and important *requirements and design considerations* for building explainable methods in this space. We chose experts as our target population due to their expertise in developing, using and debugging such algorithms—and thus their knowledge of typical failures, complexities and usage issues that explainability could address. We gathered participants from both academia and industry so as to obtain insights related to requirements and real-world uses and users of explanations.

This paper can be used as a starting point for developers wishing to design real-world explanation-generation methods for MAPF/MRMP. Our contributions are both a taxonomy of explanations, a characterization of uses and users of explanations in multiple applications (warehouse automation, mining and construction, computer games), and a set of requirements and considerations.

2 RELATED WORK

Various methods, studies, and notions of explainability have been put forward in the context of AI algorithms [1, 3, 10, 21].

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online.

Most relevant to this paper is the work on eXplainable AI Planning (XAIP) [10, 11, 13], single-agent path planning [7] and single-agent motion planning explanations [5]. The kinds of explanations provided by these algorithms are not guaranteed to be applicable or useful in the MAPF/MRMP context, however, since explanations are known to be contextual [5, 6, 21]. This contextual nature of explanations has motivated recent user studies eliciting, categorizing, and qualifying notions of explanation in single-agent motion planning [5]. The methodology of Brandao et al. [5] uses open-text and example-driven questionnaires with expert users, to elicit design considerations for explainable single-agent planners. In this paper we use a similar methodology, extended with initial interviews and a questionnaire-based refinement stage for comprehensive elicitation, to characterize explanations in the context of *multi-agent* planning systems—both discrete (MAPF) and continuous (MRMP). Another related work is that of Almagor [2], that proposes a system to generate visual explanations for why a MAPF plan is free of collisions. Explanations of feasibility are generated also for the continuous MRMP case in the recent work of Kottinger et al. [18]. Preliminary work in explainable MAPF has also suggested there could be a need to obtain explanations of infeasibility and sub-optimality ("why is this plan infeasible/sub-optimal?"), as well as "why an agent is waiting too long at a location" [4]. In this paper we include these as potential kinds of questions about plan/planner behavior, while at the same time identifying a large number of (20+) overlooked explanation problems in MAPF/MRMP contexts.

3 METHODOLOGY

3.1 Overview

As described in Section 1, the goal of conducting the present user study was to characterize the notion of explanation in MAPF/MRMP, and gather requirements and design considerations for the development of explainable systems. We were interested in answering the following research questions. **RQ1:** What kinds of questions and explanations can be made about MAPF/MRMP output? **RQ2:** What are potential uses (and users) of MAPF/MRMP explanations? **RQ3:** What requirements and design considerations should developers have in mind when developing explainable MAPF/MRMP systems?

We used the questions above to elaborate a set of interviews and an online questionnaire to gather insights from experts in MAPF/MRMP systems. The interviews served to get preliminary answers to RQ1-3 through long discussions, as well as to sketch prototype explainable planners. The questionnaires then served to refine those preliminary answers through: 1) direct questions about RQ1-3; 2) presentation of the preliminary (interview-based) taxonomy for feedback; and 3) example-driven elicitation methods, i.e. by asking feedback about specific explanations generated by the prototype methods. This two-stage methodology allowed to both explore various themes deeply (in interviews) and to collect feedback on concrete taxonomies and explanation methods (in the questionnaire)—leading to comprehensive and refined answers to the research questions through various elicitation methods.

3.2 Interviews

We conducted interviews with four experts in MAPF/MRMP, in leading roles (three Industry Research Scientists/Leads and one Academic Professor with strong industry links), and who use these algorithms on a daily basis. We selected the interviewees through purposeful sampling [22], carefully identifying experts that cover experience in a diverse set of real-world applications and planning approaches. Taken together the four interviewees' work covers applications in warehouse automation, open pit mining, construction, and computer games; and their technical experience covers different planning approaches, from lifelong and large-scale MAPF, to unsafe (collision-prone) planning with online conflict resolution, to continuous motion planning (see Table 1).

Each interview was semi-structured, roughly one hour long, and conducted by one interviewer. The interviewees agreed to being recorded and transcribed, with recordings being subsequently deleted and transcriptions anonymized. The interviews started by a brief introduction of the purpose of our study, and then asked for a description of how planning algorithms are used in the interviewee's application domains, what the interviewee's role is, how the team is structured, and which algorithms are used. Then, we used multiple strategies to elicitate kinds of questions and explanations for MAPF/MRMP behavior (RQ1). We asked if the interviewees could think of situations in which someone wants to know the reasons behind MAPF/MRMP behavior, we asked for opinion about what kind of questions explainable planners should be able to answer, and we asked for typical failure modes, unexpected outputs, and reasons behind such behavior. Towards answering RQ2, we asked who in the management/development/use chain is more likely to ask which kinds of questions, we asked for pain points in development and whether/how explainability could tackle them, and possible purposes of explainable planners. Requirements and design considerations were raised throughout the interviews (RQ3).

