Finding relevant causes in complex systems: A generic method adaptable to users and contexts

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

3rd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

***Abstract*—Complex adaptive systems (CAS) feature intricate dynamics that are difficult to track and control. This paper addresses the challenge of providing causal explanations for unwanted or unexpected events in CAS, e.g. "why did this happen?". Previous work in cognitive science indicates that satisfactory explanations should identify relevant causes tailored to the user's query and context. While automated methods exist for generating causal explanations, they typically overlook relevance or rely on a static notion of "best cause". We propose a novel two-step methodology to provide relevant causes: (1) identify a wide range of causes; (2) rank and filter them using a flexible evaluation framework that can be tailored to users. We rely on an existing cause-detection method (1); and focus on selecting the most relevant ones (2). We propose three relevance metrics and a flexible method for combining them. We validate our framework using a flocking simulation where agents encounter obstacles. Experiments demonstrate how the flexibility of our method allows us to generate diverse causal explanations, each highlighting the most relevant aspects to the user.**

I. Introduction

Complex Adaptive Systems (CAS) are composed of numerous interrelated components, typically featuring non-linear dynamics and emergence [1]. This implies large-scale observations, often requiring abstraction to simplify tracking and interpretability, e.g. [2]. Temporal dependencies at various granularities can also yield complicated dynamics [3]. These properties imply complex causal dependencies. Finding causes from the system models, equations, or data is difficult or infeasible in this context. Each event can have numerous causes due to the number of interrelated components, the spatial and temporal system boundaries, or the variety of temporal, spatial, and conceptual abstractions. Hence, causes can form various, long, and complicated causal chains. It is hard to make sense of such causal information.

Explanations are needed when a user encounters an unwanted or unexpected event. Often, the task is to answer the question, "Why did this event happen?" This type of question is studied in philosophy under the notion of causality [4], [5]. The notion of "cause" is defined as an answer to this question [6]. Notably, intuitions in philosophy and law suggest that the cause is a necessary condition for the consequence, i.e., "but for" the cause, the consequence would not have happened. Following these considerations, a mathematical formalization has been proposed for "counterfactual causes", i.e. in counterfactual situations, the consequence did not happen when the cause did not happen. This notion has already been used for explanations in various engineering systems, e.g. [7], [8], [9].

Research in cognitive science has shown that what people expect when they ask questions like "Why did this event happen?" are specific causes that are filtered according to their relevance to the context [10]. Typical research on explainability fails to match this intuition. Some methods focus on interpretability and do not generate causes as facts, which is the case for the majority of eXplainable Artificial Intelligence (XAI) [10], [11]. Other methods focus on the cause generation process and do not consider any notion of relevance [9], [12], [13]. Finally, some solutions aim to optimize for one "best cause" [7], [14], without considering the fact that users might differ in their profile and context-dependent expectations.

Our approach consists in: (1) finding a wide range of causes; and (2) ranking and filtering them in a flexible way according to relevance. We identify notions of relevance from the literature; and propose practical metrics to score causes based on these notions. Some dimensions are more important than others, depending on the user and context. A full theory of relevance would provide the appropriate ranking given a context. However, we do not tackle the challenge of modeling the user and the context. Instead, we propose a flexible aggregation method for selecting causes (2) from a large predetermined set (1), based on user preferences and context.

The proposed method is summarized in Fig. 1, where dotted boxes represent external inputs. The method is triggered when a user asks for an explanation for a surprising event. It starts with an external cause-identification step. Next, it assesses the relevance of causes using several metrics. Finally, it aggregates these, based on an external configuration, to propose the most relevant causes to the user (i.e. explanation).

To illustrate the feasibility of our proposal, we implemented a flocking simulation (inspired by [15]) where agents move and eventually hit an obstacle (i.e. unwanted event). We designed three scenarios where the expected explanations differ. We show that our method (a) can generate different relevant explanations depending on the context, and (b) can be tuned to produce diverse explanations adapted to various types of users.

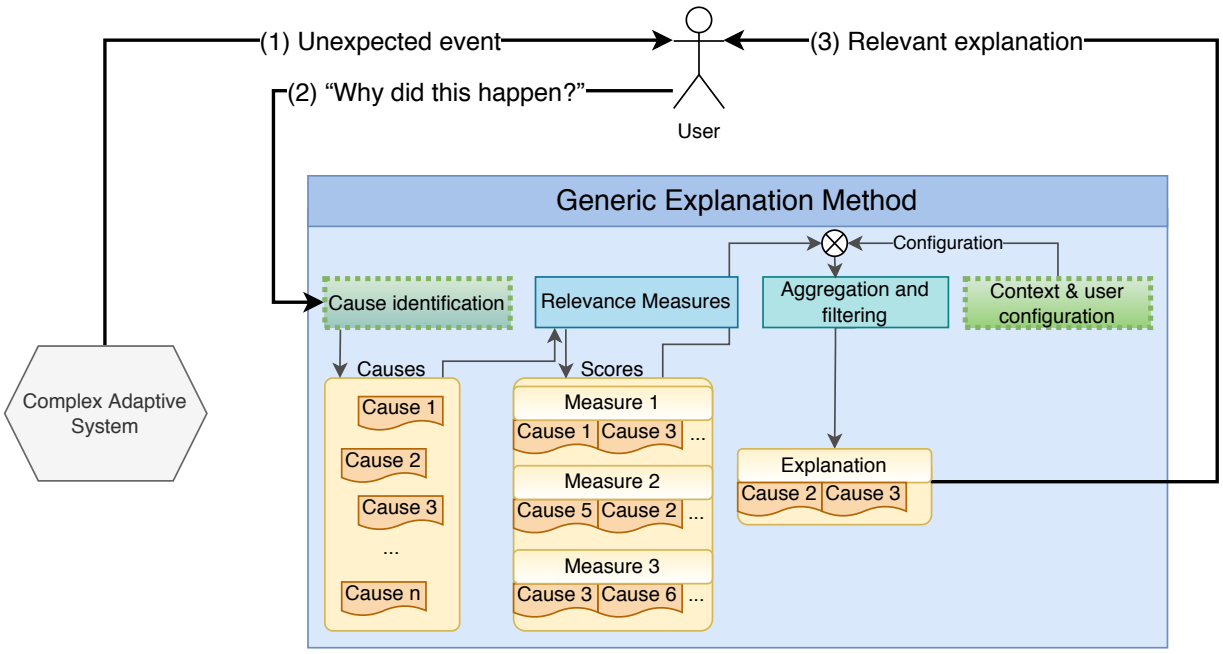


Fig. 1: An overview of the proposed generic explanation method.

II. Background

In this section, we review the notions that we use in the remainder of the paper and that motivate our approach.

A. Causality

Our approach provides explanations to users asking questions such as "Why did this event happen?". This challenge has been studied extensively in philosophy [4], [5] and has been formalized by Pearl [6]. The key intuition is that a cause is a necessary condition for the consequence, i.e. if the situation did not occur, neither would the consequence.

