Acta Informatica (2025) 62:38
https://doi.org/10.1007/s00236-025-00507-w

RESEARCH

Explainability requirements as hyperproperties

Bernd Finkbeiner[1] · **Julian Siber**[1]

Received: 29 November 2024 / Accepted: 21 July 2025 / Published online: 13 October 2025
© The Author(s) 2025

Abstract
Explainability is emerging as a key requirement for autonomous systems. While many works have focused on what constitutes a valid explanation, few have considered formalizing explainability as a system property. In this work, we approach this problem from the perspective of hyperproperties. We start with a combination of three prominent flavors of modal logic and show how they can be used for specifying and verifying counterfactual explainability in multi-agent systems: With Lewis' counterfactuals, linear-time temporal logic, and a knowledge modality, we can reason about whether agents know *why* a specific observation occurs, i.e., whether that observation is *explainable* to them. We use this logic to formalize multiple notions of explainability on the system level. We then show how this logic can be embedded into a hyperlogic. Notably, from this analysis we conclude that the model-checking problem of our logic is decidable, which paves the way for the automated verification of explainability requirements.

1 Introduction

The increase in system complexity and opaqueness perceived in recent years has been answered by a plethora of techniques aimed at providing some sort of explanation for observed system behavior [2, 9, 47, 53]. While this demonstrates a need for systems to be explainable, there is no formal theory to specify different notions of explainability and to algorithmically verify them. In this paper, we make the claim that hyperproperties, and their respective logics, are an excellent basis for such a formal theory of explainability. We start from previous theories for individual instances of *explanations* [25, 26], which combine counterfactual and epistemic reasoning. Besides extending them to system specifications, we add temporal reasoning to specify explainability on the possibly infinite executions

✉ Julian Siber
julian.siber@cispa.de

Bernd Finkbeiner
finkbeiner@cispa.de

[1] CISPA Helmholtz Center for Information Security, Stuhlsatzenhaus 5, Saarbrücken 66123, Saarland, Germany

of multi-agent systems. We use modal operators for these three reasoning dimensions to express explainability requirements such as:

$$\square\Big(\neg \mathit{offer} \rightarrow \big(\bigvee_{\alpha,\beta \in Att(a)} \mathtt{K}_a \left((\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}\right)\big)\Big) \ ,$$

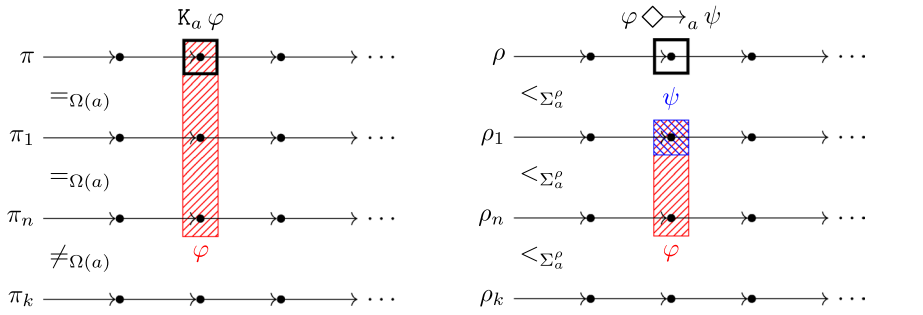
which we simply term *Internal Counterfactual Explainability* (ICE). Interpreted in a hiring system where some agent a applies to get a job offer, ICE states that, whenever agent a does not get the offer (i.e., atomic proposition *offer* does not hold), they know that if they had applied with some (other) attribute values $\alpha, \beta \in Att(a)$, they would have gotten the offer. We call this notion *internal* because it depends only on actions performed by agent a themselves. The formula for ICE uses operators from all three modal logics that we fuse together: It uses the temporal operator $\square$ to specify that the requirement holds at every time point and it uses the knowledge operator $\mathtt{K}_a$ to express that agent a has knowledge about some counterfactual dependency expressed with the counterfactual operator $\diamond\!\!\rightarrow_a$. Later on, we will use this logic to formalize other aspects of explainability, such as *weak*, *external*, and *general* explainability. We will also see how these notions discriminate between – intuitively – explainable and unexplainable systems in Section 3. This appeal to intuition is without alternative: There is no universally correct definition of explainability [33] and much depends on the context and the agents involved. The strength of our modal-logic approach is exactly that it provides a flexible specification language that can be applied to varying contexts and definitions, while retaining a general model-checking algorithm.

Double-fusions of the three modal logics we consider have been studied extensively: Epistemic temporal logic has been used in security for information-flow control [4, 24], counterfactual temporal logic for expressing causal dependencies in reactive systems [15, 18], and counterfactual epistemic logics to characterize notions of rationality in game theory [49, 52]. Our work brings these diverse frameworks together based on the viewpoint that explainability is an *intended flow of information about counterfactual dependencies*. This interpretation stands in the tradition of a long line of works on *individual* causal explanations [5, 22, 26, 37]. With this paper, we shift the focus away from the question of what constitutes a valid individual explanation toward analyzing the abstract epistemic properties that the global system needs to fulfill such that an explanation is available to an agent whenever it is needed. In short, we do not analyze explanations, but *explainability*.

1.1 Knowledge and counterfactuals

The logic we consider is an extension of epistemic temporal logic, in particular of Linear Temporal Logic (LTL) with the knowledge modality $\mathtt{K}$ (KLTL). We extend KLTL with counterfactual conditionals as defined by Lewis [36], which we interpret on paths of a multi-agent system.

We illustrate the semantics of these two modal operators in Figure 1. The epistemic formula $\mathtt{K}_a\,\varphi$ is satisfied at a given position of a given trace if all traces that are indistinguishable for agent a satisfy φ (cf. Subfigure 1a). Indistinguishability is defined based on the observation-equivalence $=_{\Omega(a)}$ that compares the prefixes of two traces with respect to the observations of agent a. The Lewisian counterfactual $\varphi \diamond\!\!\rightarrow_a \psi$, on the other hand, informally has the following meaning: "If φ had been true then ψ might also have been



(a) Semantics of the epistemic operator $\mathtt{K}_a$. (b) Semantics of the counterfactual $\diamond\!\!\rightarrow_a$.

Fig. 1 Illustrating the semantics of the statements $\mathtt{K}_a\,\varphi$ and $\varphi \diamond\!\!\rightarrow_a \psi$ on a set of traces. The statements are evaluated at the second position of the trace π, which is framed by the black square. The epistemic operator $\mathtt{K}_a$ (cf. Subfigure 1a) requires φ to hold at the same position on all traces π' with an observation equivalent prefix, i.e., where $\pi =_{\Omega(a)} \pi'$ is satisfied. These positions are covered by the area with diagonal lines (colored red). In contrast, the counterfactual $\diamond\!\!\rightarrow_a$ (cf. Subfigure 1b) requires that the trace *closest* to ρ that satisfies φ, which is in this case ρ_1, also satisfies ψ at the same position. This is marked by the area with crossed lines (colored red and blue). $<_{\Sigma_a^\rho}$ is used in the illustration to denote, e.g., $(\rho_n, \rho_k) \in \Sigma_a^\rho \wedge (\rho_k, \rho_n) \notin \Sigma_a^\rho$, which means that ρ_n is strictly more similar to ρ than ρ_k

true". More formally, the counterfactual formula has the following semantics: it holds on a position i of a given trace if one of the *closest* traces that satisfy φ at i also satisfies ψ at i. These semantics are illustrated in in Subfigure 1b, where there is in fact a unique closest trace to ρ that satisfies φ, which also satisfies ψ, such that the counterfactual formula holds. Closeness of traces is modeled through a binary similarity relation Σ_a^ρ that defines whether some trace is at least as similar to ρ as another trace. Our approach is parametric for varying agent-specific similarity metrics, such that Σ_a^ρ depends on agent a. This, in particular, allows to model different internal causal models for different agents. We follow Lewis' formulation of counterfactuals and do not assume that there is a unique closest execution for every antecedent in a counterfactual, which is often termed the limit assumption and endorsed by the competing counterfactual theory of Stalnaker [51]. It has been noted in previous work that this assumption is easily violated when combining counterfactuals and temporal logics [18].

The combination of knowledge and counterfactual operators gives a specification like ICE the following semantics: it holds at a given position if there is a combination of attribute values α and β such that on all traces ρ that are indistinguishable for agent a, making the minimal changes to the trace such that $\alpha \wedge \beta$ holds results in a trace where agent a gets the offer. The nature of the minimal changes is defined by the similarity relation Σ_a^ρ, i.e., the internal causal model of agent a. Hence, ICE requires that on any trace agent a knows about some counterfactual explanation $\alpha \wedge \beta$ for the outcome *offer* whenever this outcome does not happen.

