Evaluating contrastive explanations for AI planning with non-experts: a smart home battery scenario

Yiwei Shi, Kevin McAreavey, Weiru Liu
School of Computer Science, Electrical and Electronic Engineering, and Engineering Maths
University of Bristol, Bristol, UK
{yiwei.shi.2019, kevin.mcareavey, weiru.liu}@bristol.ac.uk

***Abstract*—Smart home systems with AI planning functionality have the potential to improve the lives of users. However, there is an emerging expectation that users should understand and trust the decision-making processes of these systems. In this paper, a smart home battery system is developed with a supplementary explanation module that allows non-expert users to intuitively visualise the planning process and to better understand its recommendations. The module relies on a notion of contrastive explanations, related to iterative planning, allowing users to ask contrastive questions based on state- and action-constraints that may or may not be satisfiable. The system is intended for an experimental study where participants interact with the planning system and complete an questionnaire, with the research objective being to evaluate the usefulness of the explanation module.**

***Index Terms*—planning, explainable AI (XAI), contrastive explanations**

I. Introduction

Artificial intelligence (AI) technology is developing at a rapid rate, yet as AI systems become more complex, it is increasingly difficult for stakeholders to understand their decision-making processes [1]–[3]. The field of explainable AI (XAI) seeks to address this issue by improving human-understanding of AI methods [4].

The authors of [5], [6] identify several challenges faced by XAI research. Firstly, there are many subfields of AI and each has a need for explanations, yet much of the current XAI research has focused on machine learning, with other subfields (e.g. AI planning) receiving less attention. Secondly, AI systems exhibit many stakeholders, including non-expert end-users, yet much of the current XAI research has focused on expert stakeholders (e.g. machine learning experts). Thirdly, different stakeholders interact with AI technology in different ways, and in the case of non-expert stakeholders evaluating XAI research depends on an underlying AI system that can sufficiently engage users and thus give rise to a need for explanations. In this paper, we address some aspects of the above challenges by developing a smart home battery system for non-experts that relies on AI planning and offers an explanation module.

Planning is an important subfield of AI. For example, planning can navigate scattered robots to reach a predetermined formation [7] and reduce global vehicle scheduling times [8]. However, fully understanding planning algorithms and their solutions can be extremely difficult for non-experts, if not experts. The field of explainable AI planning (XAIP) is the subfield of XAI as it relates to planning [9]. One example is the work of Eifler et al., who in [10] propose an approach to XAIP inspired by planning as an iterative process. This setting refers to a type of human-in-the-loop planning where features of the planning problem (e.g. goals, preferences) are only partially understood. The basic idea is that these aspects of the planning problem can be refined through an iterative process of altering the problem, (re)planning, and then observing the outputs.

The notion of contrastive explanation is well-established in philosophy and social science [11]. The observation is that when humans seek explanations they do not ask simply *why P?* but instead ask *why P rather than Q?* where Q is some contrastive event known as a foil. In AI planning, a planning problem describes a state-action transition system with an objective (e.g. goal states) such that a solution is an optimal plan that specifies applicable actions to execute in order to achieve the objective. If we specify constraints on valid states or actions then it is possible to limit the space of valid plans such that every solution must satisfy the constraint. In the human-in-the-loop setting for example, such constraints may express user preferences. A natural application of contrastive explanations to the setting of AI planning then is to allow users to ask of the system questions of the form *why did the system recommend this plan rather than one that satisfies constraint C?*. Suitable explanations may be that the constraint is unsatisfiable, that the constraint is satisfiable but leads to higher cost, or that the constraint can indeed be satisfied with equal cost.

A planning system was proposed in [5] for scheduling a smart home battery so as to optimise home electricity costs. In this paper we extend this work by developing an interactive planning system that offers constrastive explanations. There are three main contributions of this paper. First, we design and implement a fully interactive planning system. Second, we propose a notion of contrastive explanations based on state- and action-constraints. Third, we design an explanation module that allows users to request contrastive explanations, having both visual and textual representations. The system includes two variants: in one variant (for the treatment group) the XAI module is included, and in the other variant (for the control group) the XAI module is excluded. The rest of the paper is organized as follows. In the Section 2, we introduce the main concepts used in smart home battery planning.

This work is partially funded by the EPSRC CHAI project (EP/T026820/1).

