An Information-Flow Perspective on Explainability Requirements: Specification and Verification

Bernd Finkbeiner[1] , **Hadar Frenkel**[2] , **Julian Siber**[1]

[1]CISPA Helmholtz Center for Information Security, Saarbrücken, Germany
[2]Bar-Ilan University, Ramat Gan, Israel
finkbeiner@cispa.de, hadar.frenkel@biu.ac.il, julian.siber@cispa.de

Abstract

Explainable systems expose information about why certain observed effects are happening to the agents interacting with them. We argue that this constitutes a positive flow of information that needs to be specified, verified, and balanced against negative information flow that may, e.g., violate privacy guarantees. Since both explainability and privacy require reasoning about knowledge, we tackle these tasks with epistemic temporal logic extended with quantification over counterfactual causes. This allows us to specify that a multi-agent system exposes enough information such that agents acquire knowledge on why some effect occurred. We show how this principle can be used to specify explainability as a system-level requirement and provide an algorithm for checking finite-state models against such specifications. We present a prototype implementation of the algorithm and evaluate it on several benchmarks, illustrating how our approach distinguishes between explainable and unexplainable systems, and how it allows to pose additional privacy requirements.

1 Introduction

Contemporary autonomous systems are increasingly complex and opaque, yet also deployed in consequential applications such as hiring (Dastin 2018), healthcare (Thomas and Ravi 2019), and criminal sentencing (Angwin et al. 2016). This tension has led to an extensive inquiry into methods that provide explanations for the behavior of these systems, such that agents interacting with, e.g., a hiring system may know *why* their application is rejected (Mersha et al. 2024).

Although an explanation may provide critical and actionable recourse to one agent, it may also reveal private information about another (Nguyen et al. 2025). For instance, a rejected job application may be explained by the applicant's pay requirement being over a certain threshold, but if the applicant observes a sufficiently similar application being accepted, they can infer the future salary of the other agent. This line of reasoning leads to an inherent tradeoff between explainability and privacy, which we study in this paper.

Privacy requirements are commonly formalized as *information-flow policies* (Goguen and Meseguer 1982; Kozyri, Chong, and Myers 2022), i.e., a system must not only restrict direct access to private information but also restrict propagation of this information through indirect channels. We approach the formalization of explainability requirements from the same perspective and express them as end-to-end information-flow policies in a specification language for multi-agent systems. On a high level, such an explainability requirement defines the necessary flow of information in the following way: It specifies what needs to be explained (the *explanandum*), how it needs to be explained (the *explanans*), and when it needs to be explained, to whom. We propose a verification algorithm that can then check whether a given system ensures sufficient flow of information to meet a requirement in our specification language. With the same language, we can also express privacy requirements, such that we can similarly check that explainability does not come at the expense of security.

Our specification language extends epistemic temporal logic (Fagin et al. 1995). Epistemic and temporal logics have been popular frameworks for studying information-flow security (Halpern and O'Neill 2008; Balliu, Dam, and Guernic 2011; Clarkson et al. 2014; Coenen et al. 2019). This inspires us to study the flow of information in explainable systems through the same lens. We build on the popular framework of counterfactual explanations (Halpern and Pearl 2005; Wachter, Mittelstadt, and Russell 2018) and instrument our logic with operators to reason about temporal causes (Finkbeiner et al. 2024). These are based on a temporal variant of actual causation (Halpern 2016), which uses counterfactual reasoning (Lewis 1973) to explain a given execution and has received significant attention in the literature on explainable AI (Miller 2019). A temporal cause can for instance be described symbolically by the formula $\bullet a_1 \wedge \blacklozenge a_2$, which means that action a_1 at the previous time point and action a_2 at any earlier time point have jointly caused some effect: The cause needs to be satisfied and describe the minimal changes necessary to obtain a counterfactual execution where the effect does not happen.

The combination of counterfactual, epistemic and temporal reasoning allows us to express explainability requirements of the following form:

$$\square\left(\psi \rightarrow \exists X.\, \mathtt{K}_a(X \overset{Act(a)}{\rightsquigarrow} \psi)\right) ,$$

which states that the explanandum ψ is explainable to agent a whenever ψ occurs, via the temporal cause X serving as explanans. The cause X is constrained to reason only over the actions $Act(a)$ of agent a, which is why we term this requirement *Internal Causal Explainability (ICE)*. The requirement uses the temporal operator $\square$ to enforce its con-

straint on every time point of an execution, and the epistemic operator $\mathtt{K}_a$ to express its key epistemic component: The semantics of this latter operator require that the same property X causes ψ on all executions that are indistinguishable to agent a, which means that agent a has acquired knowledge of the associated causal dependency.

Contributions and Outline. We give a detailed example to illustrate how ICE encodes explainability in an information-flow sense in Section 3, after establishing necessary preliminaries in Section 2. The motivating example highlights how explainability requirements, such as ICE, require tradeoffs when juxtaposed with privacy requirements placed on the same system. We then introduce the formal details of our logic in Section 4, where we discuss other explainability requirements beside ICE, and outline an algorithm to verify whether a given system satisfies a requirement specified in our logic. The main challenge for the algorithm is the second-order quantification that ranges over sets of traces, as related logics with unrestricted quantifiers of this kind cannot be verified automatically (Beutner et al. 2023a). We show how to exploit the fact that causes are uniquely determined to encode the second-order quantifiers over sets of traces into the decidable satisfiability problem of a temporal logic with first-order quantification over atomic propositions only. We then report on experiments with a prototype implementation of our approach in Section 5. Our prototype can verify both explainability and privacy requirements as introduced throughout the paper, on multi-agent systems with up to several thousand states. We use classic games and an auction system for these experiments. Last, we discuss related work in Section 6 and close with a short summary and outlook on future work in Section 7.

2 Preliminaries

We recall the formal background on transition systems as models of multi-agent systems, temporal logics for specifying system requirements, and temporal causality for defining counterfactual dependencies between temporal properties.

Multi-Agent Systems. We consider *transition systems* as the fundamental model of the logics we will study in this paper. A transition system is a tuple $\mathcal{T} = (S, S_0, \Delta, \mathrm{AP}, \Lambda)$, where S is a finite set of *states*, S_0 is a set of *initial states*, $\Delta : S \mapsto 2^S$ is a *transition function* such that $\Delta(s) \neq \emptyset$ for all states $s \in S$, AP is a set of *atomic propositions*, and $\Lambda : S \times S \mapsto 2^{\mathrm{AP}}$ is a *labeling function* marking edges with atomic propositions. Executions of a system are modeled as follows. A *path* $\rho = \rho[0]\rho[1] \ldots \in S^\omega$ of $\mathcal{T}$ is an infinite sequence of states following the transition function: $\rho[i+1] \in \Delta(\rho[i])$ for all *time points* $i \in \mathbb{N}$. The *trace* $\pi = \pi[0]\pi[1] \ldots \in (2^{\mathrm{AP}})^\omega$ of a path ρ is the sequence of corresponding labels, i.e., we have $\pi[i] = \Lambda(\rho[i], \rho[i+1])$ for all $i \in \mathbb{N}$. Let $\Pi(\mathcal{T})$ denote the set of traces of initial paths, i.e., of ρ such that $\rho[0] \in S_0$. For some trace π, $\pi[0, n] \in S^*$ is its *prefix* of length $n+1$. For two traces $\pi, \pi \in S^\omega$ and $A \subseteq \mathrm{AP}$ we write $\pi =_A \pi'$ if $\pi[i] \cap A = \pi'[i] \cap A$ for all time points $i \in \mathbb{N}$. We model a multi-agent system as an *extended transition system* $\mathcal{E} = (\mathcal{T}, \Omega, Act)$ that includes an *observation map* $\Omega : \mathrm{AG} \mapsto 2^{\mathrm{AP}}$ to reason about the observations of a set of agents AG and an *action map* $Act : \mathrm{AG} \mapsto 2^{\mathrm{AP}}$ to reason about their controllable actions. We will use the shorthand Act for the image of AG under Act, i.e., the set of all actions. We call every atomic proposition that is not an action a system *output*. For some agent $a \in \mathrm{AG}$, $\Omega(a)$ describes the set of atomic propositions that are observable to agent a, and $Act(a)$ describes which actions are controllable by a. We assume $Act(a) \subseteq \Omega(a)$. For some trace π of $\mathcal{T}$, $\Omega_a(\pi) \in (2^{\mathrm{AP}})^\omega$ are the partial observations of a along the trace: $\Omega_a(\pi)[i] = \pi[i] \cap \Omega(a)$. We further assume that all actions are possible from every state (although they may have no effect): For all $s \in S$ and $A \subseteq Act$, there is an $s' \in S$ such that $s' \in \Delta(s)$ and $A = \Lambda(s, s') \cap Act$. We say that a multi-agent system is *deterministic* if there exists exactly one such s' for all $s \in S$ and $A \subseteq Act$. The set of traces of $\mathcal{E} = (\mathcal{T}, \Omega)$ is denoted $\Pi(\mathcal{E}) = \Pi(\mathcal{T})$.