1.2 Expressivity and model checking

For the logic of the combined three modal systems, we construct a translation function that maps formulas to sentences in first-order logic of order with an equal-level predicate (FO[<,*E*]) [19]. This logic allows to quantify over tuples of traces and positions and hence it is a logic for hyperproperties [12]. Our translation serves two purposes. On the one hand, it

is a first result on the comparative expressiveness of this logic in relation to other hyperlogics, i.e., it places our logic for explainability into the hierarchy of hyperlogics [13]. On the other hand, it proves that model-checking formulas in this logic on finite-state multi-agent systems is decidable, and provides an algorithm via the proposed encoding into FO[<,*E*]. As far as we know, this is the first positive decidability result for model-checking of arbitrarily nested temporal and counterfactual operators, as our earlier study relegated counterfactuals to top-level operators [18]. Moreover, this previous work did not include knowledge operators which are necessary for formalizing explainability.

1.3 Contributions

In short, we make the following contributions.

- We define a combined logic of counterfactuals, knowledge and temporal modalities on the executions of multi-agent systems. This is an extension of our earlier work that did not consider knowledge operators [18].
- We formalize multiple notions of explainability in this logic, demonstrate practically how they distinguish explainable systems, and theoretically study their entailment relations.
- We outline a model-checking algorithm for this logic on multi-agent systems with a finite state-space. This algorithm relies on an encoding into the hyperlogic FO[<,*E*], which also yields some first insights into the comparative expressiveness of our presented logic.

2 Preliminaries

We recall some background on extended Kripke structures as models of multi-agent systems and on temporal, epistemic, and hyper logics as specification languages.

2.1 Multi-agent systems

We consider *Kripke structures* as the fundamental model of temporal logic. A Kripke structure $\mathcal{K} = (S, s_0, \Delta, AP, \Lambda)$ is a tuple, where S is a set of states, s_0 is the initial state, $\Delta : S \mapsto 2^S$ is a transition function such that $\Delta(s) \neq \emptyset$ for all states $s \in S$, AP is a set of *atomic propositions*, and $\Lambda : S \mapsto 2^{AP}$ is a function labeling states with atomic propositions. We call $\mathcal{K}$ a *finite* Kripke structure if the set of states S is finite. A *path* $\rho = \rho[0]\rho[1] \ldots \in S^\omega$ of a Kripke structure K is an infinite sequence of states such that the transition function is respected: $\rho[i+1] \in \Delta(\rho[i])$ for all $i \in \mathbb{N}$. The *trace* $\pi = \pi[0]\pi[1] \ldots \in (2^{AP})^\omega$ on a path ρ is the sequence of corresponding state labels, i.e., we have $\pi[i] = \Lambda(\rho[i])$ for all $i \in \mathbb{N}$. Let $\Pi(\mathcal{K})$ denote the set of traces on initial paths starting in s_0, i.e., on ρ such that $\rho[0] = s_0$. For some trace π, $\pi[0, n] \in S^*$ denotes its prefix of length $n+1$. We can extend a Kripke structure $\mathcal{K}$ with an *observation map* $\Omega : A \mapsto 2^{AP}$ to reason about the local observations of a set of agents A. For some agent $a \in A$, $\Omega(a)$ describes the set of atomic propositions that are observable to agent a. For some trace π of $\mathcal{K}$, $\Omega_a(\pi) \in (2^{AP})^\omega$ are the partial observations of a along the trace: $\Omega_a(\pi)[i] = \pi[i] \cap \Omega(a)$. We say $\mathcal{E} = (\mathcal{K}, \Omega)$ is finite if $\mathcal{K}$ is finite. The

set of traces of $\mathcal{E} = (\mathcal{K}, \Omega)$ is denoted $\Pi(\mathcal{E}) = \Pi(\mathcal{K})$. The set of extended Kripke structures that satisfy some logical formula φ is denoted by $Mod(\varphi)$.

2.2 Epistemic temporal logic

The basis of our logic is KLTL, which extends Linear Temporal Logic (LTL) [44] with a knowledge modality [17]. The syntax of KLTLis defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \bigcirc\varphi \mid \varphi \,\mathtt{U}\, \varphi \mid \mathtt{K}_a\, \varphi \mid \bigcirc^{-}\varphi \mid \mathtt{U}^{-}\varphi \ ,$$

where $p \in AP$ is an atomic proposition and $a \in A$ is an agent. Additionally, KLTLincludes the following derived operators: Boolean constants ($true$, $false$) and connectives ($\vee$, $\rightarrow$, $\leftrightarrow$), and the temporal operator 'Eventually' ($\diamond\varphi \equiv true\,\mathtt{U}\,\varphi$) as well as its dual, 'Globally' ($\Box\varphi \equiv \neg \diamond \neg\varphi$). The semantics of a KLTLformula φ with respect to an extended Kripke structure $\mathcal{E} = (\mathcal{K}, \Omega)$, a trace $\pi \in \Pi(\mathcal{K})$, and a position i is defined by the following satisfaction relation:

$$\begin{aligned}
&\mathcal{E}, \pi, i \vDash p && \text{iff } \ p \in \pi[i],\\
&\mathcal{E}, \pi, i \vDash \neg\varphi && \text{iff } \ \mathcal{E}, \pi, i \nvDash \varphi,\\
&\mathcal{E}, \pi, i \vDash \varphi_1 \vee \varphi_2 && \text{iff } \ \mathcal{E}, \pi, i \vDash \varphi_1 \vee \mathcal{E}, \pi, i \vDash \varphi_2,\\
&\mathcal{E}, \pi, i \vDash \bigcirc\varphi && \text{iff } \ \mathcal{E}, \pi, i+1 \vDash \varphi,\\
&\mathcal{E}, \pi, i \vDash \varphi_1\,\mathtt{U}\,\varphi_2 && \text{iff } \ \exists k \geq i : \mathcal{E}, \pi, k \vDash \varphi_2 \wedge \forall i \leq j < k : \mathcal{E}, \pi, j \vDash \varphi_1,\\
&\mathcal{E}, \pi, i \vDash \mathtt{K}_a\,\varphi && \text{iff } \ \forall \pi' \in \Pi(\mathcal{K}) : (\Omega_a(\pi)[0,i] = \Omega_a(\pi')[0,i]) \rightarrow \mathcal{E}, \pi', i \vDash \varphi.
\end{aligned}$$

Hence, an agent a has knowledge of some property φ, expressed through $\mathtt{K}_a(\varphi)$, iff this property holds on all observation-equivalent prefixes of the same length. These semantics of the knowledge modality $\mathtt{K}_a$ correspond to the so-called synchronous perfect recall semantics [27, 40], which means that agents gain knowledge through distinguishing prefixes of different length and based on divergence at any point in the past. System-level satisfaction is based on a universal application of the trace semantics: $\mathcal{E} = (\mathcal{K}, \Omega)$ satisfies φ, denoted by $\mathcal{E} \vDash \varphi$, iff for all traces $\pi \in \Pi(\mathcal{K}) : \mathcal{E}, \pi, 0 \vDash \varphi$. We denote the set of KLTLformulas over some alphabet AP by $\mathcal{L}_{\mathrm{KLTL}}(AP)$.

2.2.1 Past-operators

Since an explanation for some effect is usually found in its past, we use KLTLwith temporal past-operators. We define these operators as usual in the literature [38]. Given an extended Kripke structure $\mathcal{E} = (\mathcal{K}, \Omega)$, an initial trace $\pi \in \Pi(\mathcal{K})$, and a position i, we define the semantics of the past-operators as follows:

$$\begin{aligned}
&\mathcal{E}, \pi, i \vDash \bigcirc^{-}\varphi && \text{iff } \ i > 0 \wedge \mathcal{E}, \pi, i-1 \vDash \varphi,\\
&\mathcal{E}, \pi, i \vDash \varphi_1\,\mathtt{U}^{-}\varphi_2 && \text{iff } \ \exists k \leq i : \mathcal{E}, \pi, k \vDash \varphi_2 \ \wedge \forall i \geq j > k : \mathcal{E}, \pi, j \vDash \varphi_1 \ .
\end{aligned}$$

The 'Before' modality $\bigcirc^{-}$ refers to a previous time point, we define it such that it is trivially false at the start of a given trace. The 'Since' operator $\mathtt{U}^{-}$ is a mirror image of 'Until' ($\mathtt{U}$): It requires that φ_2 was true at some earlier time point k, and that φ_1 holds on all time

points in between. We also add the derived past operators 'Once' ($\diamond^- \varphi \equiv true \, \mathtt{U}^- \varphi$), as well as its dual, 'Historically' ($\square^- \varphi \equiv \neg \diamond^- \neg \varphi$).