In Section 3, we give the formal definition of contrastive explanation in planning and describe the architecture of the system. In Section 4, we provide details of the interactive interface and how it is used by two different user groups. In Section 5 we conclude with a short discussion.

II. Preliminaries

In this section we give a very brief introduction of MDP and the main definitions in [5] that are used to design and implement the back-end of a smart home battery planner.

MDP is a tuple (S, A, T, C) where S is a set of states, A is a set of actions, $T : S \times A \to S$ is a transition function, and $C : S \times A \to \mathbb{R}$ is a cost function. A finite-horizon MDP extends the standard MDP definition by including a (decision) horizon $t_{max} \in \mathbb{N}$ with $D = \{1, \ldots, t_{max}\}$ the set of timesteps. A (non-stationary) policy is a function $\pi : S \times D \to A$ where the cumulative cost of π in state $s \in S$ at timestep $t \in \mathbb{N}$ is defined as:

$$V(s, t, \pi) = \begin{cases} C(s, a) & \\ \;+ V(s', t+1, \pi) & \text{if } 1 \leq t \leq t_{\max} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

such that $a = \pi(s, t)$ and $s^{'} = T(s, a)$. A policy π^* is an optimal policy if it minimises $V(s, t, \pi^*)$ for all $s \in S$ and all $t \in D$.

Definition 1. *A battery scheduling problem is a tuple* $(\beta, s_1, \lambda, t_{\max}, U, P_I, P_E)$ *where:*

- $\beta \in \mathbb{R}^{\geq 0}$ *is the (battery) capacity constant*
- $s_1 \in [0, \beta]$ *is the current (battery) level*
- $\lambda \in [0, \beta]$ *is the (dis)charge rate per timestep*
- $t_{\max} \in \mathbb{N}$ *is the horizon with* $D = \{1, 2, \ldots, t_{\max}\}$
- $U : D \to \mathbb{R}$ *the (electricity) consumption forecast*
- $P_I : D \to \mathbb{R}$ *the (electricity) import price forecast*
- $P_E : D \to \mathbb{R}$ *the (electricity) export price forecast*

Definition 2. *Let* $(\beta, s_1, \lambda, t_{\max}, U, P_I, P_E)$ *be a battery scheduling problem. A battery scheduling model is an MDP* $(S, A, T, C, t_{\max}, s_1)$ *where:*

- $S = [0, \beta]$ *is the set of (battery level) states*
- $A = \{-1, 0, 1\}$ *is the set of (battery) actions with 1 the charge action, -1 the discharge action, and 0 the no-op action*
- $T : S \times A \to S$ *is the transition function defined for each* $s \in S$ *and each* $a \in A$*:*

$$T(s, a) = \min\{\beta, max\{0, s + a\lambda\}\} \tag{2}$$

- $C : S \times D \times A \to \mathbb{R}^{\geq 0}$ *is the cost function defined for each* $s \in S$ *and* $t \in D$ *as:*

$$C(s, t, a) = u_a^+ P_I(t) + u_a^- P_E(t) - C^*(t) \tag{3}$$

$$C^*(t) = \min \begin{Bmatrix} u_{\max}^+ P_I(t) + u_{\max}^- P_E(t), \\ u_{\min}^+ P_I(t) + u_{\min}^- P_E(t) \end{Bmatrix} \tag{4}$$

where, given s *and* t*:*

$$u_a = \begin{cases} U(t) + \min\{\lambda,\ \beta - s\} & \text{if } a = 1 \\ U(t) - \min\{\lambda,\ \beta - s\} & \text{if } a = -1 \\ U(t) & \text{if } a = 0 \end{cases} \tag{5}$$

$$u_{max} = U(t) + \lambda \tag{6}$$

$$u_{min} = U(t) - \lambda \tag{7}$$

such that $x^+ = max\{0, x\}$ *and* $x^- = min\{0, x\}$ *for any* $x \in R$

- $t_{max} \in N$ *is the horizon with* $D = \{1, 2, \ldots, t_{max}\}$
- $s_1 \in S$ *is the initial state*

III. Design of Smart Home Battery XAI system

This section introduces the design of the Smart Home Battery (SHB) XAI system. Our first research question is to examine *if visual information alone is sufficient or visual information plus contrastive explanation is more useful for non-experts* to understand a planner's output. In order to achieve this, we designed two versions of a user interface, one for each user group (which will be discussed in detail in the next section).