Example 1. *Consider the multi-agent system $\mathcal{A}_{explain}$ shown in Figure 1. It models an explainable auction between three bidders. Nodes and edges depict states and the transition function, respectively. The single initial state is indicated through an incoming edge without a source state. For brevity, edge-labels use symbolic notation for actions (left of the bar) and explicit sets for the outputs (right of the bar). If there are no outputs, we only depict the action constraints. A trace of this system is $\pi = \{o, b_1, b_2, e\}(\{o, b_1, b_2\})^\omega$, which corresponds to path $\rho_1 = \mathrm{init}\,(\mathrm{win}_1)^\omega$ or $\rho_2 = \mathrm{init}\,(\mathrm{win}_2)^\omega$, since the set $\{o, b_1, b_2\}$ satisfies $o \wedge b_1$ as well as $o \wedge b_2$. The ω-superscript denotes infinite repetition of the subsequence.*

Temporal Logics. The basis of our logic is the epistemic temporal logic KLTL, which extends *Linear Temporal Logic (LTL)* with a knowledge modality. We defer KLTL and start with a definition of LTL (Pnueli 1977). We include past operators, which do not increase expressive power (Lichtenstein, Pnueli, and Zuck 1985), but will make some formulas more readable. The syntax is given as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \bigcirc\varphi \mid \varphi\,\mathtt{U}\,\varphi \mid \bullet\varphi \mid \varphi\,\mathtt{U}^-\,\varphi \ , \quad (1)$$

where $p \in \mathrm{AP}$ is an atomic proposition. Additionally, LTL includes the following derived operators: Boolean constants ($true$, $false$) and connectives ($\vee$, $\rightarrow$, $\leftrightarrow$), the temporal operator 'Eventually' ($\diamondsuit\varphi \equiv true\,\mathtt{U}\,\varphi$) as well as its dual, 'Globally' ($\square\varphi \equiv \neg\diamondsuit\neg\varphi$), and the derived past operators 'Once' ($\blacklozenge\varphi \equiv true\,\mathtt{U}^-\,\varphi$) and 'Historically' ($\blacksquare\varphi \equiv \neg\blacklozenge\neg\varphi$). The semantics of an LTL formula φ is defined with respect to a trace π and a time point i as follows:

$$\begin{aligned}
&\pi, i \vDash p && \text{iff } p \in \pi[i],\\
&\pi, i \vDash \neg\varphi && \text{iff } \pi, i \nvDash \varphi,\\
&\pi, i \vDash \varphi_1 \vee \varphi_2 && \text{iff } \pi, i \vDash \varphi_1 \text{ or } \pi, i \vDash \varphi_2,\\
&\pi, i \vDash \bigcirc\varphi && \text{iff } \pi, i+1 \vDash \varphi,\\
&\pi, i \vDash \bullet\varphi && \text{iff } \pi, i-1 \vDash \varphi \text{ and } i > 0,\\
&\pi, i \vDash \varphi_1\,\mathtt{U}\,\varphi_2 && \text{iff } \exists k \geq i : \pi, k \vDash \varphi_2 \text{ and}\\
& && \quad \forall i \leq j < k : \pi, j \vDash \varphi_1,\\
&\pi, i \vDash \varphi_1\,\mathtt{U}^-\,\varphi_2 && \text{iff } \exists g \leq i : \pi, g \vDash \varphi_2 \text{ and}\\
& && \quad \forall g \leq h < i : \pi, h \vDash \varphi_1.
\end{aligned}$$

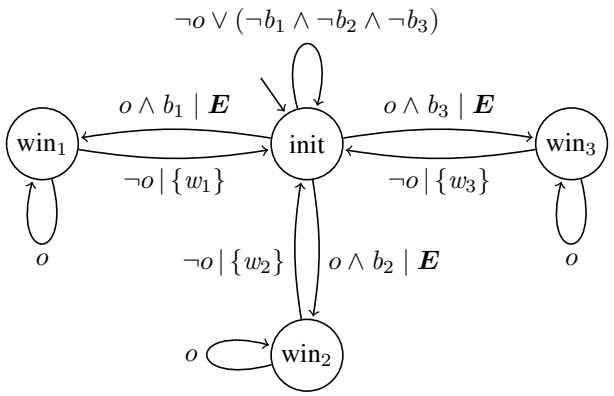

Figure 1: A multi-agent system modeling a Dutch auction with three agents bidding through actions b_1, b_2, b_3 and one auctioneer opening and closing the auction with action o. The winner is announced through outputs w_1, w_2, w_3. We consider three versions of the auction in this paper. In the first version $\mathcal{A}_{blind}$ an agent i observes only their own action, output, and the auctioneer: b_i, w_i, o. In the second version $\mathcal{A}_{public}$ an agent i additionally observes all actions Act. For both of these versions, the explanatory output $\boldsymbol{E}$ is empty, while we have $\boldsymbol{E} = \{e\}$ for the third version $\mathcal{A}_{explain}$, which otherwise is like $\mathcal{A}_{blind}$ except e is observable by all agents.

We call the combination of trace and time point an *anchor point*. System-level satisfaction is based on a universal application of the trace semantics: $\mathcal{T}$ *satisfies* φ, denoted by $\mathcal{T} \vDash \varphi$, iff for all traces $\pi \in \Pi(\mathcal{T}) : \pi, 0 \vDash \varphi$. The *language* $\mathcal{L}(\varphi)$ is the set of all traces satisfying the LTL formula φ.

The *epistemic temporal logic KLTL* (Fagin et al. 1995) extends LTL with a knowledge modality that expresses the knowledge of an agent $a \in \text{AG}$, i.e., it adds the rule $\mathtt{K}_a\,\varphi$ to Grammar 1. The semantics of a KLTL formula is as for LTL, but additionally refers to an extended transition system $\mathcal{E} = (\mathcal{T}, \Omega)$. For the epistemic operator $\mathtt{K}_a$, we have:

$$\mathcal{E}, \pi, i \vDash \mathtt{K}_a\,\varphi \text{ iff } \forall \pi' \in \Pi(\mathcal{T}) : (\Omega_a(\pi)[0,i] = \Omega_a(\pi')[0,i]) \rightarrow \mathcal{E}, \pi', i \vDash \varphi \ .$$

Hence, agent a has knowledge of some property φ on a trace π at point i, expressed through the formula $\mathtt{K}_a\,\varphi$, if this property holds on all traces that are indistinguishable for a from π up to this point. This corresponds to the so-called synchronous perfect recall semantics (van der Meyden and Shilov 1999; Halpern, van der Meyden, and Vardi 2004), since the agents can distinguish prefixes of different length and based on divergence at any point in the past. Similar to LTL, we have $\mathcal{E} \vDash \varphi$ iff for all traces $\pi \in \Pi(\mathcal{T}) : \mathcal{E}, \pi, 0 \vDash \varphi$.

Temporal Causality. Since multi-agent systems are sequential processes where the timing of actions is causally relevant, we adapt temporal causality (Finkbeiner et al. 2024) to express causal dependencies between temporal properties. This utilizes counterfactual reasoning based on similarity (Lewis 1973). Similarity is defined with a trace-wide extension of the *symmetric difference* $A \oplus B = (A \setminus B) \cup (B \setminus A)$ of two sets A, B. For two traces $\pi, \pi' \in (2^{\text{AP}})^\omega$ and some $A \subseteq \text{AP}$, we define $\pi \oplus_A \pi' = \{(a, i) \in A \times \mathbb{N} \mid a \in \pi[i] \oplus \pi'[i]\}$. For three traces $\pi, \pi', \pi'' \in (2^{\text{AP}})^\omega$ and $A \subseteq Act$ we say π is at least as similar to π' as π'' over A, denoted with $\pi \leq^A_{\pi'} \pi''$, iff $(\pi \oplus_A \pi') \subseteq (\pi'' \oplus_A \pi')$.

We now have everything we need to define temporal causality: The *cause* for some temporal property ψ at time point i on a trace π of system $\mathcal{E} = (\mathcal{T}, \Omega)$ with respect to $A \subseteq Act$ is the following temporal property:

$$Cause(\psi, \pi, i, A) = \{\, \pi' \in (2^A)^\omega \mid \forall \pi'' \in \Pi(\mathcal{T}). \; (\pi'' \leq^A_\pi \pi' \wedge \pi =_{(Act \setminus A)} \pi'') \rightarrow \mathcal{E}, \pi'', i \vDash \psi \,\} \ .$$

A temporal cause describes the largest set of action sequences satisfying the effect ψ that is downward closed in $(\Pi(\mathcal{T}), \leq^A_\pi)$, i.e., all sequences with traces that satisfy the effect ψ such that all at least as similar traces also satisfy ψ. Causes capture semantically what actions of a trace need to be changed to negate the effect. The parameter A allows to constrain which actions can be changed and which actions are fixed between the traces. A cause may be described symbolically as the language of a temporal logic formula.

Example 2. *Consider again system $\mathcal{A}_{explain}$ in Figure 1, effect $\psi = \Diamond w_1$ and trace $\pi = \{o, b_1, e\}\{o\}\{o, b_1\}\{w_1\}\{\}^\omega$. We have that $Cause(\psi, \pi, 0, \{b_1\})$ is the language $\mathcal{L}(b_1 \vee \bigcirc\bigcirc b_1)$. This is because all traces that have action b_1 at either the first or third point and are equal to π on all other actions satisfy the effect ψ. Other traces that satisfy ψ such as $\pi' = \{o\}\{o, b_1, e\}\{o\}\{w_1\}\{\}^\omega$ are less similar to π with respect to $\{b_1\}$ than $\{o\}\{o\}\{o\}\{\}^\omega$, which does not satisfy ψ, and hence these are not included in $Cause(\psi, \pi, 0, \{b_1\})$.*

3 From Explanations to Explainability

In this section, we give a high-level overview of this work. We particularly focus on delimiting the concepts of (individual) *explanations*, the information flow-based system requirement we call *explainability*, and how our work allows to identify tradeoffs between explainability and privacy.

We illustrate these concepts with the multi-agent systems modeling several versions of a *Dutch auction* depicted in Figure 1. In a Dutch auction, there are a number of bidders that compete for a resource, and an auctioneer that opens and closes the auction. The auctioneer sets an initial price and decrements the price in every time step until reaching a lower limit price. Participants can place their bids at any time point, with the first bidder in a given auction cycle winning the resource. The systems are nondeterministic for the case that multiple agents place the first bid in a given round. The different versions of the auction differ with respect to the explainability and privacy guarantees that they provide, as we outline in the following.