3 In-depth example

We illustrate our approach for specifying explainability at the example of a simplified hiring system consisting of two agents: Applicant and Recruiter. The high-level idea is that, in every round, Recruiter chooses their preferred values for two attributes *job* and *gender*, and Applicant chooses the attribute values with which they apply in that round. Applicant gets an offer in some round if their attributes match Recruiter's preference. The hiring goes on infinitely, such that Applicant effectively models a stream of applicants applying at the company. The difference between the explainable and unexplainable version of the hiring system is that in the former, the preference of Recruiter is observable to Applicant, while it is hidden in the latter, unexplainable hiring system. Crucially, in both versions Applicant and Recruiter fix their attributes *concurrently*, such that in the explainable hiring scenario Applicant only observes Recruiter's preference *after* the decision. This means that the outcomes Applicant can enforce are the same in both scenarios, e.g., in neither scenario Applicant has a strategy that ensures that they will eventually get an offer. What is different, however, is that in the explainable hiring system Applicant gains knowledge on *why exactly* they did not get the offer in some round, while in the unexplainable system Applicant only knows that they should have done *something* differently.

3.1 Hiring system model

To develop this hiring example more formally, consider the following Kripke structure $\mathcal{K} = (S, s_0, \Delta, AP, \Lambda)$ underlying both the explainable and unexplainable hiring system. The set of states S is determined by the different values the attributes of Applicant and Recruiter may have:

$$S = \big\{(a_{job}, a_{gen}, r_{job}, r_{gen}) \mid a_{job}, r_{job} \in \{accounting, sales, it\} \\ \wedge \; a_{gen}, r_{gen} \in \{m, f\}\big\} \cup \{s_0\} \;\; ,$$

where s_0 is a unique initial state. Since every round is effectively a new, closed hiring process, the transition function Δ connects every state with itself and every other state, i.e., $\Delta(s) = S$ for all $s \in S$, so that the underlying graph is fully connected.

The set of atomic propositions resembles the attribute choices of the two agents. For some agent x we define the corresponding attributes with the function Att^+:

$$Att^+(x) = \{x_v \mid v \in \{accounting, sales, m, f\}\} \;\; , \\ AP = Att^+(a) \cup Att^+(r) \cup \{offer\} \;\; .$$

For the specifications we sometimes need both positive and negated atomic propositions for attributes, which is covered by the function Att:

$$Att(x) = Att^+(x) \cup \{\neg p \mid p \in Att^+(x)\} \;\; .$$

The labeling function Λ labels each state with the attributes picked by the agents, and with *offer* if they are matching. The initial state is labeled with the empty set:

$$\Lambda(s_0) = \{\}, \quad \Lambda((x, y, v, w)) = \{a_x, a_y, r_v, r_w\} \cup \{\mathit{offer} \mid x = v \wedge y = w\} \ .$$

The unexplainable hiring system $\mathcal{U} = (\mathcal{K}, \Omega^{\mathcal{N}})$ differs from the explainable one $\mathcal{E} = (\mathcal{K}, \Omega^{\mathcal{E}})$ only in the observation map. We have for an agent $x \in \{a, r\}$:

$$\Omega^{\mathcal{E}}(x) = AP = \mathit{Att}^{+}(a) \cup \mathit{Att}^{+}(r) \cup \{\mathit{offer}\}, \quad \Omega^{\mathcal{U}}(x) = \mathit{Att}^{+}(x) \cup \{\mathit{offer}\} \ .$$

Hence, in the explainable hiring system Applicant can observe the preferences of Recruiter (retrospectively), while in the unexplainable hiring system this information is hidden.

As discussed in Section 1 with reference to Fig. 1, the semantics of counterfactuals are defined with respect to a similarity relation Σ_a, which – intuitively speaking – encodes the minimal changes that are necessary to go from one trace to another, based on the intenral model of agent *a*. We now give a concrete similarity relation for the application scenario. Here, we only define a relation for Applicant as the properties we consider contain only counterfactuals indexed by *a*. A trace π_1 is *at least as similar* to the reference trace π at a given time point *i* as some other trace π_2 from the perspective of agent *a*, if the following formula holds at *i*, where pairs with the trace variables π, π_1, π_2 are used to refer to atomic propositions on a specific trace:

$$\begin{aligned}\Sigma(a)(\pi, \pi_1, \pi_2) = \Box\Big(\bigwedge_{p \in A}\big((p, \pi) \not\leftrightarrow (p, \pi_1)\big) \rightarrow \big((p, \pi) \not\leftrightarrow (p, \pi_2)\big)\Big) \wedge \\ \Box^{-}\Big(\bigwedge_{p \in A}\big((p, \pi) \not\leftrightarrow (p, \pi_1)\big) \rightarrow \big((p, \pi) \not\leftrightarrow (p, \pi_2)\big)\Big) \ ,\end{aligned}$$

where $A = \mathit{Att}^{+}(a) \cup \mathit{Att}^{+}(r)$. Detailed semantics of this relational property follow in Section 4.2. In the formula, the 'Historically' operator $\Box^{-}$ is the past-time version of the 'Globally' operator $\Box$, which imposes a constraint on all previous time points in a trace. Combined with the regular 'Globally' operator, the above formula expresses that the specified requirement does not only hold in the future but also in the past. Note that the past-operator's detailed semantics are given in Section 4. The specified requirement that is invariant in both past and future states that changes between the reference trace π and the at least as similar trace π_1 also have to be present in the less similar trace π_2. The changes between π and π_2 can be a proper superset of the changes between π and π_1, but it may also be that π_1 and π_2 are identical. Such subset-based similarity has been applied in many notions of causality [14, 15, 25]. We will see examples of tuples of traces that are in the similarity relation in the following. Take note that formulas encoding the similarity relation are KLTL formulas over a modified alphabet, i.e., the alphabet is $AP \times \Pi$ where Π is a set of trace variables. This is because the similarity relation needs to relate three traces with each other: it is a hyperproperty [12].

3.2 Semantics of explainability

To see how the explainability requirement specified by ICE (cf. Section 1) discriminates between these two hiring systems, consider the following infinite trace $\pi \in \mathcal{K}$, which is present in both systems:

$$\pi = \{\}\{a_{it}, a_f, r_{sales}, r_f\}\{\}^\omega \quad .$$

The ω-superscript indicates that this part of the trace is repeated infinitely often, i.e., in this case the trace ends up looping in the initial state. Let us now check whether trace π satisfies the requirement posed by ICE. Hence, we now check the semantics that we described abstractly in Section 1 with respect to Fig. 1 for this specific trace π. The ICE requirement states that at all positions where *offer* does not hold, the knowledge predicate $\mathtt{K}_a\left((\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}\right)$ has to hold for at least one pair of attributes $\alpha, \beta \in \mathit{Att}(a)$. By the semantics of K, this is the case if the counterfactual conditional $(\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}$ holds at this position on *all* traces that are indistinguishable for Applicant (cf. Subfigure 1a). Now, consider the second position of π. Here, *offer* does not hold. The set of traces with an observation equivalent prefix are all traces π' such that $\Omega_a(\pi)[0,2] = \Omega_a(\pi')[0,2]$. For the unexplainable hiring system $\mathcal{N}$ we can now show that, no matter which pair of attributes α, β and corresponding counterfactual conditional $(\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}$ we choose, there will always be an observation-equivalent trace such that the counterfactual does not hold (cf. Subfigure 1bfor the semantics of the counterfactual conditional). For example, assume we pick the pair a_{sales} and a_f, i.e., attributes for Applicant that match the preference of Recruiter on the second position of trace π. The counterfactual conditional $(a_{sales} \wedge a_f) \diamond\!\!\rightarrow_a \mathit{offer}$ does in fact hold on π at the second position, since there is the (unique) closest trace satisfying $a_{sales} \wedge a_f$ that also satisfies *offer*, namely:

$$\pi' = \{\}\{a_{sales}, a_f, r_{sales}, r_f, \mathit{offer}\}\{\}^\omega \quad .$$

However, there also exists an observation-equivalent trace such that the same counterfactual conditional does not hold. This is a trace where Recruiter picks a different preference at the second position. Since Recruiter's preference is unobservable by Applicant, this yields the following observation-equivalent trace

$$\pi'' = \{\}\{a_{it}, a_f, r_{accounting}, r_f\}\{\}^\omega \quad ,$$

that does not satisfy $(a_{sales} \wedge a_f) \diamond\!\!\rightarrow_a \mathit{offer}$, since the (unique) closest trace satisfying $a_{sales} \wedge a_f$ is:

$$\pi''' = \{\}\{a_{sales}, a_f, r_{accounting}, r_f\}\{\}^\omega \quad ,$$

where *offer* does not hold at the second position. The crux now is that in the unexplainable system we can find such an observation-equivalent trace for any counterfactual conditional in ICE's formula, since the preference of Recruiter is not observable by Applicant, and hence may be modified freely in observation-equivalent prefixes. In contrast, the same does not work in the explainable hiring system $\mathcal{E}$ since Applicant can observe Recruiters prefer-

ence and, hence, observation-equivalent traces are restricted to have the same preferences picked by Recruiter as in π. In particular, this means that π'' is not an observation-equivalent trace with respect to π in the explainable system.