We then performed deductive and inductive analyses of the transcriptions. We used deductive analysis to identify statements related to: how planning is used, kinds of why questions that can be asked, kinds of explanations, uses/purposes of explanations, users of explanations. These findings are described in Section 4. The inductive analysis was made by identifying interesting statements related to issues and considerations in the design of explainable planners throughout the interviews, and then grouping the statements by themes. We describe these findings in Section 5.

3.3 Prototype Explanation Methods

In order to provide questionnaire participants with example explanations, and thus trigger concrete insights [5], we developed three simplistic explanation-generation methods. These methods were developed after analyzing the interviews, and were inspired by the kinds of questions and explanations raised by the interviewees.

The **first explanation method** was a metric-based method targeted at optimal planners which, given a question "why does agent X take path Y instead of Z?", solves a new MAPF problem that has an extra constraint on the agent of interest (forcing it to take path Z). We used CBS [27] to solve this problem. The method then generated an explanation based on the metric of the new solution and that of the original problem. The explanation was either "[agent X does not take path Z] because the metric would be higher" or "because the metric would be the same (and by chance the planner obtained path Y)" depending on the value of the metric.

Table 1: Interview participants

ID	Application areas	Methods
P1	Warehouse automation	Sub-optimal and lifelong MAPF
P2	Warehouse automation	Optimal, sub-optimal and lifelong MAPF
P3	Mining, construction, transportation	Sub-optimal unsafe MRMP with online conflict resolution
P4	Computer games	Sub-optimal unsafe MAPF with online conflict resolution

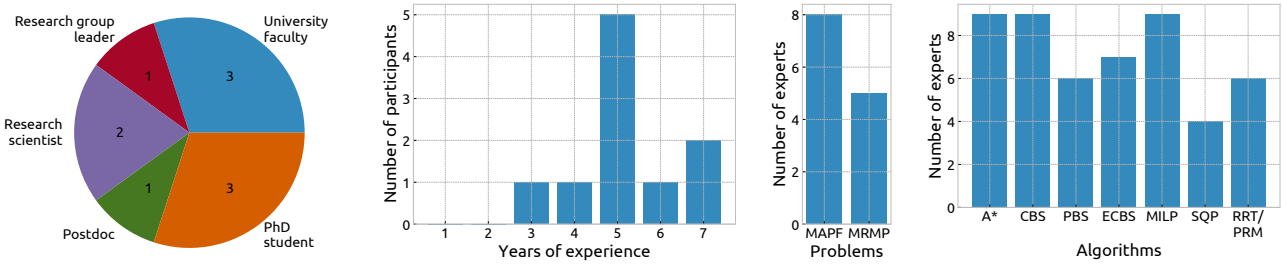


Figure 1: Questionnaire participants' occupation, years of experience, problem expertise, and algorithm expertise.

The **second explanation method** was based on agent-priorities. It assumed a setting where each agent's task (cost) is multiplied by a certain weight value before computing the overall metric—in order to induce priorities. When the question is "why does agent X take path Y instead of Z?", the method checks whether it would be possible to obtain path Z when a different weight is given to that agent. To do this, the method applies binary search on the weight value, solving a new MAPF problem at each iteration using CBS. This binary search finds the lowest possible weight that leads to the expected path (if it exists). When the explanation is applicable, it is of the form "[agent X does not take path Z] because agent X has priority <= *A*. If priority was > *A* then plan Z would be optimal."

The **third explanation method** was the single-agent map-based method of [7]. The method finds map changes (i.e. assignments of each cell to free-space or obstacle) that lead the expected path to become optimal. The method is simplistic because it is only applicable to scenarios where the paths of the agent of interest do not interfere with other agents. We used it because map-based explanations were suggested as being important during the interview stage, and we used it only in scenarios where it was applicable. These explanations are of the form "[agent X does not take path Z] because for that to be optimal there would need to be an obstacle at location L".

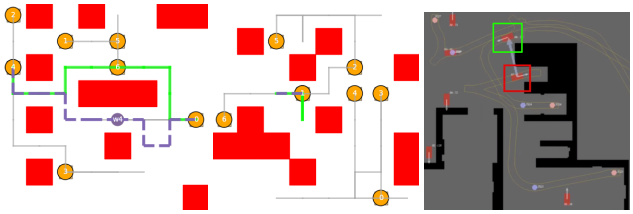
3.4 Questionnaire

After interviews were conducted and analyzed, we designed a questionnaire also targeted at experts in MAPF/MRMP. Participants were considered experts if they had authored academic papers on MAPF/MRMP, or had 1 or more years of experience in working with these algorithms. Participants were recruited using a snowballing strategy—we reached out to MAPF/MRMP experts in our network and in the MAPF community, who also disseminated to their own networks. We stopped recruiting more participants once we reached saturation (i.e. when no new kinds of questions or explanations were being produced by questionnaire answers) [14]. A total of 10 experts filled in the questionnaire. Questionnaire participant information is shown in Figure 1. The figure shows that participants' expertise was high, diverse, and balanced across type of problem and algorithm (note that experts in MRMP were also frequently experts in MAPF, thus leading to a higher number of experts in MAPF).