To formalize this notion, Pearl relies on Structural Causal Models (SCMs). An SCM is a tuple $\mathcal{M} = (\mathbf{X}, \mathbf{U}, \mathbf{F}, \mathbf{R})$ with $\mathbf{X}$ being the set of endogenous variables, $\mathbf{U}$ the set of exogenous variables, $\mathbf{F}$ the set of structural assignments, and $\mathbf{R}$ the set of domains. For any of the endogenous variables X_i, there is a corresponding exogenous variable U_i and a structural assignment F_i such that $X_i = F_i(pa_i, U_i) \in R_i$ with $pa_i \subset \mathbf{X}$ being the causal parents of X_i. The parent relationship forms a Directed Acyclic Graph (DAG), i.e., no ancestor of a causal variable can be its descendant. This allows the computation of the values of all causal variables by giving only the values $\mathbf{u}$ of $\mathbf{U}$, which is called a context. In the remainder of the paper, we will also refer to the endogenous variables X as the "causal variables".

Pearl's formalism introduces the notion of the truth of a predicate under an SCM and a context. Given a boolean function ϕ with input $\mathbf{Y} \subset \mathbf{X}$, its value can be computed based on $(\mathcal{M}, \mathbf{u})$ by applying the structural assignments to compute the values of $\mathbf{Y}$ and then applying ϕ. To express the truth value of the predicate, we write $(\mathcal{M}, \mathbf{u}) \models \phi$.

Another pivotal notion is that of interventions. Pearl's formalism allows "forcing" certain variables to take certain values regardless of the output of the structural assignments, which then influences the values of its descendants. When an intervention is performed, the truth value of the predicate can be impacted, e.g., $(\mathcal{M}, \mathbf{u}) \models [\mathbf{Y} \leftarrow \mathbf{y'}]\neg\phi$.

Finally, the notion of cause is formalized under a definition by Halpern and Pearl that is often referred to as HP-cause [16]. To address some specific cases, the notion of a "contingency set" is added to the intuition of counterfactual cause. We aim to check if the consequence does not happen when the cause does not. Additionally, we allow some variables to keep their actual value when we force the cause not to happen. This definition has been debated, and the most widely accepted version is the one that follows:

Definition: HP actual cause [17]. $\mathbf{C} \subset \mathbf{X}$ is the cause of $T \in \{0, 1\}$ with respect to $(\mathcal{M}, \mathbf{u})$ if:

- $(\mathcal{M}, \mathbf{u}) \models (\mathbf{C} = \mathbf{c})$, $(\mathcal{M}, \mathbf{u}) \models T$
- $\exists \mathbf{c'}, \mathbf{W} \subset \mathbf{X} \backslash \mathbf{C}, (\mathcal{M}, u) \models (\mathbf{W} = \mathbf{w})$ such that $(\mathcal{M}, \mathbf{u}) \models [\mathbf{C} \leftarrow \mathbf{c'}, \mathbf{W} \leftarrow \mathbf{w}]\neg T$
- No subset of $\mathbf{C}$ satisfies the above conditions.

In the remainder of the paper, we will refer to the state obtained after the intervention $[\mathbf{C} \leftarrow \mathbf{c'}, \mathbf{W} \leftarrow \mathbf{w}]$ as the **"counterfactual"** associated with the cause. This terminology is standard in the XAI literature [18].

The HP definition has been widely used in the literature. However, researchers have proposed several modifications (see sections II-B4 and II-B5). Additionally, HP actual causes identification (also referred to as actual causation analysis) is an NP-complete problem [17], which makes exact methods unrealistic in practice.

B. Explanations

In a survey of the XAI field, Miller [10] reviewed related notions in cognitive sciences. He suggests that what people expect when they ask for an explanation in everyday usage are causes that are filtered according to relevance with the context. We review several concepts from the literature that relate to measuring the relevance of causes.

1) Minimum description length: Research has shown that relevance can be measured using Kolmogorov complexity [19], which is not computable [20]. The most popular approximation for Kolmogorov complexity is the Minimum Description Length (called pure MDL to distinguish it from a homonymous notion) [21]. Pure MDL is the minimum number of bits necessary for a program to generate the described object. Computing this measure requires optimizing the sum of two terms. The first term is the description size of a program that can generate the object we are trying to describe. The second term is the size of the description that the program generates.

2) Memorability: Memorability has also been proposed as a way to compute the simplicity of events. A work related to causal explanations [22], showed that temporal proximity to significant events contributes to memorability. In the context of causation, this is equivalent to **recency** concerning the target event. Recency is key to complement HP-causes that do not consider temporality.

3) Material cost: A study about causality in CAS [23] showed how causes and explanations can be utilized either to enhance the understanding of a system or to determine what actions can be taken to avoid a particular situation. This second argument points to practical considerations such as cost or energy resources, which are particularly relevant when the explanation is used to improve control of the system.

4) Responsibility: The notion of cause has been formalized in relation to law (e.g., but-for cause). An important concept that has been concurrently introduced is the degree of responsibility. Researchers have suggested that responsibility can be measured in the context of causation as the inverse of the size of the cause [17], [24]. Formally, if $\mathbf{C} \subset \mathbf{X}$ is a cause of $T \in \{0, 1\}$ in $(\mathcal{M}, \mathbf{u})$, following the HP definition (see Section II-A), the degree of responsibility of $\mathbf{C}$ is $deg(\mathbf{C}) = \frac{1}{|\mathbf{C}|}$, where $|.|$ refers to the cardinality of a set. The intuition is that when there are more causal variables in the cause, each one is less responsible for the outcome. For instance, consider a majority vote between options A and B with 11 voters. If the outcome is 6/5, changing only one vote will be enough to reverse the outcome of the vote, and the responsibility of the agent is high. Conversely, if the score were 10/1, causes would include 5 agents and have lower responsibility. This notion has been argued to rate the importance of a cause.

5) Normality: Another classic notion introduced to complement the HP definition is normality, which has been used in several ways. Notably, researchers have pointed out that more abnormal causes are more relevant than normal ones [25], [26]. In this work, we will focus on another use of normality for causal relevance. The main intuition is that the counterfactual world, where the cause and the consequence do not happen, cannot be too abnormal. Halpern et al. [27] proposed measuring the normality of the counterfactual as its probability within the space of possible worlds. They proposed adding a condition to the HP definition that removes causes associated with abnormal counterfactuals.

III. Related work

There is a wide range of scientific work on causal explanations. Several approaches specifically focus on identifying actual causes, while others focus on the concept of explanations. We mention specific methods for explainability, notably for modern AI systems and for cyber-physical systems, which more closely match the scope of this paper.

A. Causation analysis

The literature on causality is typically focused on causal discovery [28] and causal representation learning [29]. In this work, we argue that explanations are made of causes, which channels our review to causation, i.e., answering the question "What caused this event?" Additionally, we focus on actual causation analysis and do not consider "general" or "type" causation. Type causation answers questions such as "Which factor may have caused this consequence?" and aims at statements such as "Smoking causes cancer". What we aim for, following actual causation considerations, is answering the question "Which event caused this consequence?", which corresponds to statements such as "She had cancer because she smoked for years".