3.3 Flavors of explainability

We have seen at the example of ICE how our logic uses the formalisms of counterfactual, epistemic and temporal logic to express a certain explainability requirement. Yet there are other conceivable notions of counterfactual explainability that can be specified in this logic.

3.3.1 Weak counterfactual explainability

It may, for instance, not be necessary that Applicant knows the exact attributes which would have resulted in an offer, but instead only that there were some attributes that would have let to an offer. This is specified by the following formula:

$$\Box\Big(\neg \mathit{offer} \rightarrow \mathtt{K}_a \left(\Big(\bigvee_{\alpha,\beta \in Att(a)} (\alpha \wedge \beta) \Big) \diamond\!\!\rightarrow_a \mathit{offer} \right) \Big) \ ,$$

which we term *Weak Counterfactual Explainability* (WCE). Based on the semantics of the knowledge operator, it is easy to see that ICE is the strictly stronger requirement. This yields the following proposition.

Proposition 1 ICE is strictly stronger than WCE, i.e., the models of ICE are a strict subset of WCE's models: $Mod(ICE) \subset Mod(WCE)$.

3.3.2 External counterfactual explainability

Both ICE and WCE require that Applicant is by themselves able to bring about the consequent of the counterfactual, and this is indeed the case in both the explainable and unexplainable hiring system presented in Section 3.1. This can be used to formalize *actionable* counterfactual explanations [45], i.e., counterfactual explanations that range over only attributes under the control of the agent receiving the explanation. However, consider an alteration of the explainable hiring system where Applicant cannot obtain the qualifications for accounting, while this may still be Recruiter's preference. Hence, formally we modify the state space to obtain the modified Kripke structure $\mathcal{K}'$ follows:

$$S' = \{(a_{job}, a_{gen}, r_{job}, r_{gen}) \mid r_{job} \in \{accounting, sales, it\} \wedge a_{job} \in \{sales, it\} \\ \wedge \ a_{gen}, r_{gen} \in \{m, f\}\} \cup \{s_0\} \ .$$

The resulting hiring system $\mathcal{E}' = (\mathcal{K}', \Omega^{\mathcal{E}})$ does not satisfy ICE, as, e.g., none of the counterfactuals in the formula hold at the second position of π'' as defined in Section 3.2. This is because only Recruiter can induce the necessary change by changing their preference. Since Recruiter's preference is observable to Applicant, it may still be reasonable to include explanations that Applicant can deduce from these observations, but may be out of their

control, i.e., *external* explanations. This yields the following criterion which we term *General Counterfactual Explainability* (GCE), which encompasses both internal and external explanations:

$$\square\Big(\neg \mathit{offer} \rightarrow \big(\bigvee_{\alpha,\beta \in Att(a,r)} \mathtt{K}_a \left((\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}\right)\big)\Big) \ ,$$

where $Att(a,r) = Att^*(a) \cup Att^*(r)$. Since the subformulas in the central disjunction of GCE subsume the ones present in ICE, it is again easy to deduce that the former is a strict relaxation of the latter. The strictness is witnessed by the modified hiring system $\mathcal{E}'$ discussed before.

Proposition 2 ICE is strictly stronger than GCE.

In this section, we have seen how our logic allows to formalize certain intricacies of different notions of explainability that pertain to questions such as: *Does an agent know which exact actions explain some observed outcome? And are these actions solely under the control of the agent, or dependent on other agents, too?* The logic provides an ideal basis to formalize these intricacies and construct a taxonomy of explainability that discriminates between, e.g., weak and internal explainability. In the following sections we introduce more such notions of explainability based on our logic.

4 A tri-modal logic for explainability

We now outline the formal semantics of the logic. We will use the shorthand YLTL to refer to the logic, which stands for *whY Linear-time Temporal Logic*. The structure of the Y also represents that the logic is a fusion of three modal logics. First, we present the syntax and semantics of YLTL. Afterward, we will study the model-checking problem of YLTL. We then outline a decision procedure for finite-state model checking based on translating YLTL formulas into FO[$<$,E].

4.1 Syntax

YLTL is an extension of KLTLwith the original counterfactual operators [36] and counterfactuals for non-total similarity relations [18]. This yields the following syntax for our logic YLTL:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \bigcirc\varphi \mid \varphi \,\mathtt{U}\, \varphi \mid \mathtt{K}_a\, \varphi \mid \qquad \text{(KLTL)}$$

$$\bigcirc^{-}\varphi \mid \mathtt{U}^{-}\varphi \mid \qquad \text{(past-operators)}$$

$$\varphi \,\square\!\!\rightarrow_a \varphi \mid \varphi \,\boxdot\!\!\rightarrow_a \varphi \qquad \text{(counterfactuals)}$$

where again $p \in AP$ is an atomic proposition and $a \in A$ is an agent. YLTL inherits all of the derived operators of KLTLwith past operators, as well as the counterfactual operators

'Might' ($\varphi \diamond\!\!\rightarrow_a \psi \equiv \neg(\varphi \,\Box\!\!\rightarrow_a \neg\psi)$), a dual to 'Would', and 'Existential Might', a dual to 'Universal Would' ($\varphi \,\diamondsuit\!\!\rightarrow_a \psi \equiv \neg(\varphi \,\boxdot\!\!\rightarrow_a \neg\psi)$).

We use Lewis' counterfactuals as predicates for causal reasoning because they are a common basis for a wide array of counterfactual causality definitions. While a more refined notion of causal predicates may be desirable, the literature is still divided on what refined notion generalizes to more than a few examples. Further, refined notions, such as actual causality, can often be encoded with counterfactuals [18]. We, therefore, hypothesize that an agent's desired explanation can always be expressed by Boolean combination of counterfactual dependencies with respect to the agent's similarity relation, which is covered by our logic. For instance, actual causality combines counterfactual reasoning with a minimality criterion [25]. We can require minimality of the counterfactual antecedent by enumerating all subformulas, i.e., for a specific antecedent $\alpha \wedge \beta$ we can extend the formula $(\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}$ in ICE to:

$$(\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer} \wedge \neg(\alpha \diamond\!\!\rightarrow_a \mathit{offer}) \wedge \neg(\beta \diamond\!\!\rightarrow_a \mathit{offer}) \ .$$

4.2 Semantics

YLTL inherits the semantics of all shared operators from KLTL, such that we only need to define the semantics of the past-operators and counterfactuals. Since the semantics of counterfactuals rely on a similarity-based analysis, we need to extend the extended Kripke structures of KLTLfurther to accommodate for the agent's similarity relations. Hence, the semantics of a YLTL formula is defined with respect to an similarity-extended Kripke structure $\mathcal{E}^+ = (\mathcal{K}, \Omega, \Sigma)$. Here, Σ denotes the *similarity map* $\Sigma : A \mapsto (\Pi \times \Pi \times \Pi \mapsto \mathcal{L}_{\mathrm{KLTL}}(AP \times \Pi))$ which provides a relational KLTLformula ranging over pairs of atomic propositions AP and trace variables Π.

4.2.1 Similarity map

The similarity map Σ defines the similarity relations of the different agents, each with a (relational) KLTLformula. We first define the *zipped trace* $z(\pi_1, \pi_2, \pi_3)$ of three traces $\pi_{1,2,3} \in (2^{AP})^\omega$ as follows for all $i \in \mathbb{N}$:

$$z(\pi_1, \pi_2, \pi_3)[i] = \{(a, \pi_k) \in AP \times \Pi \mid a \in \pi_k[i]\} \ .$$

The zipped trace simply fuses the three traces together while enriching the atomic propositions with the information on which trace they originate from. This now allows us to evaluate the formula obtained from the similarity map on the zipped trace, as a way to characterize the underlying similarity relation. We denote the similarity relation of some agent $a \in A$ as Σ_a, and define it as:

$$\Sigma_a = \{(\pi_1, \pi_2, \pi_3) \mid z(\pi_1, \pi_2, \pi_3) \vDash \Sigma(a)(\pi_1, \pi_2, \pi_3)\} \ .$$

Hence, three traces are related in the similarity relation of agent *a* if and only if their zipped trace satisfies the formula specified by the similarity map for agent *a*. We require the similarity relation to satisfy some assumptions, which we specify for the two place relation

$\Sigma_a^\pi = \{(\pi_1, \pi_2) \mid (\pi, \pi_1, \pi_2) \in \Sigma_a\}$ as in Lewis' original work [36]. Crucially, we allow Σ_a^π to be non-total like in our recent extension [18]. With this, we ensure that subset-based similarity relations like the one described in Section 3 and used, e.g., for actual causality [25], can be handled by our logic. We require Σ_a^π to be a preorder with π as a minimum: $\forall \pi' : (\pi, \pi') \notin \Sigma_a^\pi \rightarrow (\pi', \pi) \notin \Sigma_a^\pi$, i.e., if a trace is not at least as far from π as π itself, it is not related to π in Σ_a^π, and hence *inaccessible*. We can also use the similarity relation to encode Lewis' notion of inaccessibility by simply not relating inaccessible traces, as we have relaxed it to a non-total relation which allows such non-ordered pairs. In the following section we will outline the consequences this relaxation has on the semantics of the counterfactual operators, and how to alleviate these with two additional operators with modified semantics.