The questionnaire first asked participants about their experience with MAPF/MRMP in terms of years of experience, classes of problems (MAPF or MRMP), algorithms, typical assumptions (e.g. presence of obstacles, weighted graphs, weighted agents, optimality), and applications. After that, the participants were shown the example problems (a), (b) and (c) in Figure 2 and were asked for their own explanation of planner behavior. On the questionnaire the problems were animated (gray paths indicate other agents' paths). After providing their own explanations in open text, participants were shown the following hypothetical explanations:

(1) "because the cost would be higher/equal" (a,b,c)—generated by the first explanation method in Section 3.3;
(2) "because all agents have equal priority. If agent Y had weight >[number] then the optimal path would satisfy X" (a,b)—generated by the second explanation method in Section 3.3;
(3) "because that would require an obstacle at location Y" (a)—generated by the third explanation method in Section 3.3;
(4) "because the planner is sub-optimal" (c).

We used this strategy, of providing hypothetical explanations in concrete example problems (called "probes"), since it has been shown to be effective at eliciting insightful explanation examples and design considerations [5]. We then asked participants for qualitative feedback on the hypothetical explanations in the form of open-text criticism and suggestions for improvement. Finally, the users were shown a preliminary taxonomy of explanations (obtained from the interview analysis) and were asked for criticism and suggestions of



(a) (b) (c)

Figure 2: Questionnaire's example problems. a) "why does the agent not pass by w4 instead of taking the green path?" (MAPF). b) "why does it not take the blue instead of green path?" (MAPF). c) "why does the red-marked agent wait for the green-marked agent?" (MRMP). c) courtesy of [12].

question/explanation/category additions (RQ1). The questionnaire ended by asking participants what they think are the potential uses of automated explanations in the applications they are familiar with (RQ2). We used open-text criticism throughout the questionnaire to gather requirements and design considerations (RQ3).

We analyzed the responses of the questionnaire in the same manner as for the interviews (deductive analysis for RQ1 and RQ2, inductive for RQ3), and used the analysis to refine the preliminary answers to the research questions obtained from the interviews. In the sections that follow we report on the findings obtained from the full analysis, and where appropriate we highlight insights that were obtained only from the questionnaire (e.g. taxonomy extensions).

4 FINDINGS: CONTEXT AND TAXONOMY OF EXPLANATIONS (RQ1, RQ2)

4.1 Kinds of Why Questions

As mentioned in Section 3, we gathered and organized "why questions" about MAPF/MRMP algorithm output that were mentioned by participants in interview scripts and questionnaire answers. We first extracted such questions from the interview scripts (marked as "I" in Table 2). Interviewees mentioned a large number of potential questions that can be asked about planner output. Some questions related to specific events or properties of the paths: for example why a set of agents take specific paths or make specific waypoint decisions, why a set of agents does NOT make user-expected decisions, or why there is congestion. Other questions related to the metric ("why is the plan optimal?", "why isn't the metric higher?") and others related to the algorithm (why an algorithm, heuristic or parameter performs better than another). Some questions were related to similarities or consistencies between the behavior of an algorithm over multiple problems—why agents consistently made certain waypoint decisions, or why congestion consistently happened at specific locations. The interview-provided ("I") questions were thus organized into categories (Plan-, Metric-, Consistency-, Algorithm- and Design-centered), and shown to questionnaire participants for feedback and improvement suggestions.

Additions suggested by questionnaire participants are marked as "R" in Table 2. Questionnaire participants suggested two "why" questions that had not been considered in the initial list: "why is this plan optimal?" (1 participant) and "why does the algorithm take so long to find a path [in this problem/map]?' (4 participants).

Interview and questionnaire participants also provided non-why questions. Since explanations are typically defined as answers to "why" questions [19–21], we transformed non-why questions into pairs of why questions and explanations. For example, "What should I change about the input to get behavior X" is transformed into question "why did I not get behavior X?" and explanation "because that would require a change C to the input". Other non-why questions provided include: "how should I change the layout in order to reduce congestion?", "which algorithm would be faster to run on this input?", "what is a good metric to quantify my desired behavior X?", "what are the best hyperparamaters?". These concrete examples were translated into congestion and computation-time questions, as well as map-, algorithm- and metric-based explanations (Table 3).