Few approaches identify actual causes. Some follow the HP definition but develop methodologies that only work on boolean systems [9], [12]. Others modify the definition only slightly but use optimization methods designed to generate the best cause following one additional consideration, such as causal sufficiency [7] or normality [14]. Finally, certain approaches adapt the HP definition to specific challenges, such as temporal series [8] or causal dependencies [13]. In upcoming work, we developed an approximate search algorithm to identify various HP causes [30]. We use this algorithm as a first step in our methodology. However, none of these methodologies proposes ways to filter the causes, which rules them out to produce explanations. In a typical CAS, millions of causes can be identified. The previous methods either generate a single one or do not filter them at all.

A theory of causal relevance using simplicity theory has been proposed [31]. However, this approach focuses on modeling cognitive argumentation and only measures how users view causality within their minds. It does not address causes identified from real systems.

Finally, a concept of explanation based on causality has been proposed by Halpern and Pearl conjointly to their proposition of the HP definition [32]. They suggest that an explanation is needed when users have an incomplete knowledge of the system. Notably, users know about the cause and the consequence but not about the context. According to Halpern and Pearl, an explanation is a context in which the cause

holds. This view of explanations does not match the one we address. We follow previous work that instead characterizes explanations as causes [10], [33], [34], [35], [7].

B. XAI

The field of XAI has gained tremendous popularity with the rise of deep learning, developing a wide taxonomy of methods [36], [37]. For instance, popular methods include "black box" XAI that aims at highlighting the parts of the input that influenced the model in its prediction [38], [39]. Another family of methods called counterfactual explanations [18], resembles causation analysis. Given an input of the model and its output (supposedly unsatisfactory), they aim at exhibiting another input, as close as possible to the original one, that is predicted differently by the model.

However, the field of XAI has been criticized. It has been described as too focused on practitioners' need for interpretability instead of everyday explanations for various users [11], [10].

C. Explanations in CPS

Explanability in Cyber-Physical Systems (CPS) focuses more on practical explanations for users. Moreover, they relate more closely to the context of CAS. However, the literature is thinner and there is no taxonomy as clear as in XAI. While the concept of explainability has been laid out in clear frameworks [40], the practical methodologies are many. Some focus on identifying the origin of anomalies with statistical methods [33], while others focus on the decision process of the controllers using a method from XAI literature [41]. In a different approach [34], [35], the nature of the cause identification process is not central, and the focus is laid on an interactive dialog with the user in a decentralized framework that mimics a cognitive conflict-solving process. Finally, following consideration similar to our paper, other methods focus on relevance to the user and to the context [42], [43]. However, they disregard the importance of cause identification for explanations and only consider rule-based systems.

IV. Proposed methodology

To address these limitations from both the causality and explainability literature, we propose a two-step methodology:

1) Identify a wide range of causes.
2) Rank and filter them to highlight the relevant ones.

The SCMs (potentially at various levels of abstraction) and the cause identification algorithm are assumed to be given. Both depend on the type of system and the needs of the user. The cause identification algorithm must generate a sufficient number of HP causes.

Our main contribution is the proposal of this two-step process. Our secondary contribution is the suggestion of motivated candidates for the second step. In the remainder of this section, we present our suggestions for **(A) measuring the relevance** of the causes along three dimensions (section IV-A) and **(B) aggregating these scores** into a flexible relevance ranking (section IV-B).

A. Relevance measures

Relevance can be measured using distinct dimensions. We propose candidates and corresponding computation methods. They can be adapted depending on the system under analysis and the user's needs. Since they come from abstract research, their computation has to be approximated to consider complex adaptive systems. More accurate implementations are discussed in Section VII-B.

For each proposed dimension, we explain our motivation, its connection to the notion of relevance, and a computation methodology.

1) Description Length ('DL'): We approximate **pure MDL** (see Section II-B1) and **responsibility** (see Section II-B4).

We assume that each causal variable can be characterized by a set of items from the simulation or the SCMs (e.g., which agent is involved, the timestep, etc.). We count the number of description items necessary to identify the causal variables that compose it. When the cause includes several causal variables, the repeated items are counted only once each.

This approach approximates pure MDL by focusing on the second term of the two-part minimization process theoretically required. We suppose that the program used to generate the description of the cause is fixed (the cause is described by a set of items) and we only minimize the description itself (how many items are required).

This approach also resembles causal responsibility computations. If all causes are described with different elements, 'DL' is proportional to the length of the cause. Otherwise, the notion of responsibility can be applied to some component of the cause description (e.g., if the cause is about one agent instead of two, its responsibility is higher).

2) Time To Event ('TTE'): This metric is motivated by the notion of **memorability** described in Section II-B2. We approximate this notion by the recency of the cause relative to the target event. We compute this score by counting timesteps between the earliest variable in the cause and the event to be explained.

3) Intervention Cost ('Cost'): This metric considers both **normality** (see Section II-B5) and **material cost** (see Section II-B3). It computes the difference between the simulation state before and after the causal intervention $\mathbf{C} \leftarrow$ **c'** from the HP definition.

This relates to the notion of normality as it approximates how improbable the counterfactual world is. Instead of computing a distance to a "normal" distribution of system states, we compute the difference to our normality reference, which is the actual state.

Additionally, the intervention cost can be seen as the cost of an action to be taken to avoid unwanted consequences.

We propose a simple method to compute the intervention cost while preserving the intuition behind these notions. We compute the relative difference in the considered state between before and after the intervention. For each variable in the cause, we consider the counterfactual state that was induced by the corresponding intervention. We normalize the simulation variables for the actual and counterfactual states. Then, we

compute the total absolute difference between the two states. Each variable may vary between 0 and 100% of its range. The full change in the state can therefore vary between 0 and 100% times the number of simulation variables. We finally add the differences computed for each causal variable.

B. Aggregation and filtering

Our metrics measure several aspects of relevance. Each metric may be more relevant to one context than another. We aggregate them using a flexible method, allowing practitioners to adapt their explanations with separate approaches, such as dialoging with the user.

We propose using an ordering of the metrics and a "lexicographic" sort following this ordering. We start by sorting according to the first metric, then sort identical values according to the second metric, and so on. Simply changing this order provides the expected flexibility. Each specific ordering can be seen as a configuration of the method, adapted to a user and context.

Before proceeding to the lexicographic sort, we discretize the scores produced by the metrics, grouping their values into intervals. This ensures that there are sufficiently numerous causes with the lowest score for the prioritized metric, allowing sorting according to the following metrics to still have an impact. Practically, the intervals are computed using a geometric progression to obtain more precision for small values.

This aggregation method explicitly prioritizes metrics over one another, highlighting clear types of causes. Other methods that give softer priorities, such as a weighted average, may lead to less clear patterns (see more details in section VII-C).

Once the scores are aggregated in a final ranking, we filter the causes by keeping the one associated with the top n best scores.