A. State-constraint and action-constraint

Fig. 1 shows the dynamic process of planning without constraints, which is implemented as a search algorithm traversing the MDP model as nodes to obtain the lowest cost path (or plan), as introduced in Preliminaries.

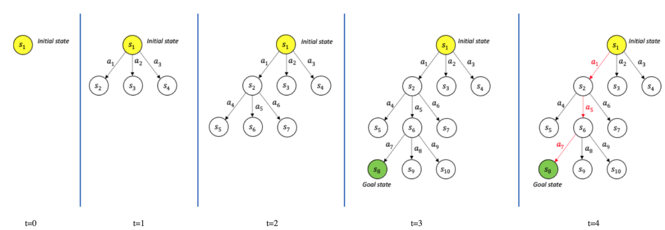

Fig. 1. Planning without constraints is shown, the yellow node is the initial state, the green node is the optimal goal state and the red path is the optimal plan (optimal policy π^*) in this goal state.

State-constraint and action-constraint are formalised as follows:

Definition 3. *Let* S *be a set of states and* D *be a set of timesteps. A state-constraint is a function* $f : D \to 2^S$ *where* $f(t) \subseteq S$ *is the set of acceptable states at timestep* $t \in D$*. A plan* π *satisfies state-constraint* f *if, for any execution of* π *from initial state* $s_1 \in S$*, it is guaranteed that* $s_i \in f(t_i)$ *for every timestep* $t_i \in D$ *with* s_i *the state at* t_i*.*

According to Definition 3, a scenario of dynamic process of planning with state-constraint is shown in Fig. 2.

Definition 4. *Let* S *be a set of states,* A *be a set of actions and* D *be a set of timesteps. An action-constraint is a function* $g : D \to 2^A$ *where* $g(t) \subseteq A$ *is the set of acceptable actions at timestep* $t \in D$*. A plan* π *satisfies action-constraint* g *if, for*

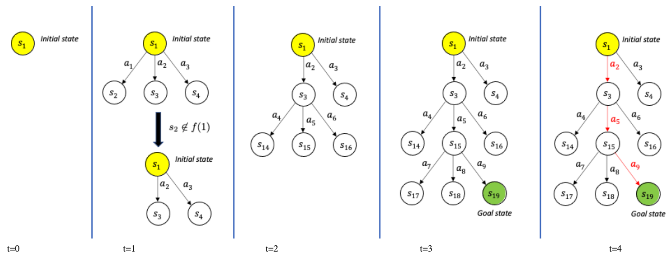

Fig. 2. Following from Fig. 1, when state-constraint is enforced, $s_2 \notin f(1)$ removes the left most branch forcing the planner to search for other policies as indicated by the red-coloured path.

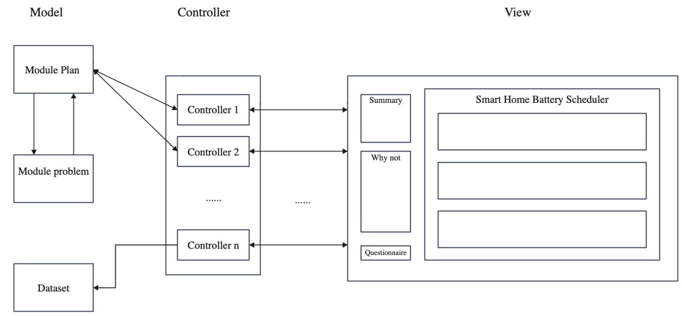

Fig. 4. System Architecture

any execution of π from initial state $s_1 \in S$, it is guaranteed that $\pi(s_i) \in g(t_i)$ for every timestep $t_i \in D$ with s_i the state at t_i.

According to Definition 4, a scenario of dynamic process of planning with action-constraint is shown in Fig. 3.

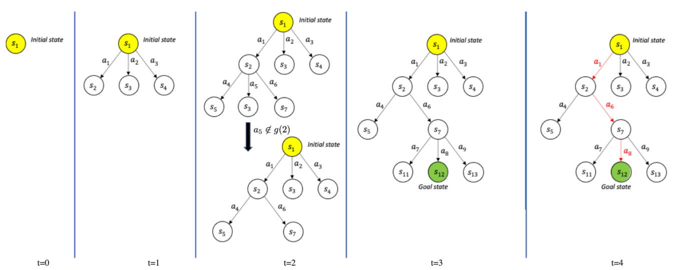

Fig. 3. Planning with action-constraint is shown, where $a_2 \notin g(2)$ affects the search path of an optimal policy.