From web searches of academic papers mentioning explainable MAPF/MRMP (see Section 2) we also gathered questions that have been tackled in the literature, which we mark as "L" in Table 2. The table shows that the literature recently focuses on only a small subset of the questions experts and practitioners believe to be useful.

4.2 Kinds of Explanations

We first extracted categories and examples of explanations from interview scripts, leading to the list in Table 3: "I". We identified various kinds of explanations, most of which related to the problem or the algorithm. Problem-based explanations related to the metric, agents' priorities, set of constraints, edge costs and obstacle/target locations, number of agents, etc. Algorithm-based explanations were related to the properties of the planner and heuristic, and planner decisions such as state pruning or planner-added constraints.

This list, together with the categories (but not sub-categories) was called a "preliminary taxonomy" and was shown to questionnaire participants for feedback. The participants provided new examples of explanations both in their answers to the probes (Figure 2 a, b, c) and, at the end of the questionnaire, through feedback to our initial taxonomy. This led to several new examples of explanations, marked with "R" in the table, that had not been provided during the interview stage. New examples were based on agent conflicts, chains of events, the metric, size and density of the map, planner properties, and multiple algorithm-based explanations. Event chains (e.g. "if robot 1 waits, it will affect robot 2, which will in turn affect robot 3") were used by multiple participants, who suggested this should be a (sub-) category of explanations by itself.

Participants suggested the explanations should be further sub-categorized in the taxonomy, since they related to different concepts—especially in the problem and algorithm-based explanations. We used such feedback to further refine the taxonomy as shown in Table 3. Participants asked for sub-categorization in particular for problem- and algorithm-based explanations. This is because problem-based explanations can refer to various causes related to either the problem's provable-properties, the metric, the constraints, the map, the number/priority of agents, or a problem examplar (e.g. "behavior X happened for the same reason it would happen in this example (smaller) problem"). Algorithm-based explanations can also refer to a range of causes, from planner properties (e.g. completeness, statistics of its behavior), specific algorithm-made

Table 2: Kinds of why questions about MAPF/MRMP
(L=Literature on XMAPF; I=Initial examples from interviewees; R=Refinements by questionnaire participants)

Kinds of questions	Example questions	L	I	R
Plan-centered	why is this plan feasible?	*		
	why is this plan infeasible?	*		
	why is there no plan? (why did the planner fail to obtain a plan?)		*	
	why did agents X traverse through locations L?		*	
	why did agents X take paths Y, when I expected them to take path Z?		*	
	why did agents X take paths that are so far off of paths Y?		*	
	why did agents X wait/stop/get-precedence here?		*	
	why did agents X wait for that other agent here?		*	
	why did agents X wait for so long here?	*	*	
	why is agent X not at location P before time T?		*	
	why is agent X on a collision course?		*	
	why is there a deadlock between agents X?		*	
	why is there a congestion in area A?		*	
Metric-centered	why isn't metric M higher?		*	
	why is this plan optimal?		*	
	why is this plan sub-optimal?	*		*
Consistency-centered	why does agent X always get precedence (over multiple problems)?		*	
	why does agent X agent always go through location X (over multiple problems)?		*	
	why is there always congestion in this area (over multiple problems)?		*	
Algorithm-centered	why does heuristic H1 perform better than H2 in this problem/map?		*	
	why does algorithm/variant A1 perform better than A2 in this problem/map?		*	
	why does the algorithm take so long to find a path in this problem/map?			*
Design-centered	why do I need this many agents?		*	

decisions, algorithm parameters, information the (decentralized) algorithm has access to, or software bugs.

4.3 Contexts, Users and Uses of Explanations

4.3.1 Contexts and users. Interviewees provided some insight into how MAPF/MRMP problems are posed, solved, and interacted with in different applications—as well as who the most likely users of explanations are in these contexts.

In warehouse automation, as discussed by P1 and P2 during the interviews, planning problems are large-scale and dynamic MAPF problems. They can have thousands of agents and require online re-planning, therefore leading to the use of sub-optimal algorithms that are fast but obtain good solutions in practice. Interviewees further explained that a lot of developers' work is spent on tuning the cost functions used, so that future behavior better matches path expectations and efficiency goals. Engineers in warehouses may also have to deal with undesired events on the spot (e.g. fix a robot, change the layout) and therefore need to change MAPF constraints on the fly in order to obtain desired behavior (e.g. avoid an area during maintenance). Interviewees with experience in warehouse automation suggested that at-site engineers are one of the potential users of planner explanations, since they could use them in order to understand behavior as it happens (asking plan-centered questions), as well as to know how to change the model or algorithm parameters in real-time in order to obtain desired behavior (using problem/algorithm-based explanations) . They also suggested that sometimes algorithm developers have an idea of what paths should look like and so they would be interested in being able to ask "why not path Z instead of Y?" (plan-centered questions) in order to tune the algorithm and objective functions (through algorithm/metric-based explanations). Finally, interviewees and questionnaire respondents in warehouse applications suggested warehouse layout designers as interested parties in explanations, since these could be used to identify layout changes (e.g. corridor, boxes, or packing station re-arrangements) that lead to desired behavior or increased efficiency (through plan/metric-centered questions and map-based explanations).