4.2.2 Counterfactuals

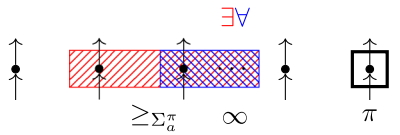
We can now proceed to specify the semantics of the counterfactual operators, for which we apply a similarity-based analysis [36]. Lewis defines counterfactuals as variably strict conditionals, which in multi-agent systems we interpret to mean that to hold on a specific trace, the consequent needs to hold in the closest accessible traces satisfying the antecedent. This now standard semantic treatment of counterfactuals in particular means that they cannot be expressed by a universal modality combined with a conditional, i.e., as Lewis argues, the semantics cannot be expressed with the usual universal modal operator. In our setting, this means that their semantics cannot be modeled with a knowledge operator and a conditional, i.e., $\mathtt{K}_a(\varphi_1 \rightarrow \varphi_2)$ is not equivalent to $\varphi_1 \mathrel{\square\!\!\rightarrow} \varphi_2$. Instead we define these counterfactuals in accordance with Lewis' original modal treatment. This results in the following semantics for a similarity-extended Kripke structure $\mathcal{E}^+ = (\mathcal{K}, \Omega, \Sigma)$, an initial trace $\pi \in \Pi(\mathcal{K})$, and a position i:

$$\begin{aligned}\mathcal{E}^+, \pi, i \vDash \varphi_1 \mathrel{\square\!\!\rightarrow}_a \varphi_2 \quad \text{iff } & (1)\ \forall \pi' \in \Pi(\mathcal{K}) : (\pi, \pi') \in \Sigma_a^\pi \rightarrow \mathcal{E}^+, \pi', i \nvDash \varphi_1 \ \vee \\ & (2)\ \exists \pi' \in \Pi(\mathcal{K}) : (\pi, \pi') \in \Sigma_a^\pi \wedge \mathcal{E}^+, \pi', i \vDash \varphi_1 \ \wedge \\ & \forall \pi'' \in \Pi(\mathcal{K}) : (\pi'', \pi') \in \Sigma_a^\pi \rightarrow \mathcal{E}^+, \pi'', i \vDash (\varphi_1 \rightarrow \varphi_2) \ .\end{aligned}$$

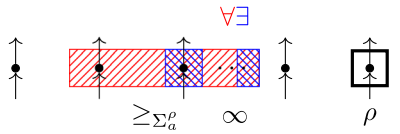
Condition (1) represents a vacuity condition such that the 'Would' counterfactual holds on a trace π if there are no accessible traces where the antecedent φ_1 holds. It is easy to see how the quantification is restricted to traces that are related to π in the similarity relation, i.e., traces that are accessible from π, through the implication after each universal quantifier and the conjunction after the existential quantifier. Condition (2), in principle, encodes the idea that on all closest counterfactual traces where φ_1 holds, φ_2 holds as well.

4.2.2.1 Infinite chains of closer traces. The complex nested quantification comes into play when there is no unique closest trace for some antecedent, as illustrated in the following example with an infinitely descending chain of progressively more similar traces.

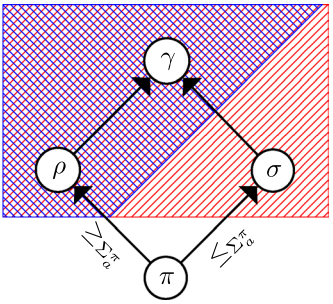
Example 1 Consider the same similarity relation as used in the example from Section 3, where $A = Att^+(a) \cup Att^+(r)$:



(a) Semantics of the counterfactual $\varphi \,\square\!\!\rightarrow_a \psi$.



(b) Semantics of the counterfactual $\varphi \,\diamond\!\!\rightarrow_a \psi$.



(c) Non-total similarity relation.

Fig. 2 Lewis' original semantics for the counterfactuals $\square\!\!\rightarrow_a$ and $\diamond\!\!\rightarrow_a$ are illustrated in Subfigures 2aand 2a, respectively. Arrows and point depict traces that are ordered in ascending similarity to π and ρ, respectively, according to the similarity relation $\geq_{\Sigma_a}$. Subfigure 2chighlights problems when evaluating the counterfactual $\varphi \,\square\!\!\rightarrow_a \psi$ in a non-total similarity relation. Here, circles represent full traces such as ϕ or γ, while arrows indicate that two traces are ordered by the similarity relation $\leq_{\Sigma_a^\pi}$. In all subfigures, areas with diagonal lines (colored red) indicate that the covered traces satisfy φ, while crossed lines (colored red and blue) indicate that the traces satisfy ψ

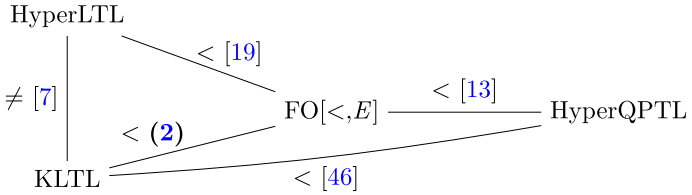


Fig. 3 KLTL's exact place in the hierarchy of hyperlogics. The result of Theorem 2 is highlighted in bold

$$\Sigma(a)(\pi,\pi_1,\pi_2) = \square\Big(\bigwedge_{p\in\mathcal{A}}\big((p,\pi) \not\leftrightarrow (p,\pi_1)\big) \rightarrow \big((p,\pi) \not\leftrightarrow (p,\pi_2)\big)\Big) \wedge$$
$$\square^-\Big(\bigwedge_{p\in\mathcal{A}}\big((p,\pi) \not\leftrightarrow (p,\pi_1)\big) \rightarrow \big((p,\pi) \not\leftrightarrow (p,\pi_2)\big)\Big) \ ,$$

and the trace $\{p\}^\omega$. We are interested in the counterfactual $(\neg\square \diamond p) \,\square\!\!\rightarrow_a \top$, in a structure that contains all traces over the alphabet $\{p\}$. In this situation, we have an infinite chain of traces that satisfy $(\neg\square \diamond p)$, i.e., $\{\}^\omega$, $\{p\}\{\}^\omega$, $\{p\}\{p\}\{\}^\omega$, etc. Hence, we cannot evaluate the consequent in a particular unique closest counterfactual trace, but instead need to make use of Lewis' elegant semantics for counterfactuals without the so-called limit assumption: We are looking for an accessible threshold trace π', such that all at least as close traces π'' that satisfy the antecedent also satisfy the consequent.

Figure 2 abstractly illustrates these semantics on infinite chains of closer traces. In Subfigure 2a, we can see how the counterfactual $\varphi \,\square\!\!\rightarrow_a \psi$ requires a continuous chain of traces satisfying ψ as soon as we move up the similarity relation from traces that satisfy $\neg\varphi$ to traces that satisfy φ, starting from the reference trace π. This is realized through the $\exists\forall$ -quantifier alternation that requires a trace satisfying φ such that all closer traces satisfying

φ also satisfy ψ. In contrast, the counterfactual $\varphi \diamond\!\!\rightarrow_a \psi$ requires for all traces satisfying φ at least one closer trace satisfying ψ – even on infinite chains. This is realized through a $\forall\exists$ -quantifier alternation and depicted in Subfigure 2b

4.2.2.2 Non-total similarity relations Unlike in Lewis' original account, we allow a similarity relation Σ_a^π of some agent a to be non-total. As a consequence, Lewis' original semantics yield some rather unintuitive inferences [18], which we illustrate in the following example.

Example 2 Consider again the similarity relation as defined in Section 3 and recalled in the previous Example and the trace $\pi = \{\}^\omega$, with the counterfactual $(p \vee q) \,\square\!\!\rightarrow_a p$, in a structure that contains all traces over the alphabet $\{p, q\}$. We depict π and three other traces with their comparative similarity in Subfigure 2c. The counterfactual is satisfied by π, as we have the closest counterfactual trace $\rho = \{p\}\{\}^\omega$ as a witness for the existential quantifier in Condition 2 of the semantics of $\square\!\!\rightarrow_a$. However, this does not match the intended semantics of the 'Would' counterfactual. The counterfactual is supposed to express that the consequent p holds on all closest counterfactual traces. However, there is the closest counterfactual trace $\sigma = \{q\}\{\}^\omega$ that does not satisfy p.