3.1 Explanations

In case a bidder does not win an auction, they may be interested in an explanation for this outcome. We consider explanations that answer counterfactual queries, e.g.: What did the agent need to do differently to obtain the desired outcome? The answer to such a query may include the temporal behavior of the agent, e.g., bidding *earlier*. Hence, we use temporal causes (cf. Section 2) as the semantic content of an

answer, i.e., as explanantia. Consider a trace π of the system $\mathcal{A}_{blind}$ where Bidder 1 loses the first cycle to Bidder 2.

Auctioneer:	$\{o\}$	$\{o\}$	$\{o\}$	$\{\}$	$\{\}^\omega$
Bidder 1:	$\{\}$	$\{\}$	$\{b_1\}$	$\{\}$	$\{\}^\omega$
Bidder 2:	$\{\}$	$\{b_2\}$	$\{\}$	$\{w_2\}$	$\{\}^\omega$
Bidder 3:	$\{\}$	$\{\}$	$\{b_3\}$	$\{\}$	$\{\}^\omega$

The auction cycle is open while the auctioneer chooses the opening action o, in this way modeling them setting the initial and limit prices. During these first three time points, Bidder 2 bids at the second time point with action b_2 and the remaining agents bid one step later with actions b_1 and b_3.

If we restrict counterfactual actions to range only over the actions of Bidder 1 themselves, it is easy to see that bidding at either the first or second time point would have allowed them to win the auction cycle, although the latter only through an advantageous resolution of nondeterminism on the action sequence. We express the counterfactual dependency between $\neg w_1$ and its cause as follows:

$$\varphi_{cf} = (\bullet^2(\neg b_1 \wedge \bullet \neg b_1)) \overset{\{b_1\}}{\rightsquigarrow} \neg w_1 \ .$$

The formula states that the minimal changes to the execution to flip the truth of outcome $\neg w_1$ at the fourth time point with all actions fixed but $\{b_1\}$ also flip the truth of $\bullet^2(\neg b_1 \wedge \bullet \neg b_1)$, i.e., Bidder 1 not bidding at two and three time points before. We use an i superscript to shorten sequences of length i of the same operator.

The goal of an explanation for $\neg w_1$ is to change the epistemic state of Bidder 1 from not knowing φ_{cf} to knowing φ_{cf}, i.e., we want to ensure that $\mathtt{K}_{\text{Bidder 1}}(\varphi_{cf})$ holds. This knowledge depends on the observations Bidder 1 can make during the execution of π: They have knowledge only of formulas that hold on all indistinguishable executions at this time point. In the blind auction $\mathcal{A}_{blind}$ where Bidder 1 can only see their own bids and the auctioneer, the execution of π is indistinguishable from the execution of π' where Bidder 2 bids with the following sequence.

Bidder 2: $\{b_2\} \mid \{\} \mid \{\} \mid \{w_2\} \mid \{\}^\omega$

On this execution, the cause for $\neg w_1$ at the fourth time point is reduced to $\bullet^3 \neg b_1$. Hence, Bidder 1 does not know the cause for $\neg w_1$ in $\mathcal{A}_{blind}$ at this point, i.e., $\mathtt{K}_{\text{Bidder 1}}(\varphi_{cf})$ does not hold. What can the auction system do such that Bidder 1 gains knowledge of the cause for $\neg w_1$? The key lies in providing additional observations to the agent that serve as *explanations*. One possible set of explanatory observations for $\neg w_1$ is $\{b_2, b_3\}$, i.e., the bidding actions of all other agents. In system $\mathcal{A}_{public}$ where they are observable, Bidder 1 can distinguish π from all other traces and hence $\mathtt{K}_{\text{Bidder 1}}(\varphi_{cf})$ holds on π at the fourth time point.

3.2 Explainability

The previous section has outlined how explanations can transport knowledge about counterfactual causes at a specific time point in a given execution. To go from explanations to explainable systems we need to define under which circumstances this knowledge should be available to which agents. In the Dutch auction system, we may for instance require that Bidder 1 knows the cause for a loss whenever the auction closes. We can again use temporal operators to express these timing requirements, as we already did in the previous section to describe that the temporal behavior of Bidder 1 is a causal antecedent for their loss at the fourth time point. A complicating factor is that a given effect can have an arbitrary number of causes at different time points. For instance, shifting π by duplicating its first time point includes an additional action in the cause for the loss of Bidder 1, which can be spun arbitrarily further. Hence, it does not suffice to use a finite number of explicit causal antecedents in a system-wide explainability requirement. We solve this by extending the logic with second-order quantifiers that allow to quantify over sets of traces. Such a set can be instantiated with different causes at different time points. Our considerations on timing and cause quantification result in the following requirement for the Dutch auction systems:

$$\square\big((\neg w_1 \wedge \neg o \wedge \bullet o) \rightarrow \exists X.\, \mathtt{K}_{\text{Bidder 1}}(X \overset{\{b_1\}}{\rightsquigarrow} \neg w_1)\big) \ .$$

The requirement states that at all future time points (enforced through the temporal operator $\square$), whenever the auction was open in the previous time point, is now closed, and Bidder 1 has not won the cycle, then there is a property X such that Bidder 1 knows that X is the cause for their loss, i.e., X is the cause on all traces that are indistinguishable for Bidder 1. The cause X is constrained to action b_1 of Bidder 1. It is an instance of ICE as introduced in Section 1.

It is easy to see that the auction system $\mathcal{A}_{blind}$ does not satisfy ICE, since the fourth time point on execution π as discussed in Section 3.1 is a counterexample to the system-wide requirement. In $\mathcal{A}_{public}$, Bidder 1 observes everything about an execution except how nondeterminism is resolved when multiple agents bid first simultaneously. In these instances, the empty set is a valid causal antecedent on all indistinguishable traces and hence X can be instantiated with it. This effectively means that Bidder 1 either knows which actions would have resulted in them winning the auction, or knows that their actions were already optimal and their loss is attributable to unmodeled actions such as randomness. Hence, we have that $\mathcal{A}_{public}$ satisfies ICE.

3.3 Balancing Explainability and Privacy

Although $\mathcal{A}_{public}$ satisfies the explainability requirement, this comes at the cost of exposing all actions of the other agents. This is clearly unsatisfactory, as systems may place privacy requirements alongside explainability. For instance, we may require that Bidder 1 never knows whether Bidder 2 places a bid at a given time point. We can express this system-wide requirement utilizing as $\square\big(\neg \mathtt{K}_{\text{Bidder 1}}(b_2)\big)$, which formally requires that Bidder 1 should never be able to distinguish a given execution from another where the value of b_2 is flipped. We call this parametric privacy notion b_2-*privacy*. In terms of privacy, the desirability of $\mathcal{A}_{public}$ and $\mathcal{A}_{blind}$ is now exactly opposite as for explainability: $\mathcal{A}_{public}$ clearly does not satisfy it as Bidder 1 can directly observe b_2. $\mathcal{A}_{blind}$ does satisfy it because Bidder 1 cannot distinguish between the actions of Bidders 2 and 3.

It turns out that making the set $\{b_2, b_3\}$ observable was too impetuous and a smaller set would have been sufficient

without sacrificing privacy: System $\mathcal{A}_{explain}$ adds the output e, which gets broadcast to all agents whenever the first bid of an open auction comes in. On the one hand, this still provides the information flow required to identify the cause for a loss of Bidder 1, such that ICE is satisfied by the system. On the other hand, this hides who placed the highest bid, such that the system also satisfies b_2-privacy.

4 A Logic for Explainability Requirements

In this section, we dig deeper into the formal details of our logic for expressing explainability requirements, which we call YLTL^2. We start with outlining its syntax and semantics in Section 4.1. We then illustrate how the logic can be used to specify a number of different explainability requirements in Section 4.2. Last, we describe a model-checking algorithm for the logic in Section 4.3.

4.1 Syntax and Semantics of YLTL^2

YLTL^2 is an extension of KLTL, i.e., linear temporal logic with the knowledge modality $\mathtt{K}_a$. Hence, the syntax and semantics of all shared operators are as described in Section 2.

Syntax. YLTL^2 extends KLTL with second-order quantification over sets of traces and allows these sets to be used in *causal predicates* $X \overset{A}{\rightsquigarrow} \varphi$, which require X to be the cause for φ over a set of actions A at the current time point of a given trace. We assume a set of second-order variables SO be given with the set of atomic propositions AP and agents AG. The syntax of YLTL^2 is as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \bigcirc\varphi \mid \varphi \,\mathtt{U}\, \varphi \mid \bullet\varphi \mid \varphi \,\mathtt{U}^{-}\, \varphi \mid \mathtt{K}_a\,\varphi \mid \exists X.\,\varphi \mid X \overset{A}{\rightsquigarrow} \varphi \ ,$$

where $p \in$ AP is an atomic proposition, $a \in$ AG is an agent, $X \in$ SO is a second-order variable, and $A \subseteq$ AP is a subset of atomic propositions. YLTL^2 includes the same derived operators as LTL (cf. Section 2), as well as universal second-order quantification derived as $\forall X.\,\varphi \equiv \neg\exists X.\,\neg\varphi$. We say a YLTL^2 formula is *well-formed* iff all of its subformulas $X \overset{A}{\rightsquigarrow} \varphi$ are in the scope of an existential quantifier $\exists X$, that is, all second-order variables are bound to a quantifier.