The definiton on *contrastive explanation* for planning is given in Definition 5.

Definition 5. *Let $\Psi(\cdot)$ denotes an explanation function. Let $\Psi(\pi)$ be an explanation for plan π and $\Psi(\pi|f)$ (or $\Psi(\pi|g)$) be an explanation for plan π after state-constraint and/or action-constraint are imposed on π. Then the visual and/or textual comparison of the effects of $\Psi(\pi)$ and $\Psi(\pi|f)$ (or $\Psi(\pi|g)$) is called contrastive explanation.*

In this study, we instantiate function $\Psi(\cdot)$ to be a procedure which accepts new constraint values on state or action variables, passes the values to the back-end planner and then feeds both the original plan and the newly generated plan to the front-end.

B. System Architecture

The architecture of SHB system is shown in Fig. 4, this system is developed according to the Model-View-Controller (MVC) design pattern. Once the system is released, users can access the SHB system through a browser.

The visualisation pages belong to the *View*. The back-end is divided into two modules, the module interfacing with the front-end belongs to the *Controller* and the other modules for data processing belong to the *Model*. After the user selects a specific operation in the front-end, the instruction is passed from the *View* to the *Controller*, where the instruction finds the corresponding interface to call the module in the *Model* to address the specific task, and then the result is returned to the view according to the previous route.

Module Plan contains the code related to planning and *Module problem* refers to the code related to the MDP model. The *Dataset* is used to store information about the user's ratings in the questionnaire which will serve the evaluation task at a later stage.

An interface of the SHB system is shown in Fig. 5 and consists of two panel: the *Visual Information* panel (Visualisation panel) on the right, and the *Textual Information* modules on the left, both presenting the outcomes of a plan in the most visual way possible.

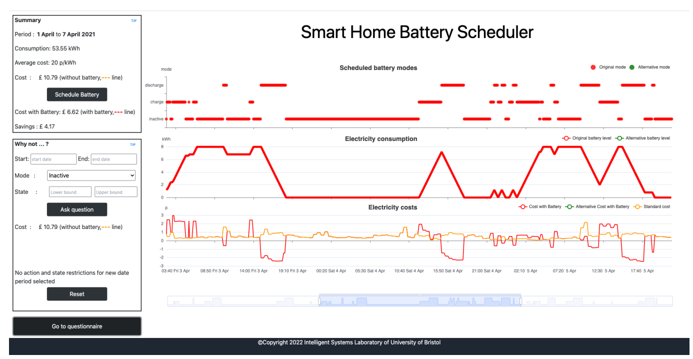

Fig. 5. An interface for presenting visual and textual information to non-experts.

C. Visual Information Interface Design

The Visual Information panel consists of three plots which show the *Scheduled battery modes*, *Electricity consumption* and *Electricity costs* from top to bottom.

The plot of *Scheduled battery modes* plots the data at this period on a scatter plot and tells the user exactly which action (mode) the battery to perform at each time step. For example, in 02:10-02:15 on 4 April the battery in the mode of "discharged" and in 13:30-13:35 on 6 April the battery in the mode of "charged" .

The plot of *Electricity consumption* is a line graph plotting the capacity of the battery at each point in time. The *Scheduled battery modes* and the *Electricity consumption* are precisely correlated. For example, if the battery is discharged at 3:55 on 3 April, the battery capacity falls and if the battery is charged at 18:45 on 3 April, the battery capacity rises.

The plot of *Electricity costs* depicts the costs of standard electricity (yellow line) and the costs of accessing (red line) the battery at each time point.

D. Textual Information Interface Design

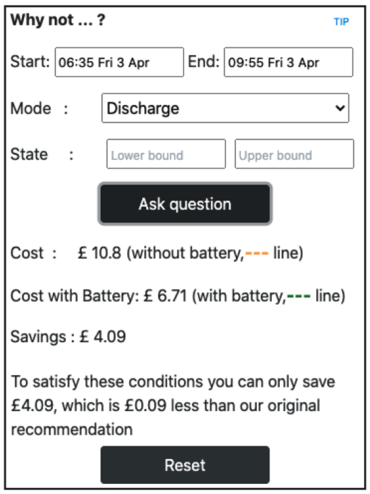

Fig. 6. The *Why not ...?* panel

The *Textual Information* is made up of two panels, the *Summary* panel and the *Why not ...?* panel. This module describes the most important information compactly.