The problem with Lewis' original semantics in non-total similarity relations is that the existential quantifier in Condition 2 implicitly also automatically quantifies existentially over the unrelated chains of at least as similar traces in the similarity relation. In Example 2, these (in this case finite, but in general possibly infinite) chains are, on the one chain, the trace changing p and, on the other chain, the trace changing q. These traces with single changes are incomparable with each other regarding their similarity to the reference trace $\{\}^\omega$, while the trace that changes both p and q is comparable to both of these traces that change only single atomic propositions (it is, of course, less similar to $\{\}^\omega$ than both of the traces). However, since the trace with the single changes already satisfy the antecedent of the counterfactual, they have to be considered as possible threshold traces for Lewis' criterion. However, the implicit existential quantification allows the semantics to ignore whole chains (in this case the chain with $\{q\}\{\}^\omega$), which then do not need a threshold trace satisfying Lewis' criterion.

In earlier work [18], we proposed an alternative counterfactual operator which we include in YLTL. The operator is called 'Universal Would' counterfactual, because the semantics are based on universal quantification over the chains of the similarity relation as follows, again for a similarity-extended Kripke structure $\mathcal{E}^+ = (\mathcal{K}, \Omega, \Sigma)$, an initial trace $\pi \in \Pi(\mathcal{K})$, and a position i:

$$\begin{aligned}\mathcal{E}^+, \pi, i \vDash \varphi_1 \boxdot\!\!\rightarrow_a \varphi_2 \;\text{ iff }\; & (1)\ \forall \pi' \in \Pi(\mathcal{K}) : (\pi, \pi') \in \Sigma_a^\pi \wedge \mathcal{E}^+, \pi', i \vDash \varphi_1 \;\rightarrow \\ & (2)\ \exists \pi'' \in \Pi(\mathcal{K}) : (\pi'', \pi') \in \Sigma_a^\pi \wedge \mathcal{E}^+, \pi'', i \vDash \varphi_1 \wedge \\ & \forall \pi''' \in \Pi(\mathcal{K}) : (\pi''', \pi'') \in \Sigma_a^\pi \rightarrow \mathcal{E}^+, \pi''', i \vDash (\varphi_1 \rightarrow \varphi_2) \ .\end{aligned}$$

Intuitively, this operator lifts Lewis' semantics of the 'Would' operator to non-total similarity relations by applying it to every chain in the relation. This is achieved by prepending the non-vacuous condition of Lewis' definition for $\square\!\!\rightarrow$, i.e., Condition 2, with a universal

quantification that effectively quantifies over chains of traces (Condition 1). The semantics of Lewis' vacuity condition is then also directly captured by the initial universal quantification and the implication, such that we do not need the same disjunction as in Lewis' definition for $\Box\!\!\rightarrow$. For every chain with at least one counterfactual world not satisfying φ_1, the same requirement as posed by Lewis' original 'Would' counterfactual has to hold: the threshold trace is π'' bound by the existential quantifier, and this threshold trace has to be found on the same chain that the universally quantified π' is on. Consequently, there has to be a threshold trace on every chain containing a trace that satisfies φ_1 that is accessible from π. Local vacuity is still allowed, i.e., a whole chain without a single trace satisfying φ_1 does not need a closest trace satisfying φ_2.

Example 3 To see how this semantics fixes the problem raised in Example 2, consider again the trace $\{\}^\omega$, with the counterfactual $(p \vee q) \Box\!\!\rightarrow_a p$, in a structure that contains all traces over the alphabet $\{p, q\}$ and under the same similarity relation used in Section 3 (with this new alphabet). The problem is that Lewis' existential quantifier allowed us to choose between the traces $\{p\}\{\}^\omega$ and $\{q\}\{\}^\omega$ as a witnessing counterfactual world. However, if we use the stronger 'Universal Would' operator $(p \vee q) \boxdot\!\!\rightarrow_a p$, the universal quantifier requires us to find a witnessing counterfactual operator on every chain. Since there is no at least as close trace π'' that satisfies p for the trace $\{q\}\{\}^\omega$, we have that $(p \vee q) \boxdot\!\!\rightarrow_a p$ is not satisfied on the trace $\{\}^\omega$ in this scenario. This is as desired, because p does not hold on all of the closest traces satisfying $p \vee q$.

4.2.3 Agent-specific similarity

A key feature of our counterfactuals is that their semantics are defined with respect to a *specific* agent's similarity relation. This represents that agents may have different internal models about the causal workings of the system. For instance, in the hiring system described in Section 3 it may be sensible that Applicant does not consider counterfactual scenarios where their gender attribute is different from the actual trace. An explanation based on such counterfactuals would not be actionable [45], and may hence be undesired in many cases. Actionable explanations range only over actions and attributes that are fully under control of the agent receiving the explanations, which clearly is not the case for the gender attribute. Whether an antecedent is actionable or not highly depends on the scenario and the agent at hand, which motivates our flexible formalism of agent-specific similarity relations. With YLTL, such requirements can then be encoded by making, e.g., the traces with a modified atomic proposition a_{gen} inaccessible from the reference trace in the similarity map Σ' for agent *a* as follows:

$$\Sigma'(a)(\pi, \pi_1, \pi_2) = \Sigma(a)(\pi, \pi_1, \pi_2) \wedge \Box^-(a_{gen}, \pi) \leftrightarrow (a_{gen}, \pi_1) \wedge (a_{gen}, \pi) \leftrightarrow (a_{gen}, \pi_2) \\ \wedge \Box(a_{gen}, \pi) \leftrightarrow (a_{gen}, \pi_1) \wedge (a_{gen}, \pi) \leftrightarrow (a_{gen}, \pi_2) \ .$$

Here, we use the previous similarity relation $\Sigma(a)(\pi, \pi_1, \pi_2)$ defined in Section 3.1. Recall that the idea of that relation was that the changes between the actual trace π and the closer trace π_1 are a subset of the changes between the actual trace π and the farther trace π_2. $\Sigma'(a)(\pi, \pi_1, \pi_2)$ now requires a_{gen} to be the same on all three traces, which means traces of

the system that change a_{gen} are not related, hence inaccessible. This makes explainability specifications such as ICE harder to satisfy.

Yet, other agents such as Recruiter may still consider counterfactual traces where the attribute a_{gen} is modified, i.e., their similarity relation is $\Sigma'(r) = \Sigma(a)$. As a result we have that the explainable system $\mathcal{E} = (\mathcal{K}, \Omega^{\mathcal{E}}, \Sigma')$ where all atomic propositions are observable by both agents does not satisfy ICE from the point of view of Applicant, but does satisfy ECE, i.e., *External Counterfactual Explainability*, from the point of view of Recruiter, where ECE is formalized as follows:

$$\square\Big(\neg \mathit{offer} \rightarrow \big(\bigvee_{\alpha,\beta \in Att(a)} \mathtt{K}_r \left((\alpha \wedge \beta) \diamond\!\!\rightarrow_r \mathit{offer}\right)\big)\Big) \ .$$

Note that since both agents can observe the same atomic propositions, and hence $\mathtt{K}_a$ and $\mathtt{K}_r$ are in principle interchangeable, this difference is completely due to the fact that there are some $\alpha, \beta \in Att(a)$ for every position i such that the counterfactual conditional $(\alpha \wedge \beta) \diamond\!\!\rightarrow_a \mathit{offer}$ holds, while this is not the case for the counterfactual $(\alpha \wedge \beta) \diamond\!\!\rightarrow_r \mathit{offer}$ that refers to the similarity relation of Recruiter.

4.3 Model checking

In this section, we develop an approach to automatically verify whether a given system satisfies a YLTL specification. Our results apply to systems defined by finite similarity-extended Kripke structures. Under this assumption, we can then show the decidability of the YLTL model-checking problem by reducing it to model checking of an equivalent formula in Extended Monadic First-Order Logic (FO[<,*E*]). This is a decidable problem [13, 19], and we now outline this logic as a preliminary.