In computer games, multi-agent problems can also be large-scale in terms of graph size and agent numbers, and dynamic due to moving players. Computation needs to be very fast and the main objective is visual realism to players. Due to this, fast sub-optimal algorithms are used, often without safety guarantees—for example through single-agent A* planning that ignores other agents, followed by online collision avoidance strategies. Game designers will play the game, look for undesired behavior (e.g. unintuitive paths, areas of congestion), and either solve the issues themselves when they can, or flag them for developers to tune graph weights, map geometries, collision-avoidance strategies, etc. These designers were therefore suggested by P4 as one of the most likely users of explanations in a game development context. They could thus use

Table 3: Kinds of explanations for MAPF/MRMP
(I=Initial examples from interviewees; R=Refinements by questionnaire participants)

Category	Sub-category	Example explanations	I	R
Plan-based	Agent interference	because that would delay agents X		*
		because that would create conflicts with agents X		*
		because of a deadlock between agents X	*	
	Event chains	because that would affect agent X, which would affect agent Y		*
		because of event X at time T that propagated	*	
	Landmarks	because a large number of agents have to go through area A	*	
Problem-based	Problem properties	because the environment is not well-formed/well-structured	*	
	Metric	because that would be worse according to the metric	*	*
		because that would require change X to the metric		*
		because the metric would be the same		*
	Constraints	because agents X have higher priority (imposed by the problem)	*	*
		because that would require an extra constraint/precondition X	*	*
		because of the kinodynamic constraints of agents X		*
	Map	because of the costs assigned to edges/regions X	*	
		because object O is at location X	*	
		because of obstacles X	*	*
		because of the size of the map		*
		because of obstacle density (at location X)		*
		because of a choke point at location X	*	*
		because that would require a change to the map	*	*
	Agents	because there are too many/few agents	*	*
		because that would require X more/less agents	*	*
		because that would require agents to have priorities X	*	
	Example	for the same reason as in this smaller problem	*	
Algorithm-based	Planner properties	because the planner does not provide safety guarantees	*	
		because the planner is incomplete	*	
		because the planner is sub-optimal	*	
		because heuristic H is inadmissible	*	
		because the planner is better on small/large maps		*
		because the planner cannot handle a large number of agents well		*
		because the planner explores movement in direction X first		*
		because heuristic prefers moving agents close to their destination first		*
	Planner decisions	because of an incorrect state expansion/prune	*	
		because of planner-added constraint X (e.g. PBS, MAPF/C, k-delay MAPF)	*	*
		because of the priority ordering of the agents (imposed by planner)	*	*
	Planner parameters	because algorithm A1 was used instead of A2		*
		because heuristic H1 was used instead of H2		*
		because hyperparameter X was equal/lower/higher than Y		*
	Information	because of incorrect information (decentralized algorithms)		*
		because of insufficient information (decentralized algorithms)		*
		because the algorithm was not trained on similar examples (learning algorithms)		*
	Bug	because of a bug in function/class/file/line X	*	*
Execution-based	Execution gap	because agents X stopped due to technical failure	*	
		because agents X are moving slower/faster than expected		*

explanations to better understand the reason for the undesired behavior (e.g. asking plan-centered questions "why is there no plan?", "why did agent X wait?", and map-based explanations such as mesh connections or costs of different terrains) and to understand what to change about the map or agent capabilities in order to improve the game (e.g. connect a mesh, change edge costs).

In open-pit mining and some transportation domains planning problems need to be solved with continuous MRMP methods that allow kinodynamic constraints on robots. Environments are

more unstructured than in other applications, and the results of executing plans subject to higher uncertainty—therefore typically requiring the use of distributed planning and execution. Formal methods are often applied in order to still be able to provide certain guarantees demanded by industry. P3 suggested management could benefit from explanations as a way of intuitively showing why metrics of interest are being optimized as much as they can (through metric-based explanations).

In all applications, the majority of interviewees and questionnaire respondents suggested that algorithm developers and researchers could use explanations in order to inspect the models and algorithms, to help identify bugs, or to understand why some heuristics or algorithm variants are better in some situations. Thus, respondents identified developers and researchers themselves as interested *users* of algorithm explanations.