Semantics. The semantics of all operators shared between YLTL^2 and KLTL are as defined in Section 2, but they additionally refer to a second-order assignment $\Theta : \text{SO} \mapsto \mathbb{P}(\Sigma^\omega)$, which is a mapping from second-order variables to sets of traces. This assignment plays a central part in the semantics of the new operators in YLTL^2:

$$\mathcal{E}, \pi, i, \Theta \vDash \exists X.\,\varphi \quad \text{iff } \exists Y \subseteq (2^{\text{AP}})^\omega \text{ s.t. } \mathcal{E}, \pi, i, \Theta[X \mapsto Y] \vDash \varphi,$$

$$\mathcal{E}, \pi, i, \Theta \vDash X \overset{A}{\rightsquigarrow} \varphi \text{ iff } \Theta(X) = \mathit{Cause}(\varphi, \pi, i, A).$$

Hence, the semantics of the second-order quantification $\exists X.\,\varphi$ requires that there exists a set of traces, assigned to the second-order variable X, such that the subformula φ is satisfied. In the subformula, this second-order variable may be used in an arbitrary number of causal predicates $X \overset{A}{\rightsquigarrow} \varphi$ and it has to qualify as a cause in all of them. This essentially allows to succinctly specify equality of causes at an arbitrary (even infinite) number of anchor points in the scope of a single quantifier. For an extended transition system $\mathcal{E}$ and a well-formed YLTL^2 formula φ, we have $\mathcal{E} \vDash \varphi$ iff for all traces $\pi \in \Pi(\mathcal{T}) : \mathcal{E}, \pi, 0, \mathfrak{E} \vDash \varphi$, where $\mathfrak{E}$ denotes an empty second-order assignment that maps all variables to $\bot$. The YLTL^2 *model-checking* problem is to decide whether $\mathcal{E} \vDash \varphi$ for such an $\mathcal{E}$ and φ.

4.2 Formalizing Explainability Requirements

Besides ICE, YLTL^2 can be used to define other explainability requirements. We establish some results on entailment between these requirements and other notions.

Outcomes may not only be explainable to an agent by their own actions, but also through the actions of other agents: For instance, Bidder 1 losing an auction cycle also depends causally on the actions of the other bidders. Information flow about such causes can be specified as follows.

Definition 1 (External Causal Explainability)**.** *An effect ψ is explainable for agent a according to External Causal Explainability (ECE) in some system $\mathcal{E}$, iff $\mathcal{E}$ satisfies the following property:*

$$\square\big(\psi \rightarrow \exists X.\,\mathtt{K}_a(X \overset{Act \setminus Act(a)}{\rightsquigarrow} \psi)\big) \ .$$

Hence, ECE requires simply changing the actions that a cause must range over from $Act(a)$, i.e., the actions of agent a, to the actions of all other agents: $Act \setminus Act(a)$.

Example 3. *Consider the auction system $\mathcal{A}_{explain}$ (cf. Figure 1) and the following trace:*

$$\{o\} \mid \{o, b_2, e\} \mid \{o, b_2, b_3\} \mid \{o, b_1\} \mid \{w_2\} \mid \{\}^\omega$$

The temporal cause for $\neg w_1$ at the fifth time point is now composed of Bidder 2 bidding at the second and third time point and Bidder 3 bidding at the third time point:

$$(\bullet^2(b_2 \vee b_3 \vee \bullet b_2)) \overset{\{b_2, b_3, o\}}{\rightsquigarrow} \neg w_1 \ .$$

Next, the combination of ICE and ECE requires full knowledge about any causal dependencies.

Definition 2 (Full Causal Explainability)**.** *An effect ψ is explainable for agent a according to Full Causal Explainability (FCE) in a system $\mathcal{E}$, iff $\mathcal{E}$ satisfies the following property:*

$$\square\big(\psi \rightarrow \exists X.\,\mathtt{K}_a(X \overset{Act}{\rightsquigarrow} \psi)\big) \ .$$

Intuitively, adding additional atomic propositions to the causal predicates effectively requires more information to flow to agent a. We can show formally this formally.

Proposition 1. *ICE and ECE are weaker criteria than FCE, i.e., we have for all systems $\mathcal{E}$ that $\mathcal{E} \vDash$ FCE implies $\mathcal{E} \vDash$ ICE and $\mathcal{E} \vDash$ ECE.*

Proof. It suffices to show that for any sets $A \subseteq B$ and usual semantic parameters, (1) $\mathcal{E}, \pi, i, \Theta \vDash \exists X.\,\mathtt{K}_a(X \overset{B}{\rightsquigarrow} \psi)$ implies (2) $\mathcal{E}, \pi, i, \Theta \vDash \exists X.\,\mathtt{K}_a(X \overset{A}{\rightsquigarrow} \psi)$. From the definition of temporal cause (cf. Section 2) it follows for any trace π' that (3) $\mathit{Cause}(\psi, \pi', i, A) = \{\pi'' \in \mathit{Cause}(\psi, \pi', i, B) \mid$

$\pi' =_{(Act \setminus A)} \pi''\}$, i.e., the cause over A is exactly the subset of the cause over B that is fixed over the larger action set $Act \setminus A$. From (1) we know that $Cause(\psi, \pi, i, B) = Cause(\psi, \rho, i, B)$ for any trace ρ indistinguishable to π up to i, i.e., where $\Omega_a(\pi)[0, i] = \Omega_a(\rho)[0, i]$. With (3), we then also have for the same trace ρ that $Cause(\psi, \pi, i, A) = Cause(\psi, \rho, i, A)$ and (2) follows. □

Explanations Imply Knowledge. The previously introduced definitions of explainability such as FCE (cf. Definition 2) do not explicitly require agent a to know that the effect ψ holds. This is a deliberate design decision to keep the specifications succinct, as knowledge of the outcome is implied by knowledge of the counterfactual dependency, except in the case of nondeterminism on the action sequence.

Proposition 2. *If a deterministic system $\mathcal{E}$, a trace $\pi \in \Pi(\mathcal{E})$, a time point $i \in \mathbb{N}$, some $A \subseteq AP$, and an arbitrary second-order assignment Θ satisfy*

$$\psi \wedge \exists X. \mathtt{K}_a(X \overset{A}{\rightsquigarrow} \psi) \ , \quad \textit{then also} \quad \mathtt{K}_a(\psi) \ .$$

Proof. From the left side of the implication we know there is a set T that can instantiate X, such that $\mathtt{K}_a(X \overset{A}{\rightsquigarrow} \psi)$ holds on π at i. We have assumed $\pi \vDash \psi$, and since $\mathcal{E}$ is deterministic there is no π'' such that $\pi'' \leq^A_\pi \pi$, $\pi'' =_{(Act \setminus A)} \pi$ and $\pi'' \nvDash \psi$ (only a trace with the same action sequence could be as similar to π as π itself). Hence, we have $\pi|_A \in T$, where $\pi|_A[i]$ is the projection of π to A such that $\pi|_A[i] = \pi[i] \cap A$ for all $n \in \mathbb{N}$. Now, let π' be any trace indistinguishable to π up to i, i.e., $\Omega_a(\pi)[0, i] = \Omega_a(\pi')[0, i]$. From $\mathtt{K}_a(X \overset{A}{\rightsquigarrow} \psi)$ we know that $T = Cause(\psi, \pi', i, A) \ni \pi|_A$, and since we trivially have that $\pi' \leq^A_{\pi'} \pi|_A$ and $\pi' =_{(Act \setminus A)} \pi'$, it holds that $\pi' \vDash \psi$ from the definition of a temporal cause (cf. Section 2), hence the claim follows. □

Thus, in a deterministic system an agent can only explain present facts that they have knowledge of, while in a nondeterministic system an agent may know that some fact could be caused by nondeterminism, in which case they may not be sure whether it actually holds on a given trace.

The resolution of nondeterminism can be included more directly in explanations. This requires modeling nondeterminism as an additional agent n, which intuitively flips a number of coins that then determine which previously nondeterminisitc transition is taken. By including the actions $Act(n)$ of this agent in the respective counterfactual predicates of the explainability requirements such as ICE, an explainable system is then required to transmit information about the outcomes of these coin flips. Notably, the system resulting from such a construction is deterministic, such that Proposition 2 applies.

4.3 Model Checking YLTL^2

We now outline an algorithm for model checking finite-state multi-agent systems against YLTL^2 formulas. The central challenge is the second-order quantification ranging over sets of traces. Logics with unrestricted quantifiers of this kind are known to have an undecidable model-checking problem (Beutner et al. 2023a). YLTL^2 restricts usage of second-order variables to causal predicates. The solution to such a causal predicate is uniquely determined for every anchor point, and this allows us to frame second-order quantification in YLTL^2 less as a search for a solution and more as a check for equality between the unique solutions at different anchor points. If a causal predicate appears in the scope of an epistemic or temporal operator, there may be an infinite number of such anchor points that need to be compared, which makes such a check non-trivial to achieve. In the proof of the following Theorem 1, we show that the check can essentially be encoded through quantification over individual traces. We express this trace quantification in a logic that can quantify over fresh atomic propositions by restricting these to follow the dynamics of the system at hand. This idea has been used in a number of related results on model-checking logics with trace quantification, i.e., so-called *hyperlogics* (Clarkson et al. 2014; Bozzelli, Maubert, and Pinchinat 2015). However, none of these include second-order quantifiers.