4.3.1 Extended monadic first-order logic

FO[<,*E*] is the monadic first-order logic of order (FO[<]) extended with the equal-level predicate E [19] for expressing hyperproperties [12], i.e., properties that relate multiple executions of a system to one another. For a predefined set V of first-order variables, the syntax of FO[<,*E*] is defined by the following grammar:

$$\begin{aligned} \varphi &::= \psi \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x.\ \varphi \\ \psi &::= P_p(x) \mid x < y \mid x = y \mid E(x,y) \ , \end{aligned}$$

where $p \in AP$ is an atomic proposition and $x, y \in V$ are first-order variables. An FO[<,*E*] formula is *closed* when all variables are bound by a quantifier. FO[<,*E*] formulas are interpreted over a set of traces Π. The first-order variables range over the domain $\Pi \times \mathbb{N}$. The order $<$ is now only interpreted over variables referring to the same trace: $< ::= \{((\pi, n_1), (\pi, n_2)) \in (\Pi \times \mathbb{N})^2 \mid n_1 < n_2\}$. The equal-level predicate holds if two variables refer to the same position in (possibly) two different traces: $E ::= \{((\pi_1, n), (\pi_2, n)) \in (\Pi \times \mathbb{N})^2\}$. The predicate P_p encodes the truth-value of atomic propositions: $P_p ::= \{(\pi, n) \mid p \in \pi[n]\}$. We say that a closed FO[<,*E*] formula φ is satis-

fied by an extended Kripke structure $\mathcal{E}$, denoted by $\mathcal{E} \vDash \varphi$, iff φ interpreted over $\Pi(\mathcal{K})$ is true.

We now outline our result on YLTL model checking. This utilizes a translation function *fo* described in the proof of the following Lemma. The translation mirrors the idea used to translate LTL into first-order logic defined in Kamp's seminal theorem [30], which we extend for the knowledge operator and the counterfactuals, and for this we use the equal-level predicate provided by FO[<,*E*].

Lemma 1 *For every YLTL formula φ there exists a formula in FO[<,E] φ' that characterizes the same set of models, i.e., such that $Mod(\varphi) = Mod(\varphi')$.*

Proof The proof relies on a linear translation *fo* from YLTL to FO[<,*E*]. We will use syntactic sugar for successors and minimal positions: $succ(x, y) ::= x < y \wedge \neg\exists z.\, x < z < y$ and $min(x) ::= \neg\exists y.\, succ(y, x)$. In the end, the FO[<,*E*] formula proving our claim is obtained from φ by

$$fo(\varphi) ::= \forall x_0.\, min(x_0) \rightarrow fo(\varphi, x_0) \quad . \tag{1}$$

The FO[<,*E*] formula $fo(\varphi, x_0)$ is constructed inductively based on the current sub-formula of the YLTL formula φ (ranging over a set AP) and the current time-point of interest encoded in the second argument, which is initially x_0 but may change through trace quantification from, e.g., epistemic operators. Recall that the first-order variables of FO[<,*E*] are in fact tuples $(\pi, n) \in \Pi \times \mathbb{N}$ of a trace variable and a position, let us denote for some tuple $x_t = (\pi, n)$: $x_t|_1 = \pi$ and $x_t|_2 = n$ for projecting to the components of the tuple. Note that φ technically includes atomic propositions from the set $AP \cup (AP \times \Pi)$ since we also need to translate the KLTLformulas obtained from the similarity map, which range over tuples of atomic propositions and trace variables. These tuples from $AP \times \Pi$ are unrelated to the first-order variables $(\pi, n) \in \Pi \times \mathbb{N}$ and become relevant only when translating counterfactual operators. We start with the simpler cases, for which the construction of the FO[<,*E*] formula is as follows.

$$
\begin{aligned}
&fo((p, \pi), x_t) \;=\; P_p((\pi, x_t|_2))\\
&fo(p, x_t) \;=\; P_p(x_t)\\
&fo(\neg\varphi, x_t) \;=\; \neg fo(\varphi, x_t)\\
&fo(\varphi_1 \vee \varphi_2, x_t) = \; fo(\varphi_1, x_t) \vee fo(\varphi_2, x_t)\\
&fo(\bigcirc\varphi, x_t) = \; \exists x_t^+.\, succ(x_t, x_t^+) \wedge fo(\varphi, x_t^+)\\
&fo(\bigcirc^-\varphi, x_t) = \; \exists x_t^-.\, succ(x_t^-, x_t) \wedge fo(\varphi, x_t^-)\\
&fo(\varphi_1 \,\mathtt{U}\, \varphi_2, x_t) = \; \exists x_2 \geq x_t.\, fo(\varphi_2, x_2) \;\wedge (\forall x_1.\, x_t \leq x_1 < x_2 \rightarrow fo(\varphi_1, x_1))\\
&fo(\varphi_1 \,\mathtt{U}^-\varphi_2, x_t) = \exists x_2^- \leq x_t.\, fo(\varphi_2, x_2^-) \;\wedge (\forall x_1^-.\, x_t \geq x_1^- > x_2 \rightarrow fo(\varphi_1, x_1^-))\\
&fo(\mathtt{K}_a\, \varphi, x_t) = \; \forall x_e.\, E(x_e, x_t) \wedge (\forall x_e^-, x_t^-.\, x_e^- \leq x_e \wedge\; x_t^- \leq x_t \wedge E(x_e^-, x_t^-) \rightarrow\\
&\quad \bigwedge_{p \in \Omega(a)} P_p(x_e^-) \leftrightarrow P_p(x_t^-)) \rightarrow fo(\varphi, x_e)
\end{aligned}
$$

The most involved formulas are obtained from translating the counterfactual operators. Note that the expression $\Sigma(a)(x_t|_1, x_t|_1, x_e|_1), x_t)$ that appears throughout the formulas simply denotes the KLTLformula characterizing the similarity relation of agent a, where the parameters are in this case instantiated by the trace variables of x_t (twice) and of x_e. This double instantiation results from encoding accessibility via the similarity relation. The translation for the counterfactuals proceeds as follows.

$$\begin{aligned}
fo(\varphi_1 \,\square\!\!\rightarrow_a \varphi_2, x_t) \;=\; & (\forall x_e.\, E(x_e, x_t) \wedge fo(\Sigma(a)(x_t|_1, x_t|_1, x_e|_1), x_t) \rightarrow \neg fo(\varphi_1, x_e)) \\
& \vee \exists x_e. E(x_e, x_t) \wedge fo(\Sigma(a)(x_t|_1, x_t|_1, x_e|_1), x_t) \;\wedge fo(\varphi_1, x_e) \\
& \wedge \forall x_c. fo(\Sigma(a)(x_t|_1, x_c|_1, x_e|_1), x_t) \rightarrow (fo(\varphi_1, x_c) \rightarrow fo(\varphi_2, x_c)) \\
fo(\varphi_1 \,\boxdot\!\!\rightarrow_a \varphi_2, x_t) \;=\; & (\forall x_a.\, E(x_a, x_t) \wedge fo(\Sigma(a)(x_t|_1, x_t|_1, x_a|_1), x_t) \wedge fo(\varphi_1, x_a) \\
& \rightarrow \exists x_e. E(x_e, x_a) \wedge fo(\Sigma(a)(x_t|_1, x_e|_1, x_a|_1), x_t) \wedge fo(\varphi_1, x_e) \\
& \wedge \forall x_c. fo(\Sigma(a)(x_t|_1, x_c|_1, x_e|_1), x_t) \rightarrow (fo(\varphi_1, x_c) \rightarrow fo(\varphi_2, x_c))
\end{aligned}$$

Note that this function is well-defined only because we do not allow formulas from the similarity map to themselves include counterfactuals. Otherwise, the translation function could include a circular dependency where translating a counterfactual operator requires translating a similarity relation which in turn again requires translating a counterfactual and so on.

In the end, the equivalence between $fo(\varphi)$ and φ (cf. Equation 1) can be shown by structural induction over φ. □

While Lemma 1 alone is just a statement about comparative expressiveness, it also indirectly provides us with an algorithm that given a system as a finite Kripke structure and a specification as a YLTL formula automatically verifies whether the systems satisfies the formula. This is quite remarkable as previous results for model checking logics that combine temporal operators and counterfactuals only considered counterfactuals as top-level operators [18]. Compared to this work, we lose the ability to express ω-regular temporal properties, but gain the ability to nest counterfactuals and temporal operators, and additionally include knowledge operators. While nesting counterfactuals is mostly of theoretical interest, the additional knolwedge operators are crucial for expressing explainability. To the best of our knowledge, we present the first algorithm for model checking a logic that combines temporal operators and counterfactuals arbitrarily.

Theorem 1 *(YLTL Model Checking) There is an algorithm that, given a finite extended Kripke structure $\mathcal{E}$ and a YLTL formula φ, checks whether $\mathcal{E} \vDash \varphi$.*

Proof In Lemma 1 we have shown that we can construct an equivalent FO[<,E]-formula φ' for the YLTL formula φ. Since FO[<,E] is strictly less expressive than HyperQPTL (LTL with trace and propositional quantifiers) [13], and there is a model-checking algorithm for HyperQPTL [46], the claim follows immediately. □

The existence of a model-checking algorithm is what makes our logic useful in practice: Not only is it possible to express several notions of explainability, it is also possible to automatically verify them. While an exact complexity analysis of model checking YLTL is out of scope of this paper, it should be noted that the complexity of the current encoding is non-elementary, with the tower of exponents scaling with the number of nested counterfac-

tuals and knowledge operators. However, this is not worse than the complexity of model checking just knowledge and temporal operators [8]. Moreover, in practice, we are mostly concerned with formulas that have only a few nested operators, as is the case for all of the explainability requirements formalized in this work.