V. Experiments

To evaluate our approach, we implemented a multi-agent simulation modeling swarms (inspired by [15]), which is described in subsection V-A. Agents exhibit flocking behavior and eventually collide with an obstacle, prompting a user to question the cause of the collision. We use three different setups to define three distinct scenarios, which we present in subsection V-B. We define observations at two levels of abstraction: the agent level and the flock level (see subsection V-C). Next, we associate causal variables with these observation levels and define SCMs using the simulation to model the structural equations (subsection V-D. We then use an external algorithm to identify the causes for each SCM. The algorithm is briefly described in subsection V-F. For each scenario, we combine the causes obtained from the different SCMs. Finally, we score each cause using our metrics and report several instantiations of the final ranking using various configurations (i.e. ordering of the metrics), which we present in subsection V-G. Our code can be found in our GIT repository[1].

[1]Repository is not shown here for anonymity and will be added for the camera-ready submission

A. Simulation

The simulation models agents representing flocks of birds or fish, each characterized by a position (x and y coordinates) and a direction. In the remainder of this paper, we will refer to the x, y, and angle of the agents as the simulation variables. Agents follow five rules:

- Alignment: align direction with neighbors.
- Cohesion: move closer to neighbors.
- Separation: move away from very close neighbors.
- Avoid Edges: move away from the edge.
- Avoid Obstacle: steer away from the obstacles.

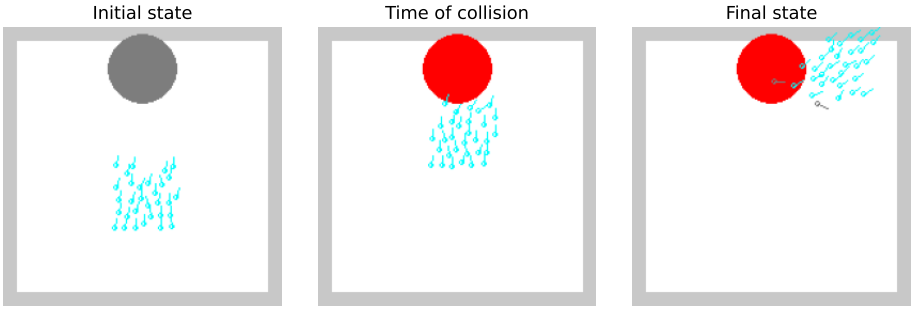
At each time step, identify their neighbors, check their proximity to edges and obstacles, adjust their directions according to the five rules, and move forward. A parameter is defined for close neighbor proximity (agents considered for the third force). Another one is instantiated for "classic" neighbors (agents considered for the first and second forces) and obstacle proximity, which we call "view radius". A third one, called padding, is used for proximity to the edges. Collisions with the obstacle are recorded for analysis. For the fourth force, the target direction is vertical when the agent is close to the top or bottom edge and horizontal otherwise. For the fifth force, the target direction is the tangent to the obstacle.

B. Scenarios

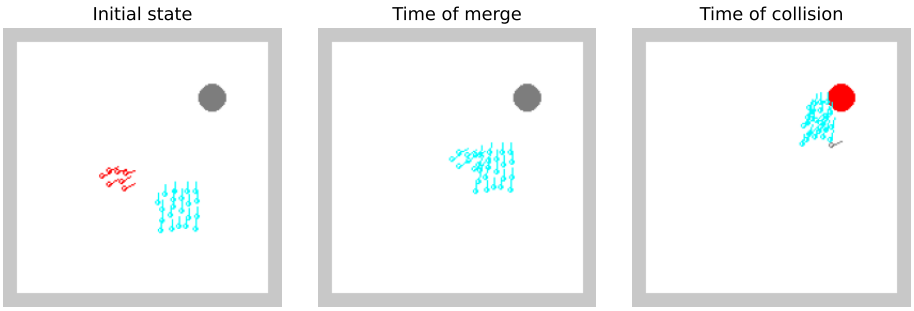
We simulate three different scenarios that we describe in this section. Significant timesteps are shown in Fig. 2. We represented the direction of the agents with a line. Colors are based on the flock identified at the given timestep. Agents colored in grey don't belong to any flock (more details on flock identification are given in section V-C). The grey area is the padding zone where agents start to execute the fourth rule. The obstacle appears in grey by default and in red when a collision is detected.

1) Single Flock Steering Towards the Obstacle: This scenario highlights the differences between flock and agent description levels. We aim at explanations that highlight the agents on the periphery, who play a crucial role in obstacle avoidance. Indeed, they are less constrained and see the obstacle first. The importance of periphery agents is also aligned with results in complex network theory. Researchers [44] have shown that dense and homogeneous networks, such as a flock, can be controlled using a few driver nodes. They also show that the driver nodes tend to have low degrees, which is the case for periphery agents. Fig. 2a shows three significant timesteps of the run.

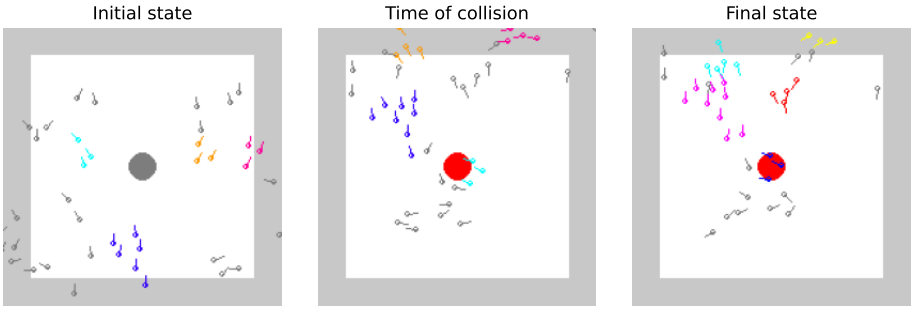
2) Two Flocks Merging and Deviating Towards the Obstacle: In this scenario, two flocks of different sizes merge, creating a dynamic similar to the first scenario. Both flocks would have avoided the obstacle if they had not merged. From a user's perspective, the smaller flock represents an obvious cause of the collision as it is intuitively interpreted as an accidental modification of the bigger flock's behavior. Fig. 2b shows three significant timesteps of the run.



(a) First scenario: Single Flock Steering Towards the Obstacle.



(b) Second scenario: Two Flocks Merging and Deviating Towards the Obstacle.



(c) Third scenario: Spontaneous Flock Formation and Collision.

Fig. 2: Significant timesteps of our three scenarios.

3) Spontaneous Flock Formation and Collision: This scenario presents a more complex situation where flocks form spontaneously. The smaller size of the responsible flock makes the difference between description levels less noticeable, posing a harder use case for causal explanation. Fig. 2c shows three significant timesteps of the run. This scenario has a larger padding area because we don't want agents staying outside the window's boundaries for long, whereas it did not matter much in the previous scenarios.