4.3.2 Purposes. Overall, interviewees suggested many purposes for explanations, aligned with the user insights above. Explanations could serve to: **Understand planner behavior.** Explanations could be used to understand the reasons behind (potentially undesired or unexpected) behavior—for a particular agent or set of agents. There is an expectation that they could help reduce the complexity of the problem related to a very large number of agents or complicated map. **Understanding limits of model and planner capabilities.** Interviewees mentioned that explanations could, by improving user understanding of the planner, better align expectations with real capabilities. In the context of computer games, P4 mentioned this could help game designers understand what can and can't be done with pathfinding (e.g. where the desired behavior can be achieved by tuning weights, or there is a need to create new capabilities). **Align expected and executed plans.** Explanations could be used to help engineers and developers identify necessary changes to the model or algorithm, so as to obtain plans that are in line with stakeholders' expectations. In other words, interviewees expressed their belief that explanations can be used to help with problem specification and algorithm improvement. This was also related to references to the usefulness of explanations for "cost tuning". **Design optimization.** This purpose was brought up especially in the context of warehouse automation, where warehouse layout plays a significant role in efficiency and robustness of plans. The idea is that warehouse layout designers could use explanations as intuitive interfaces to obtain suggestions for improvements such as new packing station locations. **Communicating with management.** As mentioned above, explanations could be used to communicate intuitively why a metric that management cares about is being appropriately optimized. **To improve quality and predictability of interaction with humans.** This purpose was mentioned by interviewee P3, who had experience in domains such as transportation and mining, where humans may interact and share space with robots, as well as partial automation domains. According to the interviewee, explanations (and simply interpretable/self-explainable behavior) could be used to align humans' expectations with planner behavior. **Debugging.** Both groups mentioned the possibility of using explanations as an aid in identifying software bugs.

4.3.3 Reception. Perhaps because of their frustrations in interpreting MAPF/MRMP planner outputs, participants were optimistic about the possibility of using explanations to improve understanding and tuning of the problem and algorithms: they said explanations "will be extremely helpful when developing algorithms and comparing them (or just debugging)", they will help "identifying the short-comings" of algorithms, and provide "leavers (...) to be able to alter the behavior". Participants also believed that explanations can help understand "how the problem constraints or objective function affect the resulting plan". They found explanations to be "interesting" because they can often provide "a way to achieve the behavior you wanted". However, one interviewee was skeptical regarding the importance of explanations for humans physically interacting with robots. P3 argued that "explicability"—the degree to which an agent's behavior matches user expectations, thus avoiding the need of an explanation—is potentially more urgent in this context. Explicable behavior is particularly urgent, P3 argued, on environments where humans work alongside robots, or where robots are partially automated. Here explanations may serve as a way to increase understanding but, ultimately, they can be avoided if robots are able to move in ways that do not trigger a desire of explanation by users. In practice, however, complex agent behavior may only become explicable after long periods of interaction time, and thus explanations can still have the role of speeding up this process.

5 FINDINGS: REQUIREMENTS (RQ3)

5.1 Large-scale and Replanning Methods

Participants agreed that real-world large-scale problems with "hundreds of robots" are extremely hard to understand: they mentioned "immense complexity", "confusing objectives", and how even as experts they sometimes do not have "good intuition for what a good solution looks like" because the causal relationship is "not visual anymore, it is cost-based". According to them, for explainable methods to be practical and useful they need to be applicable to large-scale methods, lifelong/re-planning methods, and they should focus on reducing complexity (i.e. particularly agent-interference, event chains, map, and metric-based explanations).

5.2 Sub-optimal and Incomplete Methods

One of the common themes raised by participants was that most real-world problems are solved with incomplete and sub-optimal algorithms—and thus most often explanations for behavior will be algorithm-based. As we have seen, such kind of explanation can be related to the choice or properties of the heuristics, algorithm parameter values, states that were (incorrectly) pruned, agent priorities that were imposed, etc. Most of these explanations may also be difficult to compute, since they may involve searching over the space of algorithm parameter values and heuristic choices. For explainable MAPF/MRMP algorithms to be useful in real-world applications, they should thus be able to explain the behavior of sub-optimal and incomplete planners—and tackle the inherent complexity of algorithm-based explanations.

5.3 Core Events

Multiple participants noted that undesired behavior, such as congestion or collision, is often a result of a core event that triggers a cascade of events that lead to the behavior. For example, a small deviation from planned motion in online re-planning settings can

lead to delays or de-tours in multiple agents. There is typically "a core event, or a core robot that causes a problem, or a core decision that sets off this cascade of bad behavior", which is exemplified by "a robot blocking a path, or a robot going against the flow of traffic". Even under the assumption of perfect execution, sub-optimal plans may lead an agent to wait at a certain location, or take a path that blocks passage to other agents for a long time, and thus lead to similar results. One interviewee used a car-traffic analogy: "sometimes the explanation simply has to do with the fact there's the rubber banding effect, or there was some tiny incident", or "the road has a small deformation in it" which created slowdowns "and this created other slowdowns and then this propagated". An explanation could thus point to these core events, or suggest what to change about the model or algorithm that would prevent the event.