The *Summary Panel* contains six data items as well as the *Schedule Battery* button; the *Period* indicates which period of time the system is currently displaying; the *Consumption* shows how much power was consumed during that period; the *Average cost* indicates the average cost to the user during that period; the *Cost* tells the user how much it has cost under standard conditions; and the *Cost with Battery* column tells the user how much it costs with battery access. For example, the *Summary Panel* in Fig. 5 shows that the customer consumed 93.64 kWh of electricity from 1 April 2021 to 7 April 2021, costing £10.80, with an average cost of 11.53 pence per kWh. If the battery was used, the customer would only have been charged with £6.62.

The function of the *Why not ...?* panel allows the user to set constraints and to re-plan according to the constraints. The constraints in re-planning and its outcome can be easily compared with the original planning data in the *Summary* panel demonstrating the *contrastive explanation* effect. The *Why not ...?* panel consists of three rows of input, two buttons and four items of data. This panel is described by the example in Fig 6, and the specific interaction functions are described in the next section. The original plan in Fig 6 gives the user a standard cost of £10.80, whereas the constrative replanned cost with using a battery is £6.71, a saving of £4.09. The user is then told via textual information that the new plan with the user's constraints actually costs £0.09 more than the original plan.

IV. Interaction of XAI in SHB

In our interactive interface, we allow users to reset one or both of constraints in order to see the effects of these changes on battery status and prices. Therefore, our second research question is: *which type of constraints do users use often, state-based (change the mode of battery) or action-based (change the charging state of a battery).*

We plan to divide our users into two groups: one is the *Control group* with only visual and summary information, and another is the *Treatment group* with visual, summary and contrastive interactions.

A. Interaction for Users in the Control Group

The *Control group* interface is shown in Fig. 7. The figure shows that users in this group can see *Visual Information* and limited *Textual Information*. The interactive functions of the control group are described next.

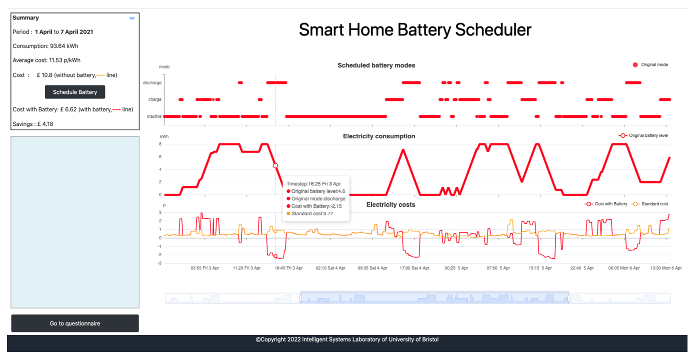

Fig. 7. Control group system interface

In *Summary* panel, the system provides the user with the function to select a time period. If the *Schedule Battery* button is clicked, a modal box will pop up allowing the user to select a new time period to be observed. The system will always display a default setting (e.g., 1st April to 7th April 2022) to start with.

In *Visualisation* panel, at the bottom of this area there is a *Datazoom* slider, which can be dragged to zoom in (see Fig. 8(a)) and out (see Fig. 8(b)) on the data in the *Visualization* interface.

(a) Zoom in on data (b) Zoom out on data

Fig. 8. Zoom in and out operations in the Visualisation panel

In this panel, there is a synchronous relationship between the three plots in terms of zooming in and zooming out operations. When a user selects a specific point on any plot, the data for all three plots corresponding to the same timestep will be displayed in the style of bullet points in tooltip.

For instance, in Fig. 7, with timestep at 16:25 on the 3rd April, the pop-up box provides all the details relevant to battery power, price of using the battery, standard price and the battery mode at the time.

B. Interaction for Users in the Treatment Group

Treatment Group users can use all the functions available to the Control Group. Functions specific to this group are introduced below.

The three lines of the input box are intended to allow a user to do the following three things: (i) set the period for re-plan; (ii) change actions; (iii) change states. The latter two affect the *mode* of a battery. These interactions are for gathering constrains from a user for generating contrastive explanations. For the start date/time and end date/time input boxes, the user is allowed to enter them manually, or select them by clicking on the points in the plot of *Scheduled* battery modes because the plot is a scatter plot and it is easier for the user to find a specific point by zooming in and out. Note that if a user chooses the latter, the selected points will replace the previous inputs in turn. Once a user has entered all the constraints, the system will re-plan upon the user clicking on the *Ask question* button.