Theorem 1 (YLTL^2 Model Checking)**.** *There is an algorithm to decide whether a given extended transition system $\mathcal{E} = (\mathcal{T}, \Omega)$ satisfies a YLTL^2 formula φ, i.e., $\mathcal{E} \vDash \varphi$.*

Proof. The claim follows from a reduction to the satisfiability problem of Quantified Propositional Temporal Logic (QPTL) (Sistla, Vardi, and Wolper 1987). QPTL extends LTL with quantifiers over fresh atomic propositions $p \in \text{AP}$:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \bigcirc\varphi \mid \bullet\varphi \mid \varphi \,\mathtt{U}\, \varphi \mid \varphi \,\mathtt{U}^{-}\, \varphi \mid \exists p. \varphi \ ,$$

The semantics of the shared fragment are exactly as for LTL. The propositional quantification semantics are as follows:

$$\pi, i \vDash \exists p. \varphi \text{ iff } \exists \pi' \in (2^{\text{AP}})^\omega : \pi =_{\text{AP}\setminus\{p\}} \pi' \text{ and } \pi', i \vDash \varphi.$$

The *satisfiability* problem of such a QPTL formula φ asks whether there is a trace $\pi \in (2^{\text{AP}})^\omega$ such that $\pi, 0 \vDash \varphi$ and is decidable (Sistla, Vardi, and Wolper 1987).

We can reduce the YLTL^2 model checking to QPTL satisfiability via a translation function enc that encodes the extended transition system $\mathcal{E} = (\mathcal{T}, \Omega)$ and the YLTL^2 formula φ into a QPTL formula. We combine several translations from Bozzelli, Maubert, and Pinchinat (2015) for the KLTL-fragment of YLTL^2. The second-order quantification requires a novel, non-trivial approach that we outline afterward. The translation models different paths of the transition system through distinct atomic propositions. Therefore, the resulting formula ranges over an augmented set $\text{AP}' = \{\, p_\pi \mid p \in (\text{AP} \cup S) \wedge \pi \in \text{PV}\}$, where PV denotes a set of *path variables*. For a YLTL^2 formula φ, it suffices to introduce one such path variable for every knowledge operator $\mathtt{K}_a$ and causal predicate, two for every second-order quantifier, and one initial variable α that encodes the universal application of the trace semantics. Hence, AP$'$ is finite. We can enforce that these new propositions in AP$'$ evolve according to the transitions of $\mathcal{E} = (\mathcal{T}, \Omega)$:

$$\theta(\pi, \mathcal{T}) = \blacklozenge\Big((\neg\bullet\top) \wedge (s_0)_\pi \wedge \square \bigwedge_{s \in S} \big[s_\pi \rightarrow \bigwedge_{t \in S \setminus \{s\}} \neg t_\pi \wedge \bigvee_{t \in \Delta(s)} \big(\bigcirc t_\pi \wedge \bigwedge_{p \in \Lambda(s,t)} p_\pi \wedge \bigwedge_{p \in \text{AP} \setminus \Lambda(s,t)} \neg p_\pi\big)\big]\Big).$$

In the end, our proof establishes the following equivalence, where $\theta(\alpha, \mathcal{T})$ is used to quantify over all initial traces of $\mathcal{T}$: The model $\mathcal{E}$ satisfies the formula φ iff

$$\forall \text{AP}_\alpha \cup S_\alpha . \theta(\alpha, \mathcal{T}) \rightarrow enc(\alpha, \varphi, v_\mathfrak{E}) \tag{2}$$

is satisfiable. The third parameter of enc is a mapping from second-order variables to trace variables, where $v_\mathfrak{E}$ denotes the empty mapping. The third parameter will play a central role later in the translation of second-order quantifiers, which we discuss after the simple cases.

Temporal Operators: Since QPTL has the same temporal operators as YLTL^2, the translation in these cases is straightforward. We define enc through recursion on φ:

$$\begin{aligned} enc(\pi, p, v) &= p_\pi \\ enc(\pi, \neg\psi, v) &= \neg enc(\pi, \psi, v) \\ enc(\pi, \psi_1 \wedge \psi_2, v) &= enc(\pi, \psi_1, v) \wedge enc(\pi, \psi_2, v) \\ enc(\pi, \bigcirc \psi, v) &= \bigcirc enc(\pi, \psi, v) \ . \end{aligned}$$

The other operators follow analogously. Intuitively, π is the currently scoped trace variable and corresponds to π in the semantics of KLTL and YLTL^2 (cf. Sections 2 and 4.1).

Epistemic Operator: For the epistemic operator $\mathtt{K}_a$, the translation exploits that quantification over paths $\pi \in \mathcal{T}$ can be encoded through QPTL quantifiers as outlined already with Formula 2. We additionally need to constrain these initial paths to be observation equivalent for agent a as follows:

$$\begin{aligned} enc(\pi, \mathtt{K}_a\, \psi, v) = \forall \text{AP}_\rho \cup S_\rho . \big(&\theta(\rho, \mathcal{T}) \wedge \blacksquare(\pi =_{\Omega(a)} \rho)\big) \\ &\rightarrow enc(\rho, \psi, v) \ , \end{aligned}$$

where observation equivalence $=_{\Omega(a)}$ for an agent a at a certain time point is translated as follows:

$$\pi =_{\Omega(a)} \rho \equiv \bigwedge_{p \in \Omega(a)} p_\pi \leftrightarrow p_\rho \ .$$

Second-Order Quantifiers & Causal Predicates: The second-order quantifiers for causes require a more complex encoding than $\mathtt{K}_a$, since they quantify over sets of traces and not single traces. The main idea of our encoding is that for any existentially quantified causal set, all initial system traces have to either satisfy *all* associated causal predicates or *none* of them. In the former case the trace is in the cause at all anchor points, while in the latter case the trace is in the complement at all anchor points. This ensures that there are no traces that are in the cause at some anchor points and not in the cause at others, which would mean these causes are not equal and no single set qualifies at every anchor point. Note that this connection exploits that causes are uniquely determined at every anchor point, if they exist. We can encode these requirements with:

$$\begin{aligned} enc(\pi, \exists X . \psi, v) = \big(&\forall \text{AP}_\rho \cup S_\rho . enc(\pi, \psi, v_1) && (3) \\ &\vee enc(\pi, \psi, v_2)\big) \ . && (4) \end{aligned}$$

The variable mappings v_1 and v_2 encode whether the subformula refers to the causal set or its complement, respectively:

$$v_1(Y) = \begin{cases} (\rho, \top) & \text{if Y = X,} \\ v(Y) & \text{otherwise, and} \end{cases}$$

$$v_2(Y) = \begin{cases} (\rho, \bot) & \text{if Y = X,} \\ v(Y) & \text{otherwise.} \end{cases}$$

Intuitively, Subformula 3 encodes that the trace of ρ is in the cause X at all causal predicates $X \overset{A}{\rightsquigarrow} \varphi$, while Subformula 4 encodes that the trace is in no such cause. The mappings v_1 and v_2 track the location of subformulas, i.e., by mapping second-order variables to $\top$ and $\bot$, respectively. It remains to encode the (non-)membership in X at the location of the causal predicates:

$$enc(\pi, X \overset{A}{\rightsquigarrow} \varphi, v) = \begin{cases} \psi_{cause} & \text{if } v(X).2 = \top, \\ \neg\psi_{cause} & \text{if } v(X).2 = \bot, \end{cases}$$

Where $v(X).2$ stands for the second component of the tuple (and $v(X).1$ for the first), and where we encode membership in $Cause(\psi, \pi, i, A)$ via quantifying over individual traces to express that $v(X).1$ is in the downward-closed set as defined in Section 2:

$$\begin{aligned} \psi_{cause} = \forall \text{AP}_\sigma \cup S_\sigma . \theta(\sigma, \mathcal{T}) \rightarrow \\ \big(\sigma \leq^A_\pi v(X).1 \rightarrow enc(\sigma, \varphi, v)\big) \ . \end{aligned}$$

The similarity relation $\leq^A_\pi$ can be translated into temporal logic as follows:

$$\begin{aligned} \sigma \leq^A_\pi \rho \equiv \blacklozenge \Big((\neg \bullet \top) \wedge \square \big[\bigwedge_{p \in Act \setminus A} p_\sigma \leftrightarrow p_\pi \\ \wedge \bigwedge_{p \in A} (p_\sigma \not\leftrightarrow p_\pi) \rightarrow (p_\rho \not\leftrightarrow p_\pi) \big] \Big) \ . \end{aligned}$$

This concludes the description of enc. The equivalence of the model-checking problem to the satisfiability of Formula 2 can be shown through structural induction on φ. □

Complexity. It is easy to see that the translation function enc may double the size of the formula when encoding a second-order quantifier (cf. Equations 3 and 4). Less obvious is an added propositional quantifier alternation, introduced by translating causal predicates $X \overset{A}{\rightsquigarrow} \varphi$ in the second disjunct (cf. Equation 4); these predicates are translated to $\neg\psi_{cause}$ by adding a negated universal quantifier. This results in a non-elementary (tower-exponential) space complexity of model-checking YLTL^2 formulas, where the tower grows with any nesting of second-order quantifiers in the scope of causal predicates and with nesting of knowledge operators and negations. Previous results suggest that this complexity is close to optimal: YLTL^2 subsumes KLTL, which has a similar non-elementary lower bound scaling with nested epistemic operators (Bozzelli, Maubert, and Murano 2024). Constructing temporal causes as automata has a doubly exponential lower bound, in the size of the consequent (Carelli, Finkbeiner, and Siber 2025), which matches with nested causal predicates growing the exponents of the model-checking complexity.

In practice, the specifications we considered, such as ICE, ECE and FCE, all only contain one propositional quantifier alternation in their encodings, and model-checking them with the algorithm outlined for Theorem 1 therefore scales exponentially in the size of the effect φ and polynomially in the size of the system. Our experiments confirm that these specifications can be verified for systems of nontrivial size.