C. Multi-Scale Obvervations

We use two levels of abstraction to report observations about the simulation: agent and flock. Flocks are determined using DBSCAN [45] on the tuple (x, y, angle) of the agents. We use the implementation from the Sklearn library[2]. This algorithm creates clusters where each point is at least as close as ϵ to at least k other points. We choose the view radius parameter of the agents as the distance threshold ϵ, and a minimum of three points per cluster for k. For better tractability, we associate a flock ID with each combination of agents that have been observed forming a flock during the run. For instance, if agents 10, 11, and 42 form a flock at timestep 1, we will associate flock ID 0 with these agents. If agent 13 joins the flock at timestep 2, another flock ID will be used. However, if agent 13 eventually leaves the flock and agents 10, 11, and 42 form a flock again, the flock ID 0 will be used again.

We measure the x, y, and angle of each entity (agents or flocks). Flocks' values are computed as the average values of their agents.

[2]https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN

D. Causal variables

SCMs are tuple $\mathcal{M} = (\mathbf{X}, \mathbf{U}, \mathbf{F}, \mathbf{R})$ (see section II-A). In this subsection, we describe three ways of defining causal variables $\mathbf{X}$ (and their associated domains $\mathbf{R}$) based on our simulation and observations.

Pearl's original formalism does not incorporate notions of temporality. While our simulation has variables that evolve with time, the causal variables can only be defined for one timestep. We sample timesteps and consider separate causal variables at different timesteps. We sample timesteps using a geometric progression from the collision until the beginning of the simulation. For instance, the 'Flock fusion' scenario has 36 simulation timesteps, the first collision is reported at timestep 29, and the causal variables are taken at timesteps 0, 17, 24, 27, and 28.

We define a type of causal variable called 'Agent Variables' (AV) where the causal variables are the simulation variables (x, y, and angle of the agents) at the selected timesteps. For instance, the causal variable `agent_11_x_30` corresponds to the simulation variable which follows the x position of the agent with ID 11 taken at timestep 30. The domain of the causal variables is the same as the corresponding simulation variables.

Our second type of causal variable is called 'Flock Variables' (FV). They represent the x, y, and angle of the observed flocks at the sampled timesteps. For instance, the causal variables `flock_42_y_30` correspond to the average y value of the agents associated with flock ID 42 at timestep 30. We instantiate causal variables solely for flocks that are identified at the sampled timesteps. The domain of the causal variables is the same as the corresponding observations (x, y, and angle of the flocks).

We consider a third type of causal variable that encompasses the notion of 'Flock Aggregation' (FA). Similarly to the FV causal variables, we instantiate an FA causal variable for each flock observed at the sampled timesteps. The domains are all composed of three values. The value `full_agg` means that the agents associated with the corresponding flock ID form one single flock. The value `partial_agg` means that they form at least two flocks. Finally, the value `no_agg` means that no flock can be identified from these agents.

E. Interventions

During our experiments, we instantiate one SCM per type of causal variable and scenario. We compute the actual values of the causal variables based on our observations. However, to perform actual causation analysis, we need to perform interventions $\mathbf{C} \leftarrow \mathbf{c'}, \mathbf{W} \leftarrow \mathbf{w}$ as defined in section II-A, where $\mathbf{C}$ and $\mathbf{W}$ are sets of causal variables, $\mathbf{c'}$ is a set of counterfactual values for $\mathbf{C}$ and $\mathbf{w}$ is the set of actual values of $\mathbf{W}$. To compute the values of the other causal variables under an intervention, we need to compute the output of the structural assignment.

Our causal variables are such that their causal parents are other causal variables at the previous sampled timestep. The simulation can be used to model the structural equations. However, when we perform interventions on the causal variables, we must reflect these changes in the simulation variables to be able to run the simulation and get the values of the causal variables that are descendants in the causal graph.

The AV causal variables are simulation variables. The causal interventions are therefore immediate. To force the value of a causal variable, we set the simulation variable regardless of the simulation output at the given timestep.

The FV causal variables are computed as the average of their agents' variables. When we perform an intervention, we translate all the variables of the agents within the flock by the same amount we applied to the average value.

The FA causal variables represent how many flocks are formed by the agents associated with the variable's flock ID. We use the following procedure to reflect the intervention on these causal variables on the simulation variables. When going from a higher value to a lower one, we iteratively move the agents away from their average position and steer away from the average angle. We use a while loop to change the x, y, and angle values with a small amount until the expected value of the causal variable is reached. For instance, take a causal variable that symbolizes flock ID 0 made of agents 11, 12, and 42 at timestep 30. In the original context, its value is `full_agg` by definition. If want to reach value `no_agg`, we move and steer agents 11, 12, and 42 apart until they are all separated. Conversely, we move and steer agents toward their average values when we want to force a causal variable to a higher value.

F. Cause identification

The following subsection briefly summarizes the external algorithm we use for cause identification (see Fig. 1).

Identifying HP causes is a challenging process. The exact computation is NP-complete. We utilized an approximate algorithm from an upcoming paper [30]. This algorithm uses a specific kind of tree search, called beam search, to identify the minimal subsets of variables that meet the definition of HP-cause.

The algorithm checks if a subset of variables is a cause by performing an intervention on these variables. It then checks if the consequence holds when running the simulation. A set of counterfactual values is sampled for each causal variable. For the AV and FV causal variables, the counterfactual values are sampled using a geometric progression from the actual value to the edges of the domain. For instance, if an agent's x-position is 59, the counterfactual values will be 5, 44, 55, 58, 59, 60, 62, 73, and 112 (the width being 200). For the FA causal variables, the counterfactual values are {`full_agg`, `partial_agg`, `no_agg`} without the actual value.

G. Metrics and configurations

This subsection provides additional details on the practical implementation of the relevance measures, and aggregation and filtering method described in our proposed methodology

(see overview in Fig. 1 and details in section IV). We also indicate how we implemented the context and user configuration in practice (see Fig. 1).

Our methodology employs three relevance metrics. The 'TTE' metric is computed by subtraction between the timestep of the earliest causal variable in the cause and the timestep of the collision in the actual run. The 'Cost' metric is computed by normalizing the x and y positions by dividing by the width and height of the simulation window, and by 2π for the angle. Then we compute the difference in the overall simulation variables between the actual state and the counterfactual state. The 'DL' metric is computed by counting the different entity types (agent or flock), entity IDs, observation types (x, y, angle or aggregation of the flock), and timesteps that appear in the cause.

We then aggregate the metrics with a lexicographic sort following a configuration (i.e. an ordering of the metrics). Each configuration may yield a different causal explanation. Out of the six possibilities, we choose to showcase three: ('Cost' - 'TTE' - 'DL'), ('TTE' - 'Cost' - 'DL'), and ('DL' - 'Cost' - 'TTE'). Each of these configurations shows one metric as first priority. We observed that the second priority did not impact the top-3 causes generated.

VI. Results

Our experiments involve the three scenarios presented in section V-B, each associated with the three SCMs described in section V-D. We obtain hundreds of causes per scenario. We rate these causes using our three relevance metrics.