The green lines/dots are for after re-plan whilst the red line/dots are for the original plan. This way, we can constratively visulize the effects of the two plans. The user can also look at a particular plan by clicking on the legend in each plot. If a user wants to experience more about the effects of different constrains on new plans and their associated cost savings/losses, they can use the *reset* button to remove the previously entered constraints and enter new constraints to re-plan again.

C. Gamification in contrastive explanation

Gamification is a useful way for a user to learn in a given setting. Here in our design, we borrow the concept of *gamification* which means we allow a use to reset some values of variables which can affect the outcome of a planner. This way, a user is able to see the comparisons through *contrastive explanation*. The outcome is displayed in the *Summary* panel and the planner is informed of the difference in cost savings between the re-plan and the original plan. The smaller this value is the less impact the restrictions added by the user have on the original plan and the closer the effect of the re-plan is to that of the original plan.

By allowing a user to experiment multiple times, we can observe if a user has gained the understanding of the explanations provided, so that their next round of reset of constraints values are closer to producing optimal outcomes (e.g., maximal savings).

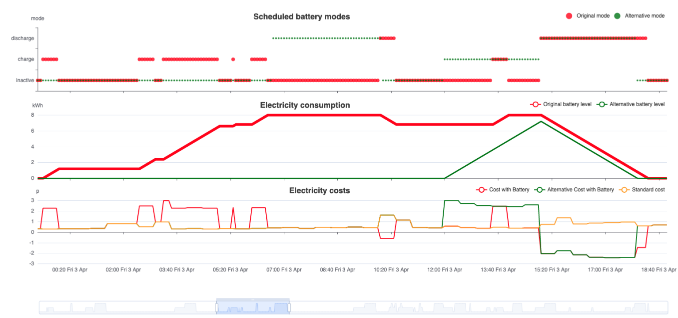

Fig. 9. Action-constraint

Once constraints are entered and a new plan generated (e.g., a revised cost obtained), the user can focus on the modified part by means of a zoom operation and visual displays of different colour-coded lines in Fig.9, where the red and green lines (or points) show details of process on planning without constraints and planning with action-constraints or state-constraints respectively. The user can clearly see the comparisons of changes in the two planned actions at each moment and the changes in battery state, as well as the changes in costs at each moment. Each user will be allocated 10 minutes to experience the system freely and after which the user will stop all operations and will proceed to click on a button leading to the questionnaire. Users can have unlimited number of tries, and for each round they do so, constraints they entered will be recorded in the log for later analysis.

TABLE I
QUESTIONNAIRE

			Likert scale labels	
		Question	(1)	(7)
C and T Group	Q1	How difficult was it for you to understand the visual information on the right panel	very easy	very difficult
	Q2	Was the visual information and additional information provided for any specific datapoint useful?	not useful at all	very useful
	Q3	Was the visual information helpful for you when you look at the Summary information?	not useful at all	very useful
	Q4	Were you able to enter more appropriate values using Schedule Battery button in subsequent attempts?	not any better	much better
	Q5	Would you like to have some explanations about a Summary?	no preference	very much so
	Q6	Would you have expected to see some textual explanations in addition to visual information?	not expected	fully expected
T Group	Q7	How useful was it to allow you provide constrains to see different prediction results in the “Why not” panel?	not useful	very useful
	Q8	Were the contrasting visual displays of the original predictions versus alternative predictions useful?	not useful	very useful
	Q9	How satisfied were you with the textual information provided in the “Why not” panel following the click of the “Ask question” button?	not useful	very useful

V. Discussion and Conclusion

A. Discussion

The evaluation of XAI differs from machine learning. Machine learning has explicit evaluation standards and a reference baseline, and typically model performance metrics, such as accuracy, increase with the number of training rounds and eventually converge to a specific value. However, these methods are difficult to be transferred directly to the evaluation of existing XAI systems, as each individual has a subjective view, which makes it difficult to obtain a uniform standard for evaluating systems.

Authors in [12] explained the decision tree model with the help of an explanatory tool, then designed a questionnaire to invite participants to evaluate the model according to its characteristics, and finally analysed the data collected.