Type	$\|B\|$	ICE	ECE	FCE	Priv. I
BLIND	2	✗/0.85	✗/0.88	✗/0.96	✓/0.25
	3	✗/1.57	✗/1.77	✗/1.73	✓/0.27
	4	✗/3.42	✗/3.68	✗/3.61	✓/0.27
	5	✗/7.85	✗/9.45	✗/10.3	✓/0.22
PUBLIC	2	✓/0.78	✓/0.87	✓/0.81	✗/0.30
	3	✓/1.47	✓/1.80	✓/1.63	✗/0.30
	4	✓/3.19	✓/3.73	✓/3.48	✗/0.50
	5	✓/7.24	✓/ 9.52	✓/10.3	✗/1.21
EXPLAIN	2	✓/0.86	✗/1.02	✗/1.02	✗/0.24
	3	✓/1.39	✗/1.89	✗/1.82	✓/0.25
	4	✓/3.40	✗/3.82	✗/4.03	✓/0.29
	5	✓/8.03	✗/ 9.92	✗/10.4	✓/0.29

Table 1: Verification results and runtime in seconds for checking the Dutch auction model introduced in Section 3 with a set of bidders B against the explainability requirements introduced in Sections 4.2 and b_2-privacy (Priv. I, cf. Section 3).

5 Experiments

We report experiments on verifying explainability and privacy requirements specified in YLTL[2]. Although we describe an encoding into QPTL satisfiability in Theorem 1 for brevity, there is no satisfiability checker for QPTL. Hence, our evaluation uses the model-checking tool AutoHyper (Beutner and Finkbeiner 2023a) as a backend procedure, because QPTL satisfiability can be encoded in HyperLTL model checking (Finkbeiner, Rabe, and Sánchez 2015). The evaluation was conducted on MacOS with an M3 Pro 4.05 GHz processor and 36GB of memory.[1]

5.1 Dutch Auction

We conducted experiments with all three versions of the Dutch auction system introduced in Section 3 and the explainability requirements introduced in Secion 4.2. We also checked b_2-privacy as introduced in Section 3. In Table 1, we list whether the respective auction version satisfies a requirement and how much time the model checker required to produce this verdict. We consider a scaling number of bidders B, which increases the size of the transition system exponentially, and this scaling is mirrored in the runtimes.

The verification verdicts match the intuition described in Section 3: The blind auction is not explainable but private, while the public auction naturally ensures sufficient flow of information to achieve all three explainability requirements, but also reveals the private bids. The explainable auction transports enough information on the causal dependencies over the actions of the bidders themselves, but does not ensure ECE or FCE. An interesting case is the privacy of the explainable auction in the case of two bidders. In this scenario, revealing the value of the highest bid allows an agent to infer that the other agent placed the bid in case they did not bid themselves, in this way violating the privacy constraint. Hence, explainability and b_2-privacy can only be achieved with more than two bidders.

[1]Code: https://doi.org/10.5281/zenodo.16421482

5.2 Rock Paper Scissors

To differentiate ICE and ECE in more detail, we model the classic game of Rock Paper Scissors, where every round two players select between the three eponymous objects (r, p, and s). Matching objects result in a draw, otherwise p beats r, s beats p, and r beats s. An agent i does not see the selection of the other, but only the outcome draw (d) or loss (l_i). We consider explainability of l_1, i.e., the loss of Agent 1. Since an agent knows what action their picked one looses to, and which action would have won instead, even this blind version of the game is fully explainable according to all three requirements. The verification times are as follows.

Type	ICE	ECE	FCE	Priv. II
STANDARD	✓/0.49	✓/0.53	✓/0.55	✗/0.24
+ WELL	✗/1.02	✓/0.93	✗/1.08	✓/0.29

We also consider a popular variant with a fourth action, Well (w), which beats s and r but looses to p. This version does not satisfy ICE because, e.g., when loosing with rock (r_1) Agent 1 does not know whether scissors (against p_2) or paper (against w_2) would have been their winning move. ECE is still satisfied because the agent knows against which moves they would have won, i.e., p_1 would have won if the other agent had played r_2 or w_2 (the cause is then $\neg r_2 \wedge \neg w_2$). We also verified the following conditional privacy specification (results reported under Priv. II): $\square(\neg d \rightarrow \neg \mathtt{K}_1(p_2))$, i.e., whenever the outcome is not a draw, an agent does not know whether the other agent played p. This is only satisfied in the extended version of the game, because an agent cannot discern between w_2 and p_2 except in case of a draw with their own picked action.

5.3 Matching Pennies

To further explore the scalability of our algorithm we consider a blind version of the game Matching Pennies, played collaboratively: Each player chooses heads or tails for their coin and all players win together when their choices match. A player i only sees their coin c_i and the outcome w, as well as an explanatory *blaming* output b_i, which is enabled if their coin was the only mismatch. Without the blaming output, the setup satisfies all three explainability requirements only in the 2-player case. With the blaming output, it additionally satisfies ICE for any number of players. Runtime results for the latter experiments are shown in Figure 2. They confirm that model-checking the three explainability requirements scales polynomially with an increasing system size. The plot also shows that the time needed to verify the conditional privacy requirement $\square(\neg w \rightarrow \neg \mathtt{K}_1(c_2))$ stays practically constant (cf. Priv. III). This property states that when Player 1 does not win, they do not know the value of Player 2's coin. This is satisfied only by the non-blaming version with more than two players. We suspect this property allows the model checker to fully abstract away from the coins of the other players, while this is not possible for the explainability specifications, which place constraints on these coins in the encoding of $\leq_\pi^A$ (cf. proof of Theorem 1).

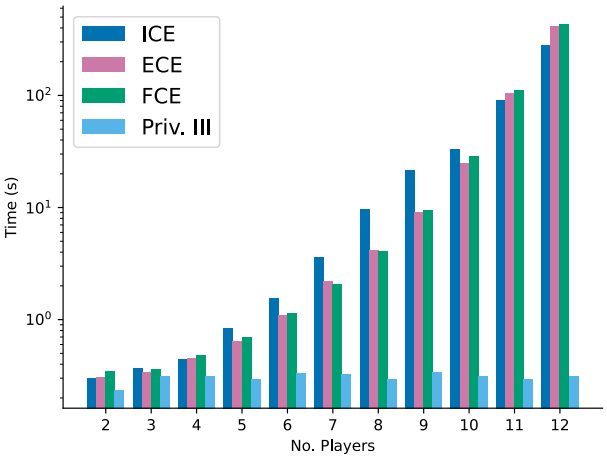

Figure 2: Runtime results for verifying explainability and privacy requirements of blind Matching Pennies games with blaming output. Note that system size scales exponentially with the number of players, i.e., the 12-player game has 4096 transitions.

6 Related Work

Explanation Generation: Our work is concerned with formalizing explainability as a system property and not with generating explanations like other literature on explainable AI (Ribeiro, Singh, and Guestrin 2016; Lundberg and Lee 2017; Shih, Choi, and Darwiche 2018). Nevertheless, these generation methods are a central motivation for us because the systems resulting from their implementation will need to tread the fine line between explainability and privacy that we study here. Applying our insights may be easier for model-based approaches that deal with descriptive systems such as decision trees (Darwiche and Ji 2022; Arenas et al. 2021; Arenas et al. 2024; Carbonnel, Cooper, and Marques-Silva 2023) than for black-box methods based on, e.g., abduction (Ignatiev, Narodytska, and Marques-Silva 2019) or heuristics (Angelino et al. 2017; Li et al. 2018).

Formal Explanations: Logical reasoning techniques to compute explanations are widely studied (Darwiche 2023; Wu, Wu, and Barrett 2023; Leofante, Botoeva, and Rajani 2023). Khan and Lespérance (2021) combine epistemic and causal reasoning in the situation calculus, which has been extended to explain agent behavior (Khan and Rostamigiv 2023). Temporal properties are used as explanations in AI planning (Kim et al. 2019; Eifler et al. 2020); another popular framework in planning is model reconciliation (Chakraborti et al. 2017). Planning with explanatory actions (Chakraborti et al. 2019; Sreedharan et al. 2019) follows a similar goal as us in effecting the epistemic state through observations. Several works on axiomatizing explainable classifiers (Amgoud and Ben-Naim 2022; Liu and Lorini 2023) employ counterfactual reasoning and consider partial knowledge. There are works on actual causality (Halpern 2016) for explanations (Chockler and Halpern 2024), as well as counterfactual modal logic (Aguilera-Ventura et al. 2023), but we use temporal causes (Coenen et al. 2022; Finkbeiner et al. 2024) to deal with the sequential and possibly non-terminating nature of multi-agent systems.

Logics for Information Flow: Logics for hyperproperties (Clarkson and Schneider 2010) have been a popular framework to study information flow and partly subsume epistemic temporal logic (Bozzelli, Maubert, and Pinchinat 2015; Rabe 2016), which has similarly been used for information-flow control (Balliu, Dam, and Guernic 2011; Halpern and O'Neill 2008). Counterfactual reasoning has been encoded in such *hyperlogics* (Coenen et al. 2022; Finkbeiner and Siber 2023; Beutner et al. 2023b) for checking hypotheses. Several works study hyperlogics in multi-agent systems without second-order quantifiers (Beutner and Finkbeiner 2023b; Beutner and Finkbeiner 2024a; Beutner and Finkbeiner 2024b). The closest work to ours considers a second-order hyperlogic without decidable model checking (Beutner et al. 2023a; Beutner et al. 2024).