We review the distribution of the metrics and report the top 3 causes for the three selected configurations for each scenario. We show that these selected causes match the expectations presented in section V-B. Additionally, they reveal diverse types of causes that are relevant for different types of usages (e.g. better understanding or control of the system).

A. Distribution of the metrics

The distribution of the metrics' scores is reported in Fig. 3. We observe four main patterns.

1) When the causal variables are less abstract (based on agents rather than flocks), our algorithm identifies more causes. The numbers above the boxes indicate the number of causes identified. There are always fewer than ten causes for the FA variables, fewer than 100 for the FV variables, and more than 200 for the AV variables.
2) When the causal variables are less abstract, there are longer causes. We can see that the 'DL' metric generally has a higher median for AV variables than for FV. Additionally, the scores of the causes based on FA variables are always 4.
3) Less abstract causal variables yield causes related to earlier interventions. The medians for the 'TTE' metrics follow this trend in the 'Large Flock' and 'Flock Merging' scenarios. However, there is an exception with the 'Free Movement Scenario' where the median is high.

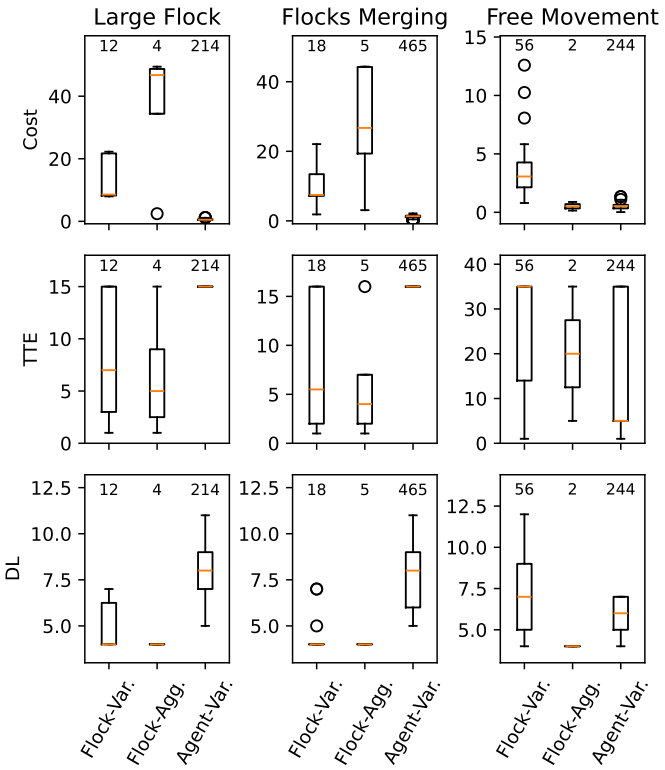


Fig. 3: Distributions of our metrics' scores.

4) Less abstract causal variables are associated with causes with higher costs. Notably, values of the 'Cost' metric are notably low with a low variance for each scenario.

Despite these statistical patterns, the metric distributions exhibit extreme values, suggesting that a more precise analysis is necessary. For instance, the median of the 'TTE' metric is large in the 'Free Movement' scenario, but the values drop to 0 in the lowest decile.

B. Explanations

We report the three best causes according to our three configurations in tables I, II, and III. Each configuration suggests a different causal explanation.

1) Intervention cost: Prioritizing 'Cost' suggests that the collision is due to the early position or direction of "significant" agents. This type of cause matches the expectations of users who want to find the best actions to avoid the consequences with minimal material cost. In the first two scenarios, these significant agents are either in the front left of the flock or in its center. The causes involving the front left agents point to their angle (e.g. agents 6 or 17 in scenario 1, and 14 or 18 in scenario 2) while the causes involving centered agents point to their positions (e.g. agents 1 or 14 in scenario 1, and 22 and 26 in scenario 2). This suggests that the flock direction is controlled by front-left agents, while the density of the center of the flock contributes to its inertia. This is in agreement with the expectations that we mentioned in section

TABLE I: Explanations in the "One Flock" scenario.

Cause				Metrics		
Ent. Type	Ent. Id	Dim.	t	Cost	TTE	DL
Priorities: Cost - TTE - DL						
Agent	6 9	angle	0	37%	15	5
Agent	1 17	x angle	0	28%	15	6
Agent	14 17	x angle	0	22%	15	6
Priorities: TTE - Cost - DL						
Flock	3	x	14	824%	1	4
Flock	3	angle	14	2226%	1	4
Flock	3	agg.	14	4943%	1	4
Priorities: DL - Cost - TTE						
Flock	3	agg.	8	244%	7	4
Flock	3	x	14	824%	1	4
Flock	5	x	12	798%	3	4

TABLE II: Explanations for the "Flock Fusion" scenario.

Cause				Metrics		
Ent. Type	Ent. Id	Dim.	t	Cost	TTE	DL
Priorities: Cost - TTE - DL						
Agent	22 14	x angle	0	18%	16	6
Agent	18 26	angle y	0	22%	16	6
Agent	17 14	x angle	0	18%	16	6
Priorities: TTE - Cost - DL						
Flock	4	x	15	718%	1	4
Flock	4	y	15	718%	1	4
Flock	4	angle	15	1350%	1	4
Priorities: DL - Cost - TTE						
Flock	0	y	0	186%	16	4
Flock	0	agg.	0	308%	16	4
Flock	4	x	15	718%	1	4

TABLE III: Explanations in the "Free Movement" scenario.

Cause				Metrics		
Ent. Type	Ent. Id	Dim.	t	Cost	TTE	DL
Priorities: Cost - TTE - DL						
Agent	11	x	30	2%	5	4
Agent	4 39	x	30	8%	5	5
Agent	4 39	y x	30	8%	5	6
Priorities: TTE - Cost - DL						
Flock	17	x	34	80%	1	4
Flock	17	y	34	80%	1	4
Agent	16 4	angle	34	88%	1	5
Priorities: DL - Cost - TTE						
Agent	11	x	30	2%	5	4
Agent	11	x	21	7%	14	4
Agent	11	y	21	7%	14	4

V-B. In the third scenario, the flock that hits the obstacle follows very different dynamics. The significant agents are the ones from the colliding flock and another nearby one. This illustrates the difficulty of this scenario.

2) Time to Event: Prioritizing 'TTE' suggests that the collision is caused by the flock just before it happens. This type of cause matches the expectation of users who want the best actions to avoid the consequences at the last moment. In the first two scenarios, this appears clearly. In the third scenario, one of the top 3 causes is about two agents from the colliding flock, so they still convey the same idea.

3) Description length: Prioritizing 'DL' suggests descriptions specific to each scenario.

- For scenario 1, the collision is due to the flock trajectory (early flock aggregation and x position over time). The flock ID in the third cause differs due to one agent leaving the flock. The main flock is still the one pointed out despite a different flock ID.
- For scenario 2, two out of the top 3 causes suggest that the collision is due to the smaller flock before the merge. This follows the expected intuition about the scenario. Notably, the flock aggregation seems to point out the responsibility of the flock itself, and the flock angle seems to suggest that its trajectory is responsible (hence the fact that the flocks merged).
- For scenario 3, the collision is due to one specific agent's position (namely, agent 11, which is part of the flock that eventually collides with the obstacle). A single agent is clearly highlighted as an explanation for the collision, aligning with the "simplicity" condition that the 'DL' metrics approximate.