5.4 Levels of Explanation

Another common theme was that of explanations being contextual. Not only do the kinds of questions asked depend on the user and application (Section 4.3), but also the content of resulting explanations may vary depending on multiple factors. An explanation may be offered at multiple levels of detail and use more or less technical language, depending on user expertise and interests. A high-level explanation could refer only to robots that "have to pass", but lower-level explanations should refer to how states "were expanded, and the heuristic values for example", or should "render the criteria for determining precedence clear". Some users such as managers could be more interested on high-level metric-based explanations, but engineers would need lower-level explanations that provide "as many knobs as possible to be able to alter the behavior". Additionally, multiple explanations may exist for certain behavior—e.g. the reason for a certain agent's path being the way it is may be related to properties of the map, the metric, or algorithm parameters or properties. So there will often be "many alternative answers to [a] question". And "answering why is not easy, because you would have to compute and give many alternative answers". Therefore, not only a choice of explanation, but a choice of a set of explanations will have to be made depending on who the user is and what they are interested in. Designing the right explanation system will thus be a complex human-computer interaction (HCI) design problem: involving multiple iterations of 1) technical development of explanation-generation methods and user interfaces, and 2) the evaluation of these through quantitative metrics [9, 16] and qualitative methods of user-experience from HCI and social science research [15, 17, 23].

5.5 Interactions Between Multiple Agents

As criticism to the explanation example probes used in the questionnaire (Figure 2), multiple participants mentioned the need to explain the interaction between many (or the most relevant) agents in order to be useful—instead of focusing only on pair-wise interactions. They mentioned that our explanation (2a) was "all about agent 4" but that it should also explain the effects on other agents; it should focus on "critical agents" and provide a list of the affected agents. Therefore, explanation-generation methods need to be able to identify core sets of agents influencing behavior of interest: these could be sets of agents that are always involved in satisfying a task, or conflict with each other in all optimal (e.g. minimal flowtime) solutions to a problem, or agents that interact with each other but do not influence the costs of other agents, etc.

6 CONCLUSION AND DISCUSSION

In this paper we described an expert/practitioner-based user study we conducted with the goal of characterizing the concept of *explanation* in MAPF and MRMP. The paper makes several key contributions to the field of explainable AI and multi-agent planning—substantive, methodological, and practical. **The substantive contribution** consists of a taxonomy of questions, explanations and purposes that explainable multi-agent planners should be able to consider. Importantly, we identify a large number of currently overlooked forms of explanation: relating to the plan, plan-execution, the problem (e.g. its properties, metric, map), and the algorithm (e.g. its properties, parameters). **The methodological contribution** consists of using a two-stage elicitation approach that 1) obtains preliminary findings from purposefully-sampled interviews, based on which we define prototype explainable methods to generate examples, and 2) uses example-driven elicitation and taxonomy refinement on the second stage for comprehensive results. **The practical contribution** consists of a set of requirements and design considerations that developers should focus on when designing explainable MAPF/MRMP methods. In particular, according to our study, explainable systems should focus on reducing the complexity of large-scale plans, identifying core events that trigger undesired behavior, they should be able to generate explanations at multiple levels of abstraction, and be able to explain the behavior of sub-optimal and incomplete planners. This is in contrast with recent work on explainable MAPF and MRMP, which has focused on explanations of feasibility and on optimal methods.

Our end goal is the development of explainable MAPF/MRMP systems, and this paper presents a first step towards this direction: mapping the kinds of explanations that will be useful and their requirements. While it was not part of the goal of this study to identify specific *methods* for generating explanations, some participants suggested the use of the "nullspace of the optimization" and the use of highly-parallel computing to "spawn simulations under different conditions" in order to understand whether the reason for certain behavior was related to the choice of heuristics, problem parameters, etc. Additionally, explainable MAPF/MRMP methods could potentially build on various AI and path/motion planning work: these could range from plan summarization [25] and eXplainable AI Planning [11] to inverse optimization [7] or design optimization [8]. We believe this paper will provide developers with the concepts, requirements and focus points that are necessary to purse such efforts. Interesting future research directions include user studies with end-users such as at-site warehouse engineers, computer game designers, and lay users; and the adaptation of single-agent path finding [7] and motion planning explanation-generation algorithms [6] into the multi-agent setting.

ACKNOWLEDGMENTS

This work was supported by AFOSR (FA9550-18-1-0245), EPSRC THuMP (EP/R033722/1), and UKRI TAS Hub (EP/V00784X/1).

REFERENCES

[1] Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y Lim, and Mohan Kankanhalli. 2018. Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda. In *ACM Conference on Human Factors in Computing Systems (CHI)*. 1–18.