Interdisciplinary Aspects: Our perspective on explainability is rooted in logic and information-flow theory. Building self-explanatory systems from explainable system models is an intriguing interdisciplinary problem that lies outside the scope of this paper. There are user studies regarding which explanations are preferred by users and how to visualize them (Seimetz, Eifler, and Hoffmann 2021; Schlicker et al. 2021; Horak et al. 2022; Eifler et al. 2022; Brandao et al. 2022; Delaney et al. 2023). We use temporal causes, which capture the idea of minimally editing previous actions of the agents, but our information-flow perspective may also be applied to other explanantia such as presenting counterfactual executions. This requires modifying the encoding in the proof of Theorem 1. High-level taxonomies for explainability requirements (Köhl et al. 2019; Langer et al. 2021) have inspired us to concretize them with formal logic and multi-agent systems.

7 Summary & Conclusion

This paper presents a logic for multi-agent systems to formally specify explainability requirements of the form: when a certain outcome happens, an agent knows *why*, i.e., what actions caused the outcome. This is expressed with a combination of counterfactual, epistemic, and temporal operators, as well as second-order quantification over sets of executions. Privacy requirements can be encoded in the same logic and we have described an algorithm to automatically verify whether a system model satisfies such specifications.

Our theoretical and experimental results suggest that our formal explainability specifications capture what it means for a system to be explainable in a qualitative information-theoretic sense: An explainable system needs to ensure sufficient flow of information about causes via a number of observations that serve as explanations. On an abstract level, this means that in explainable systems observation-equivalence needs to be finer than causality.

We discovered an inherent tradeoff between explainability and privacy, which is an intriguing avenue for future work. Automatic satisfiability checking of our logic may allow to identify general rules for this tradeoff beyond analyzing a given system, while repair algorithms may add a minimal set of explanatory observations to a system model to achieve explainability without sacrificing privacy.

Acknowledgements

This work was partially supported by the DFG in project 389792660 (TRR 248 – CPEC) and funded by the European Union through ERC Grant HYPER (No. 101055412). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This work was supported in part by The Israel Science Foundation (grant No. 655/25).

References

Aguilera-Ventura, C.; Herzig, A.; Liu, X.; and Lorini, E. 2023. Counterfactual reasoning via grounded distance. In Marquis, P.; Son, T. C.; and Kern-Isberner, G., eds., *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, KR 2023, Rhodes, Greece, September 2-8, 2023*, 2–11.

Amgoud, L., and Ben-Naim, J. 2022. Axiomatic foundations of explainability. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, 636–642. ijcai.org.

Angelino, E.; Larus-Stone, N.; Alabi, D.; Seltzer, M. I.; and Rudin, C. 2017. Learning certifiably optimal rule lists for categorical data. *J. Mach. Learn. Res.* 18:234:1–234:78.

Angwin, J.; Larson, J.; Mattu, S.; and Kirchner, L. 2016. Machine bias. there's software used across the country to predict future criminals. and it's biased against blacks. *ProPublica*.

Arenas, M.; Báez, D.; Barceló, P.; Pérez, J.; and Subercaseaux, B. 2021. Foundations of symbolic languages for model interpretability. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y. N.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, 11690–11701.

Arenas, M.; Barceló, P.; Bustamante, D.; Caraball, J.; and Subercaseaux, B. 2024. A uniform language to explain decision trees. In Marquis, P.; Ortiz, M.; and Pagnucco, M., eds., *Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning, KR 2024, Hanoi, Vietnam. November 2-8, 2024*.

Balliu, M.; Dam, M.; and Guernic, G. L. 2011. Epistemic temporal logic for information flow security. In Askarov, A., and Guttman, J. D., eds., *Proceedings of the 2011 Workshop on Programming Languages and Analysis for Security, PLAS 2011, San Jose, CA, USA, 5 June, 2011*, 6. ACM.

Beutner, R., and Finkbeiner, B. 2023a. Autohyper: Explicit-state model checking for HyperLTL. In Sankaranarayanan, S., and Sharygina, N., eds., *Tools and Algorithms for the Construction and Analysis of Systems - 29th International Conference, TACAS 2023, Paris, France, April 22-27, 2023, Proceedings, Part I*, volume 13993 of *LNCS*, 145–163. Springer.

Beutner, R., and Finkbeiner, B. 2023b. Hyperatl*: A logic for hyperproperties in multi-agent systems. *Log. Methods Comput. Sci.* 19(2).

Beutner, R., and Finkbeiner, B. 2024a. Hyper strategy logic. In Dastani, M.; Sichman, J. S.; Alechina, N.; and Dignum, V., eds., *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024, Auckland, New Zealand, May 6-10, 2024*, 189–197. International Foundation for Autonomous Agents and Multiagent Systems / ACM.

Beutner, R., and Finkbeiner, B. 2024b. On alternating-time temporal logic, hyperproperties, and strategy sharing. In Wooldridge, M. J.; Dy, J. G.; and Natarajan, S., eds., *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, 17317–17326. AAAI Press.

Beutner, R.; Finkbeiner, B.; Frenkel, H.; and Metzger, N. 2023a. Second-order hyperproperties. In Enea, C., and Lal, A., eds., *Computer Aided Verification - 35th International Conference, CAV 2023, Paris, France, July 17-22, 2023, Proceedings, Part II*, volume 13965 of *Lecture Notes in Computer Science*, 309–332. Springer.

Beutner, R.; Finkbeiner, B.; Frenkel, H.; and Siber, J. 2023b. Checking and sketching causes on temporal sequences. In André, É., and Sun, J., eds., *Automated Technology for Verification and Analysis - 21st International Symposium, ATVA 2023, Singapore, October 24-27, 2023, Proceedings, Part II*, volume 14216 of *Lecture Notes in Computer Science*, 314–327. Springer.

Beutner, R.; Finkbeiner, B.; Frenkel, H.; and Metzger, N. 2024. Monitoring second-order hyperproperties. In Dastani, M.; Sichman, J. S.; Alechina, N.; and Dignum, V., eds., *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024, Auckland, New Zealand, May 6-10, 2024*, 180–188. International Foundation for Autonomous Agents and Multiagent Systems / ACM.

Bozzelli, L.; Maubert, B.; and Murano, A. 2024. On the complexity of model checking knowledge and time. *ACM Trans. Comput. Log.* 25(1):8:1–8:42.

Bozzelli, L.; Maubert, B.; and Pinchinat, S. 2015. Unifying hyper and epistemic temporal logics. In Pitts, A. M., ed., *Foundations of Software Science and Computation Structures - 18th International Conference, FoSSaCS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015. Proceedings*, volume 9034 of *Lecture Notes in Computer Science*, 167–182. Springer.

Brandao, M.; Mansouri, M.; Mohammed, A.; Luff, P.; and Coles, A. J. 2022. Explainability in multi-agent path/motion planning: User-study-driven taxonomy and requirements. In Faliszewski, P.; Mascardi, V.; Pelachaud, C.; and Taylor, M. E., eds., *21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland,*

New Zealand, May 9-13, 2022, 172–180. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS).

Carbonnel, C.; Cooper, M. C.; and Marques-Silva, J. 2023. Tractable explaining of multivariate decision trees. In Marquis, P.; Son, T. C.; and Kern-Isberner, G., eds., *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, KR 2023, Rhodes, Greece, September 2-8, 2023*, 127–135.

Carelli, M.; Finkbeiner, B.; and Siber, J. 2025. Closure and complexity of temporal causality. In *40th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2025, Singapore, June 23-26, 2025*. IEEE.

Chakraborti, T.; Sreedharan, S.; Zhang, Y.; and Kambhampati, S. 2017. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 156–163. ijcai.org.

Chakraborti, T.; Kulkarni, A.; Sreedharan, S.; Smith, D. E.; and Kambhampati, S. 2019. Explicability? legibility? predictability? transparency? privacy? security? the emerging landscape of interpretable agent behavior. In Benton, J.; Lipovetzky, N.; Onaindia, E.; Smith, D. E.; and Srivastava, S., eds., *Proceedings of the Twenty-Ninth International Conference on Automated Planning and Scheduling, ICAPS 2019, Berkeley, CA, USA, July 11-15, 2019*, 86–96. AAAI Press.

Chockler, H., and Halpern, J. Y. 2024. Explaining image classifiers. In Marquis, P.; Ortiz, M.; and Pagnucco, M., eds., *Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning, KR 2024, Hanoi, Vietnam. November 2-8, 2024*.

Clarkson, M. R., and Schneider, F. B. 2010. Hyperproperties. *J. Comput. Secur.* 18(6):1157–1210.

Clarkson, M. R.; Finkbeiner, B.; Koleini, M.; Micinski, K. K.; Rabe, M. N.; and Sánchez, C. 2014. Temporal logics for hyperproperties. In Abadi, M., and Kremer, S., eds., *Principles of Security and Trust - Third International Conference, POST 2014, Grenoble, France, April 5-13, 2014, Proceedings*, volume 8414 of *Lecture Notes in Computer Science*, 265–284. Springer.

Coenen, N.; Finkbeiner, B.; Hahn, C.; and Hofmann, J. 2019. The hierarchy of hyperlogics. In *34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019*, 1–13. IEEE.

Coenen, N.; Finkbeiner, B.; Frenkel, H.; Hahn, C.; Metzger, N.; and Siber, J. 2022. Temporal causality in reactive systems. In Bouajjani, A.; Holík, L.; and Wu, Z., eds., *Automated Technology for Verification and Analysis - 20th International Symposium, ATVA 2022, Virtual Event, October 25-28, 2022, Proceedings*, volume 13505 of *Lecture Notes in Computer Science*, 208–224. Springer.

Darwiche, A., and Ji, C. 2022. On the computation of necessary and sufficient explanations. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, 5582–5591. AAAI Press.

Darwiche, A. 2023. Logic for explainable AI. In *38th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2023, Boston, MA, USA, June 26-29, 2023*, 1–11. IEEE.

Dastin, J. 2018. Amazon scraps secret ai recruiting tool that showed bias against women. https://www.reuters.com/article/idUSKCN1MK0AG/ (Accessed: 28.04.2025).