These types of causes match expectations from users who aim for a better understanding of the system by simplifying the dynamics in the given context.

VII. Discussion and future work

A. Final form of the explanation

In this work, explanations are selected as the top-n causes according to some configuration. This is a first step towards ideal causal explanations. However, our result tables, reporting raw causes, have low interpretability compared to how we described them. Explanations should not be the causes themselves but descriptions that only mention the type of the best causes. We did this work manually in section VI. In future work, we would like to perform this step automatically.

B. Limitations of our relevance scores

Our three metrics approximate different aspects of the notion of relevance. They can be improved or adapted for specific use cases. For instance, the cost function could be adaptive to the level of abstraction of the SCM. The intervention cost could also be made more useful with a model that allows for actions associated with utilities. The description length could be computed based on "user predicates" that would be more interpretable than the ones we used, e.g., identifiers. Finally, we could improve our account of normality by considering the normality of the cause itself (see section II-B5) and by using statistics to model the distribution of possible states.

C. Metric aggregation

In this work, we aggregated the metrics using a discretization of the values followed by a lexicographic sort with a given configuration. A similar result might have been reached with

a simpler method, such as a weighted average. However, to generate clear types of causes, one metric must be prioritized. To achieve this with weights, one value must prevail. This induces the risk that the other metrics are ignored. In our method, this is overcome by discretizing the scores. With weights, we need to tune the values very carefully so they are uneven enough to produce clear patterns but not too much so that the other metrics are not ignored. This process becomes more complex than our method, which takes both considerations into account.

D. Abstractions

We highlighted how different levels of abstraction can yield different relevant causes. In future work, we would like to automatically build the abstractions. We also did not consider time abstraction and chose a time sampling arbitrarily. In future work, we would like to apply the same method with various levels of time abstraction. Additionally, the abstractions used to create the causal variables should also be constructed following considerations of relevance. Filtering causal variables before the cause identification step could be an additional step in a future version of our method.

E. Unifying relevance

In future work, we would like to further investigate the use of Simplicity Theory (ST) [19] to model the relevance of causes. ST allows for the expression of various dimensions of relevance in terms of Kolmogorov complexity computations. Each dimension of relevance that we mentioned can be expressed as a description problem and measured in terms of the bits necessary to express this description. For instance, memorability [22] measures how simple it is to describe when an event occurs.

Computing descriptions requires a level of abstraction at which we conceptualize events. This notion is especially relevant in Complex Adaptive Systems (CAS) where emergence occurs and some levels of abstraction offer better descriptions of the system [2], [46]. In theory, modeling the user and context (e.g., through the query for explanation) can help determine the right level of abstraction to use to describe each part of the cause.

Kolmogorov complexity expresses all metrics in a common unit, i.e., bits of information. The relevance score would be immediately aggregated using a sum. The importance of one metric over another would be determined by the levels of abstraction associated with each relevance metric. For instance, if time requires a more precise description, it is arguably more relevant and would be described using more information, which would give it more importance in the aggregation.

F. Approximate cause identification

Identifying causes is an NP-complete problem. We used an approximate method that builds causes iteratively from smaller to larger ones. There are two types of approximations, controlled by two different parameters: the width and the depth of the search. When considering smaller widths, we encounter the risk of missing causes of smaller sizes. This exposes two types of errors. On the one hand, some causes may be missing. On the other hand, some causes may not be minimal, as the minimal set of variables may have been missed in a previous iteration of the algorithm. To mitigate these errors, one might use a larger value of the parameter controlling the width of the search. However, this would imply heavier computations. Another source of approximation is the limited depth of the search. However, this approximation filters out longer causes, which are very likely to get bad relevance rankings due to the 'DL' metrics. Therefore, this approximation has a limited impact on our method.

VIII. Conclusion

In this paper, we addressed the challenge of generating relevant causal explanations for Complex Adaptive Systems. While literature suggests that explanations are causes filtered based on their relevance to the context, most existing work lacks the flexibility to adapt to user expectations. This paper proposes a step toward better consideration of the contextual expectations of users when generating explanations. We identify related notions to assess the relevance of causes, such as simplicity, memorability, material cost, responsibility, and normality of the counterfactual world.

We propose a two-step methodology that includes these notions in a practical framework for explanations in CAS. Our methodology first identifies a wide range of causes, potentially at various levels of abstraction. Then, we measure the relevance of these causes based on the duration between the cause and the consequence, the cost of the intervention necessary to "cancel the consequence", and the description length of the cause. We finally flexibly aggregate them by discretizing their values and ranking them with a lexicographic sort.

We illustrated this method by explaining why a flock of agents collided with an obstacle. Our method, tuned in three different ways, manages to generate causal explanations that adapt to various user expectations and contexts. With a given configuration, the method produces context-specific causes matching the scenario's specificities. In other configurations, the method produces what could be expected by a user aiming to learn the best action to avoid the obstacle as cheaply or as late as possible.

Most of our implementations are approximations of abstract notions. We showed their efficiency in a concrete example and provided several suggestions for future work to match more closely the target concepts.

References

[1] J. H. Holland, "Complex adaptive systems," *Daedalus*, vol. 121, pp. 17–30, 1992.

[2] A. Diaconescu, D. King, K. Bellman, C. Landauer, and P. Nelson, "Towards systems that dynamically change and evaluate abstractions," in *13th Conference on Cognitive Situation Management (CogSIMA)*, Oct. 2023.

[3] P. Mellodge, A. Diaconescu, and L. J. Di Felice, " Timing configurations affect the macro-properties of multi-scale feedback systems ," in *IEEE Intl.Conf.on Autonomic Computing and Self-Organizing Systems (ACSOS)*, online, Oct. 2021, pp. 100–109.

[4] D. Hume, "A Treatise of Human Nature," in *Late Modern Philosophy: Essential Readings with Commentary*. Wiley-Blackwell, 2007.

[5] D. Lewis, "Causation," *The Journal of Philosophy*, vol. 70, no. 17, pp. 556–567, May 1974.

[6] J. Pearl, *Causality*, 2nd ed. Cambridge University Press, 2009.

[7] S. Reyd, A. Diaconescu, and J.-L. Dessalles, "CIRCE: A Scalable Methodology for Causal Explanations in Cyber-Physical Systems," in *2024 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, Sep. 2024, pp. 81–90.

[8] A. Rafieioskouei and B. Bonakdarpour, "Efficient Discovery of Actual Causality Using Abstraction Refinement," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 43, no. 11, pp. 4274–4285, Nov. 2024.

[9] A. Ibrahim, S. Rehwald, and A. Pretschner, "Efficient Checking of Actual Causality with SAT Solving," in *Engineering Secure and Dependable Software Systems*. IOS Press, 2019, pp. 241–255.