Authors in [13], [14] first conducted a theoretical study of planning, after which the previous research results were developed into an explanatory system [15]. The way both studies evaluated the explanation was similar that is, both invited users to evaluate their system through a questionnaire. The questionnaires used the Likert scale, the former on a 5-point scale and the latter on a 7-point scale.

However, there are still some differences in the workflow, [12] did not develop a complete system and only presented the explanation of the model to the user in a static interface, there was no interaction in the process. Research in [13]–[15] was much more systematic in that they first conducted theoretical work and then developed the theoretical results plus explanations into a system for users to experience and evaluate. These studies also introduced a control group into their evaluation so that comparisons could more visually highlight the usefulness of the explanations. The latter evaluation will be more convincing than the former. However, whereas the latter gives explanations involving a large amount of text, our explanations are predominantly visual and we show more detail as well as provide action-constraint and state-constraint service. Different from the above two studies, our XAI interface is interactive, with both visula and textual information. Furthermore, we provide constrstive explanation in both of these forms too. Table I, shows the questions we are going to use in our questionnaire.

Our next step is to invite participations and randomly divide them into the Control and Treatment groups. Our analysis and evaluation are to address the two research questions we posed early and to provide some deeper understanding as how contrastive explanation can influence a user's understanding of an XAI system. We will also want to discover if visual information alone can achieve significant satisfaction from user about the basic understanding of the system, so that in some situations, textual or contrastive explanation can be omitted.

B. Conclusion

In this paper, we introduced, designed and implemented an XAI system for a planner for battery management at homes. Our XAI interface design was guided by two research questions, that is, is visual information alone useful or shall we always provide more sophisticated explanation options (e.g., contrastive explanation). The paper discussed how users can better understand the system and build trust with it through comparative explanations and iterative planning. This is followed by a discussion of related research and our future work.

References

[1] D. Zhang, Z. Wu, J. Chen, R. Zhu, A. Munawar, B. Xiao, Y. Guan, H. Su, W. Hong, Y. Guo *et al.*, "Human-robot shared control for surgical robot based on context-aware sim-to-real adaptation," *arXiv preprint arXiv:2204.11116*, 2022.

[2] H. Bo, R. McConville, J. Hong, and W. Liu, "Social influence prediction with train and test time augmentation for graph neural networks," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[3] H. Tang, D. Xu, N. Sebe, Y. Wang, J. J. Corso, and Y. Yan, "Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2417–2426.

[4] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial intelligence*, vol. 267, pp. 1–38, 2019.

[5] K. McAreavey, K. Bauters, and W. Liu, "A smart home testbed for evaluating xai with non-experts." in *ICAART (3)*, 2022, pp. 773–784.

[6] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller, "Explaining deep neural networks and beyond: A review of methods and applications," *Proceedings of the IEEE*, vol. 109, no. 3, pp. 247–278, 2021.

[7] R. Li, J. Wang, and Y. Shi, "Passivity-based formation control for second-order multi-agent systems with linear or nonlinear coupling," *International Journal of Control*, pp. 1–12, 2022.

[8] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations research*, vol. 35, no. 2, pp. 254–265, 1987.

[9] M. Fox, D. Long, and D. Magazzeni, "Explainable planning," *arXiv preprint arXiv:1709.10256*, 2017.

[10] R. Eifler, M. Brandao, A. J. Coles, J. Frank, and J. Hoffmann, "Plan-property dependencies are useful: A user study," in *ICAPS 2021 Workshop on Explainable AI Planning*, 2021.

[11] P. Lipton, "Contrastive explanation," *Royal Institute of Philosophy Supplements*, vol. 27, pp. 247–266, 1990.

[12] Y. Zhang, K. McAreavey, and W. Liu, "Developing and experimenting on approaches to explainability in ai systems." in *ICAART (2)*, 2022, pp. 518–527.

[13] R. Eifler, M. Cashmore, J. Hoffmann, D. Magazzeni, and M. Steinmetz, "A new approach to plan-space explanation: Analyzing plan-property dependencies in oversubscription planning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 9818–9826.

[14] R. Eifler, M. Steinmetz, A. Torralba, and J. Hoffmann, "Plan-space explanation via plan-property dependencies: Faster algorithms & more powerful properties," in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021, pp. 4091–4097.

[15] R. Eifler and J. Hoffmann, "A web-based platform for iterative planning with plan explanations."