[2] Shaull Almagor and Morteza Lahijanian. 2020. Explainable Multi Agent Path Finding. In *International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*. 34–42.

[3] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion* 58 (2020), 82–115.

[4] Aysu Bogatarkan. 2021. Flexible and Explainable Solutions for Multi-Agent Path Finding Problems. In *International Conference on Logic Programming (Technical Communications)*.

[5] Martim Brandao, Gerard Canal, Senka Krivic, Paul Luff, and Amanda Coles. 2021. How experts explain motion planner output: a preliminary user-study to inform the design of explainable planners. In *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*.

[6] Martim Brandao, Gerard Canal, Senka Krivic, and Daniele Magazzeni. 2021. Towards providing explanations for robot motion planning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*.

[7] Martim Brandao, Amanda Coles, and Daniele Magazzeni. 2021. Explaining Path Plan Optimality: Fast Explanation Methods for Navigation Meshes Using Full and Incremental Inverse Optimization. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*.

[8] Martim Brandao, Rui Figueiredo, Kazuki Takagi, Alexandre Bernardino, Kenji Hashimoto, and Atsuo Takanishi. 2020. Placing and scheduling many depth sensors for wide coverage and efficient mapping in versatile legged robots. *The International Journal of Robotics Research (IJRR)* 39, 4 (2020), 431–460. https://doi.org/10.1177/0278364919891776

[9] Tathagata Chakraborti, Anagha Kulkarni, Sarath Sreedharan, David E Smith, and Subbarao Kambhampati. 2019. Explicability? legibility? predictability? transparency? privacy? security? the emerging landscape of interpretable agent behavior. In *International Conference on Automated Planning and Scheduling (ICAPS)*, Vol. 29. 86–96.

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

[11] Tathagata Chakraborti, Sarath Sreedharan, Yu Zhang, and Subbarao Kambhampati. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In *International Joint Conference on Artificial Intelligence (IJCAI)*. 156–163.

[12] Paolo Forte, Anna Mannucci, Henrik Andreasson, and Federico Pecora. 2021. Online Task Assignment and Coordination in Multi-Robot Fleets. *IEEE Robotics and Automation Letters (RAL)* 6, 3 (2021), 4584–4591. https://doi.org/10.1109/LRA.2021.3068918

[13] Maria Fox, Derek Long, and Daniele Magazzeni. 2017. Explainable planning. *arXiv preprint arXiv:1709.10256* (2017).

[14] Judith Green and Nicki Thorogood. 2004. *Qualitative methods for health research*. Sage.

[15] Christian Heath and Paul Luff. 2018. The Naturalistic Experiment: Video and Organizational Interaction. *Organizational Research Methods* 21, 2 (2018), 466–488. https://doi.org/10.1177/1094428117747688

[16] Robert R Hoffman, Shane T Mueller, Gary Klein, and Jordan Litman. 2018. Metrics for explainable AI: Challenges and prospects. *arXiv preprint arXiv:1812.04608* (2018).

[17] John F Kelley. 1984. An iterative design methodology for user-friendly natural language office information applications. *ACM Transactions on Information Systems (TOIS)* 2, 1 (1984), 26–41.

[18] Justin Kottinger, Shaull Almagor, and Morteza Lahijanian. 2021. MAPS-X: Explainable Multi-Robot Motion Planning via Segmentation. In *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE.

[19] D Lewis. 1986. Causal Explanation. *Philosophical Papers* (1986), 214–240.

[20] Peter Lipton. 1990. Contrastive explanation. *Royal Institute of Philosophy Supplements* 27 (1990), 247–266.

[21] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267 (2019).

[22] Lawrence A Palinkas, Sarah M Horwitz, Carla A Green, Jennifer P Wisdom, Naihua Duan, and Kimberly Hoagwood. 2015. Purposeful sampling for qualitative data collection and analysis in mixed method implementation research. *Administration and policy in mental health and mental health services research* 42, 5 (2015), 533–544.

[23] Peter G Polson, Clayton Lewis, John Rieman, and Cathleen Wharton. 1992. Cognitive walkthroughs: a method for theory-based evaluation of user interfaces. *International Journal of man-machine studies* 36, 5 (1992), 741–773.

[24] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should i trust you?" Explaining the predictions of any classifier. In *ACM SIGKDD International Conference on Knowledge Discovery and Data mining (KDD)*. 1135–1144.

[25] Stephanie Rosenthal, Sai P Selvaraj, and Manuela M Veloso. 2016. Verbalization: Narration of Autonomous Robot Experience.. In *International Joint Conference on Artificial Intelligence (IJCAI)*, Vol. 16. 862–868.

[26] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206–215.

[27] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R Sturtevant. 2015. Conflict-based search for optimal multi-agent pathfinding. *Artificial Intelligence* 219 (2015), 40–66.