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

[11] Z. Wang, C. Huang, and X. Yao, "A Roadmap of Explainable Artificial Intelligence: Explain to Whom, When, What and How?" *ACM Trans. Auton. Adapt. Syst.*, vol. 19, no. 4, pp. 20:1–20:40, Nov. 2024.

[12] M. Hopkins, "Strategies for determining causes of events," in *Eighteenth National Conference on Artificial Intelligence*. USA: American Association for Artificial Intelligence, Jul. 2002, pp. 546–552.

[13] L. Albantakis, W. Marshall, E. Hoel, and G. Tononi, "What Caused What? A Quantitative Account of Actual Causation Using Dynamical Causal Networks," *Entropy*, vol. 21, no. 5, p. 459, May 2019.

[14] C. Chuck, S. Vaidyanathan, S. Giguere, A. Zhang, D. Jensen, and S. Niekum, "Automated Discovery of Functional Actual Causes in Complex Environments," Apr. 2024.

[15] C. W. Reynolds, *Flocks, herds, and schools: a distributed behavioral model*. New York, NY, USA: Association for Computing Machinery, 1998, p. 273–282.

[16] J. Y. Halpern and J. Pearl, "Causes and explanations: a structural-model approach: part i: causes," in *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, ser. UAI'01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, p. 194–202.

[17] J. Y. Halpern, "A modification of the Halpern-Pearl definition of causality," in *Proceedings of the 24th International Conference on Artificial Intelligence*, ser. IJCAI'15. Buenos Aires, Argentina: AAAI Press, Jul. 2015, pp. 3022–3033.

[18] Y.-L. Chou, C. Moreira, P. Bruza, C. Ouyang, and J. Jorge, "Counterfactuals and causability in explainable artificial intelligence: Theory, algorithms, and applications," *Information Fusion*, vol. 81, pp. 59–83, May 2022.

[19] J.-L. Dessalles, *Algorithmic Simplicity and Relevance*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 119–130.

[20] M. Li and P. Vitányi, *An Introduction to Kolmogorov Complexity and Its Applications*, ser. Texts in Computer Science. Cham: Springer International Publishing, 2019.

[21] P. Vitanyi and Ming Li, "Minimum description length induction, Bayesianism, and Kolmogorov complexity," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 446–464, Mar. 2000.

[22] É. Houzé, J.-L. Dessalles, A. Diaconescu, and D. Menga, "What Should I Notice? Using Algorithmic Information Theory to Evaluate the Memorability of Events in Smart Homes," *Entropy*, vol. 24, no. 3, p. 346, Mar. 2022.

[23] S. Reyd, A. Diaconescu, J.-L. Dessalles, and L. Esterle, "A Roadmap for Causality Research in Complex Adaptive Systems," in *2024 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C)*, Sep. 2024, pp. 35–40.

[24] H. Chockler and J. Y. Halpern, "Responsibility and Blame: A Structural-Model Approach," *Journal of Artificial Intelligence Research*, vol. 22, pp. 93–115, Oct. 2004.

[25] D. Kahneman and D. T. Miller, "Norm theory: Comparing reality to its alternatives." *Psychological review*, vol. 93, no. 2, p. 136, 1986.

[26] T. F. Icard, J. F. Kominsky, and J. Knobe, "Normality and actual causal strength," *Cognition*, vol. 161, pp. 80–93, 2017.

[27] J. Y. Halpern and C. Hitchcock, "Graded Causation and Defaults," *The British Journal for the Philosophy of Science*, vol. 66, no. 2, pp. 413–457, Jun. 2015.

[28] M. J. Vowels, N. C. Camgoz, and R. Bowden, "D'ya Like DAGs? A Survey on Structure Learning and Causal Discovery," *ACM Computing Surveys*, vol. 55, no. 4, pp. 82:1–82:36, Nov. 2022.

[29] B. Scholkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, "Toward Causal Representation Learning," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 612–634, May 2021.

[30] *, "Practical causation analysis: A search algorithm to identify actual causes." To be submitted, Anonymous for the blind submission.

[31] G. Sileno and J.-L. Dessalles, "Qualifying Causes as Pertinent," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2018, pp. 2491–2496.

[32] J. Y. Halpern and J. Pearl, "Causes and Explanations: A Structural-Model Approach. Part II: Explanations," *The British Journal for the Philosophy of Science*, Dec. 2005.

[33] F. Ziesche, V. Klös, and S. Glesner, "Anomaly Detection and Classification to enable Self-Explainability of Autonomous Systems," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Feb. 2021, pp. 1304–1309.

[34] É. Houzé, A. Diaconescu, J.-L. Dessalles, and D. Menga, "A generic and modular reference architecture for self-explainable smart homes," in *2022 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, Sep. 2022, pp. 101–110.

[35] É. Houzé, J.-L. Dessalles, A. Diaconescu, D. Menga, and M. Schumann, "A Decentralized Explanatory System for Intelligent Cyber-Physical Systems," in *Intelligent Systems and Applications*, ser. Lecture Notes in Networks and Systems, K. Arai, Ed. Springer International Publishing, 2022, pp. 719–738.

[36] D. Minh, H. X. Wang, Y. F. Li, and T. N. Nguyen, "Explainable artificial intelligence: A comprehensive review," *Artificial Intelligence Review*, vol. 55, no. 5, pp. 3503–3568, Jun. 2022.

[37] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018.

[38] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, Aug. 2016, pp. 1135–1144.

[39] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[40] M. Blumreiter, J. Greenyer, F. J. Chiyah Garcia, V. Klös, M. Schwammberger, C. Sommer, A. Vogelsang, and A. Wortmann, "Towards Self-Explainable Cyber-Physical Systems," in *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, Sep. 2019, pp. 543–548.

[41] M. Camilli, R. Mirandola, and P. Scandurra, "XSA: eXplainable Self-Adaptation," in *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, ser. ASE '22. New York, NY, USA: Association for Computing Machinery, Jan. 2023, pp. 1–5.

[42] M. Sadeghi, L. Herbold, M. Unterbusch, and A. Vogelsang, "SmartEx: A Framework for Generating User-Centric Explanations in Smart Environments," in *2024 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Mar. 2024, pp. 106–113.

[43] L. Herbold, M. Sadeghi, and A. Vogelsang, "Generating Context-Aware Contrastive Explanations in Rule-based Systems," in *Proceedings of the 2024 Workshop on Explainability Engineering*, ser. ExEn '24. New York, NY, USA: Association for Computing Machinery, Jul. 2024, pp. 8–14.

[44] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, May 2011.

[45] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1996, pp. 226–231.

[46] B. Yuan, J. Zhang, A. Lyu, J. Wu, Z. Wang, M. Yang, K. Liu, M. Mou, and P. Cui, "Emergence and causality in complex systems: a survey of causal emergence and related quantitative studies," *Entropy*, vol. 26, no. 2, p. 108, 2024.