Delaney, E.; Pakrashi, A.; Greene, D.; and Keane, M. T. 2023. Counterfactual explanations for misclassified images: How human and machine explanations differ. *Artif. Intell.* 324:103995.

Eifler, R.; Steinmetz, M.; Torralba, Á.; and Hoffmann, J. 2020. Plan-space explanation via plan-property dependencies: Faster algorithms & more powerful properties. In Bessiere, C., ed., *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, 4091–4097. ijcai.org.

Eifler, R.; Brandao, M.; Coles, A. J.; Frank, J.; and Hoffmann, J. 2022. Evaluating plan-property dependencies: A web-based platform and user study. In Kumar, A.; Thiébaux, S.; Varakantham, P.; and Yeoh, W., eds., *Proceedings of the Thirty-Second International Conference on Automated Planning and Scheduling, ICAPS 2022, Singapore (virtual), June 13-24, 2022*, 687–691. AAAI Press.

Fagin, R.; Halpern, J. Y.; Moses, Y.; and Vardi, M. Y. 1995. *Reasoning About Knowledge*. MIT Press.

Finkbeiner, B., and Siber, J. 2023. Counterfactuals modulo temporal logics. In Piskac, R., and Voronkov, A., eds., *LPAR 2023: Proceedings of 24th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, Manizales, Colombia, 4-9th June 2023*, volume 94 of *EPiC Series in Computing*, 181–204. EasyChair.

Finkbeiner, B.; Frenkel, H.; Metzger, N.; and Siber, J. 2024. Synthesis of temporal causality. In Gurfinkel, A., and Ganesh, V., eds., *Computer Aided Verification - 36th International Conference, CAV 2024, Montreal, QC, Canada, July 24-27, 2024, Proceedings, Part III*, volume 14683 of *Lecture Notes in Computer Science*, 87–111. Springer.

Finkbeiner, B.; Rabe, M. N.; and Sánchez, C. 2015. Algorithms for model checking hyperltl and hyperctl ^*. In Kroening, D., and Pasareanu, C. S., eds., *Computer Aided Verification - 27th International Conference, CAV 2015, San Francisco, CA, USA, July 18-24, 2015, Proceedings, Part I*, volume 9206 of *LNCS*, 30–48. Springer.

Goguen, J. A., and Meseguer, J. 1982. Security policies and security models. In *1982 IEEE Symposium on Security and Privacy, Oakland, CA, USA, April 26-28, 1982*, 11–20. IEEE Computer Society.

Halpern, J. Y., and O'Neill, K. R. 2008. Secrecy in multiagent systems. *ACM Trans. Inf. Syst. Secur.* 12(1):5:1–5:47.

Halpern, J. Y., and Pearl, J. 2005. Causes and explanations: A structural-model approach. part ii: Explanations. *The*

British Journal for the Philosophy of Science 56(4):889–911.

Halpern, J. Y.; van der Meyden, R.; and Vardi, M. Y. 2004. Complete axiomatizations for reasoning about knowledge and time. *SIAM J. Comput.* 33(3):674–703.

Halpern, J. Y. 2016. *Actual Causality*. MIT Press.

Horak, T.; Coenen, N.; Metzger, N.; Hahn, C.; Flemisch, T.; Méndez, J.; Dimov, D.; Finkbeiner, B.; and Dachselt, R. 2022. Visual analysis of hyperproperties for understanding model checking results. *IEEE Trans. Vis. Comput. Graph.* 28(1):357–367.

Ignatiev, A.; Narodytska, N.; and Marques-Silva, J. 2019. Abduction-based explanations for machine learning models. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, 1511–1519. AAAI Press.

Khan, S. M., and Lespérance, Y. 2021. Knowing why - on the dynamics of knowledge about actual causes in the situation calculus. In Dignum, F.; Lomuscio, A.; Endriss, U.; and Nowé, A., eds., *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, 701–709. ACM.

Khan, S. M., and Rostamigiv, M. 2023. On explaining agent behaviour via root cause analysis: A formal account grounded in theory of mind. In Gal, K.; Nowé, A.; Nalepa, G. J.; Fairstein, R.; and Radulescu, R., eds., *ECAI 2023 - 26th European Conference on Artificial Intelligence, September 30 - October 4, 2023, Kraków, Poland - Including 12th Conference on Prestigious Applications of Intelligent Systems (PAIS 2023)*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, 1239–1247. IOS Press.

Kim, J.; Muise, C.; Shah, A.; Agarwal, S.; and Shah, J. 2019. Bayesian inference of linear temporal logic specifications for contrastive explanations. In Kraus, S., ed., *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, 5591–5598. ijcai.org.

Köhl, M. A.; Baum, K.; Langer, M.; Oster, D.; Speith, T.; and Bohlender, D. 2019. Explainability as a non-functional requirement. In Damian, D. E.; Perini, A.; and Lee, S., eds., *27th IEEE International Requirements Engineering Conference, RE 2019, Jeju Island, Korea (South), September 23-27, 2019*, 363–368. IEEE.

Kozyri, E.; Chong, S.; and Myers, A. C. 2022. Expressing information flow properties. *Found. Trends Priv. Secur.* 3(1):1–102.

Langer, M.; Oster, D.; Speith, T.; Hermanns, H.; Kästner, L.; Schmidt, E.; Sesing, A.; and Baum, K. 2021. What do we want from explainable artificial intelligence (xai)? - A stakeholder perspective on XAI and a conceptual model guiding interdisciplinary XAI research. *Artif. Intell.* 296:103473.

Leofante, F.; Botoeva, E.; and Rajani, V. 2023. Counterfactual explanations and model multiplicity: a relational verification view. In Marquis, P.; Son, T. C.; and Kern-Isberner, G., eds., *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, KR 2023, Rhodes, Greece, September 2-8, 2023*, 763–768.

Lewis, D. K. 1973. *Counterfactuals*. Cambridge, MA, USA: Blackwell.

Li, O.; Liu, H.; Chen, C.; and Rudin, C. 2018. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In McIlraith, S. A., and Weinberger, K. Q., eds., *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, 3530–3537. AAAI Press.

Lichtenstein, O.; Pnueli, A.; and Zuck, L. D. 1985. The glory of the past. In Parikh, R., ed., *Logics of Programs, Conference, Brooklyn College, New York, NY, USA, June 17-19, 1985, Proceedings*, volume 193 of *Lecture Notes in Computer Science*, 196–218. Springer.

Liu, X., and Lorini, E. 2023. A unified logical framework for explanations in classifier systems. *J. Log. Comput.* 33(2):485–515.

Lundberg, S. M., and Lee, S. 2017. A unified approach to interpreting model predictions. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 4765–4774.

Mersha, M.; Lam, K. N.; Wood, J.; AlShami, A.; and Kalita, J. 2024. Explainable artificial intelligence: A survey of needs, techniques, applications, and future direction. *Neurocomputing* 599:128111.

Miller, T. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.* 267:1–38.

Nguyen, T. T.; Huynh, T. T.; Ren, Z.; Nguyen, T. T.; Nguyen, P. L.; Yin, H.; and Nguyen, Q. V. H. 2025. Privacy-preserving explainable AI: a survey. *Sci. China Inf. Sci.* 68(1).

Pnueli, A. 1977. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science, Providence, Rhode Island, USA, 31 October - 1 November 1977*, 46–57. IEEE Computer Society.

Rabe, M. N. 2016. *A temporal logic approach to Information-flow control*. Ph.D. Dissertation, Saarland University.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "why should I trust you?": Explaining the predictions of any classifier. In Krishnapuram, B.; Shah, M.; Smola, A. J.; Aggarwal, C. C.; Shen, D.; and Rastogi, R., eds., *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 1135–1144. ACM.

Schlicker, N.; Langer, M.; Ötting, S. K.; Baum, K.; König,

C. J.; and Wallach, D. 2021. What to expect from opening up 'black boxes'? comparing perceptions of justice between human and automated agents. *Comput. Hum. Behav.* 122:106837.

Seimetz, V.; Eifler, R.; and Hoffmann, J. 2021. Learning temporal plan preferences from examples: An empirical study. In Zhou, Z., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, 4160–4166. ijcai.org.

Shih, A.; Choi, A.; and Darwiche, A. 2018. A symbolic approach to explaining bayesian network classifiers. In Lang, J., ed., *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, 5103–5111. ijcai.org.

Sistla, A. P.; Vardi, M. Y.; and Wolper, P. 1987. The complementation problem for büchi automata with appplications to temporal logic. *Theor. Comput. Sci.* 49:217–237.

Sreedharan, S.; Chakraborti, T.; Muise, C.; and Kambhampati, S. 2019. Expectation-aware planning: A unifying framework for synthesizing and executing self-explaining plans for human-aware planning.

Thomas, D., and Ravi, K. 2019. The potential for artificial intelligence in healthcare. *Future Healthc Journal* 6.

van der Meyden, R., and Shilov, N. V. 1999. Model checking knowledge and time in systems with perfect recall (extended abstract). In Rangan, C. P.; Raman, V.; and Ramanujam, R., eds., *Foundations of Software Technology and Theoretical Computer Science, 19th Conference, Chennai, India, December 13-15, 1999, Proceedings*, volume 1738 of *Lecture Notes in Computer Science*, 432–445. Springer.

Wachter, S.; Mittelstadt, B.; and Russell, C. 2018. Counterfactual explanations without opening the black box: automated decisions and the gdpr. *Harvard Journal of Law and Technology* 31(2):841–887.

Wu, M.; Wu, H.; and Barrett, C. W. 2023. Verix: Towards verified explainability of deep neural networks. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.