4.4 Side result on the expressiveness of KLTL

Besides providing an algorithm for model checking YLTL formulas, Lemma 1 also includes a translation from KLTLto FO[<,*E*] that was loosely described earlier by Hofmann [28]. This translation shows that FO[<,*E*] subsumes KLTL. We now outline how we can combine this with a result from Bozzelli et al. [7] to show that KLTLis strictly less expressive than FO[<,*E*]. For completeness, we also recall some results regarding the comparative expressiveness of KLTLand HyperLTL (LTL with quantification over traces), as well as HyperQPTL (HyperLTL with propositional quantifiers). As a reference, the syntax of HyperQPTL is build according to the following grammar:

$$\psi ::= \exists \pi.\, \psi \mid \forall \pi.\, \psi \mid \exists p.\, \psi \mid \forall p.\, \psi \mid \varphi' \quad ,$$

where p is a fresh atomic proposition and π is a trace variable. φ' is an LTL formula, i.e., build according to the grammar of KLTL(Section 2.2) without the knowledge operator and past-time operators. The syntax of HyperLTL can be obtained by removing $\forall p.\, \psi$ and $\exists p.\, \psi$ from the above grammar of HyperQPTL.

Previously, Bozzelli et al. [7] have shown that KLTL's expressiveness is incomparable to the expressiveness of HyperLTL. Rabe [46] showed that KLTLcan be encoded in HyperQPTL and Hofmann described that this encoding can be adapted for FO[<,*E*], which we have confirmed in the proof of Lemma 1. Combined, these results mean that KLTLlies strictly further down in the hierarchy of hyperlogics.

Theorem 2 FO[<,*E*] is strictly more expressive than KLTL.

Proof Lemma 1 shows that FO[<,*E*] is at least as expressive as YLTL, and since YLTL subsumes KLTLtrivially, it follows that FO[<,*E*] is at least as expressive as KLTL. It therefore only remains to show strictness. Strictness follows from previous results: (1) the proof that KLTLdoes not subsume HyperLTL presented by Bozzelli et al. [7] and (2) by the subsumption of HyperLTL through FO[<,*E*] shown by Coenen et al. [13], as follows: (1) Bozzelli et al. provide the following HyperLTL formula that cannot be expressed in KLTL:

$$\varphi_H = \exists \pi.\, \exists \pi'.\, p_\pi \,\mathtt{U}\, \big((p_\pi \wedge \neg p_{\pi'}) \wedge \bigcirc \Box (p_\pi \leftrightarrow p_{\pi'})\big) \quad .$$

The intuition behind their proof is that KLTLcannot compare two different traces at an unbounded number of positions. We refer to the full version [6] of Bozzelli et al.'s paper for the detailed proof. With (2), we know that there exists an FO[<,*E*] formula φ_{fo} that is equivalent to φ_H. φ_{fo} is then not expressible in KLTL, which proves the claimed strictness of the inclusion. □

5 Related work

There is a long line of works on combining modal logics [11]. In this section we focus only on works related to combinations of counterfactual, epistemic and temporal operators, which have been combined in pairs for a variety of applications. A connection between knowledge and counterfactual dependencies in the situation calculus has been drawn by Khan and Lespérance [31]. This has been extended to define *explanations* for agent behavior [32], in particular accounting for theory-of-mind reasoning. Contrary to these works, we focus on *explainability* as a system property and provide an approach for verification, but we also appeal to theory-of-mind reasoning with our agent-specific similarity relations, which allow to model the internal mental states of the agents. Knowledge and causality have been combined to reason about deceptive AI [48]. Counterfactuals and the knowledge modality have also been combined to express hypothetical knowledge [23] and rationality [49, 52] in game theory. Liu and Lorini [39] study modal logics for defining individual explanations for classifiers, and Aguilera-Ventura et al. [1] have recently studied grounding similarity relations for counterfactuals.

Besides counterfactual epistemic logics, our work also builds on a long line of research into logics that reason about knowledge and time, which originated in the analysis of distributed protocols [17, 34] and have been applied to a variety of applications such as information-flow security [4, 24, 41], as well as knowledge-based programs [42]. Counterfactual and temporal reasoning has been combined to reason about temporal aspects of causality [10, 15, 20, 54].

Our work studies the epistemics of explainability and abstracts away from questions such as how to visualize explanations, and what explanations are relevant for a human user in a given context. There is a variety of works that study these orthogonal questions [9, 29, 33, 35, 43, 50]. Moreover, there are several works on generating explanations for more complex system architectures [3, 16].

6 Conclusion & Outlook

We have studied a logic that combines the long-studied modal operators of counterfactual, epistemic and temporal logics for specification and verification of explainability requirements. We have demonstrated how the logic can be used to define the first formal taxonomy of counterfactual explainability that encompasses the notions of internal, external, general, and weak explainability. We believe this aspect of our study can be spun much further by introducing additional features to the logic, for instance minimality constraints on counterfactual antecedents [18], or by considering combinations of counterfactual and probabilistic reasoning [54] as explanatory properties. As another aspect, we have proven that the YLTL model-checking problem is decidable for finite-state multi-agent systems. We plan on building on this result by developing practical model-checking tools for explainability requirements. On the theoretical side, we have made first steps toward analyzing the expressivity of the combined logic in relation to other hyperlogics. These are also, to the best of our knowledge, the first results on model checking and expressivity of counterfactual operators when combined arbitrarily with temporal operators. Building on these results, we have recently proposed an approach for analyzing explainability and privacy tradeoffs in multi-

agent systems, which uses a second-order version of YLTL to enable quantification over arbitrary counterfactual antecedents [21].

Author contributions B.F. and J.S. developed the idea for the paper in joint discussions. J.S. wrote the main manuscript text with continuous consultation by B.F.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Aguilera-Ventura, C., Herzig, A., Liu, X., et al.: Counterfactual reasoning via grounded distance. In: Marquis P, Son TC, 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, pp 2–11, https://doi.org/10.24963/kr.2023/1 (2023)
2. Almagor, S., Lahijanian, M.: Explainable multi agent path finding. In: Seghrouchni AEF, Sukthankar G, An B, et al (eds) Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020. International Foundation for Autonomous Agents and Multiagent Systems, https://doi.org/10.5555/3398761.3398771 (2020)
3. Audemard, G., Koriche, F., Marquis, P.: On tractable XAI queries based on compiled representations. In: Calvanese D, Erdem E, Thielscher M (eds) Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, September 12-18, 2020, pp 838–849, https://doi.org/10.24963/kr.2020/86 (2020)
4. Balliu, M., Dam, M., Guernic, G.L.: Epistemic temporal logic for information flow security. In: Askarov A, Guttman JD (eds) Proceedings of the 2011 Workshop on Programming Languages and Analysis for Security, PLAS 2011, San Jose, CA, USA, 5 June, 2011. ACM, p 6, (2011). https://doi.org/10.1145/2166956.2166962
5. Beckers, S.: Causal explanations and XAI. In: Schölkopf B, Uhler C, Zhang K (eds) 1st Conference on Causal Learning and Reasoning, CLeaR 2022, Sequoia Conference Center, Eureka, CA, USA, 11-13 April, 2022, Proceedings of Machine Learning Research, vol 177. PMLR, pp 90–109, (2022). https://proceedings.mlr.press/v177/beckers22a.html
6. Bozzelli, L., Maubert, B., Pinchinat, S.: Unifying hyper and epistemic temporal logic. CoRR abs/1409.2711, (2014). http://arxiv.org/abs/1409.2711
7. Bozzelli, L., Maubert, B., Pinchinat, S.: Unifying hyper and epistemic temporal logics. In: Pitts AM (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, Lecture Notes in Computer Science, vol 9034. Springer, pp 167–182, (2015). https://doi.org/10.1007/978-3-662-46678-0_11
8. Bozzelli, L., Maubert, B., Murano, A.: On the complexity of model checking knowledge and time. ACM Trans Comput Log **25**(1), 8:1-8:42 (2024). https://doi.org/10.1145/3637212

9. Brandao, M., Mansouri, M., Mohammed, A., et al.: Explainability in multi-agent path/motion planning: User-study-driven taxonomy and requirements. In: Faliszewski P, Mascardi V, Pelachaud C, et al (eds) 21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), pp 172–180, (2022). https://doi.org/10.5555/3535850.3535871, https://www.ifaamas.org/Proceedings/aamas2022/pdfs/p172.pdf
10. Carelli, M., Finkbeiner, B., Siber, J.: 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 (2025)
11. Carnielli, W., Coniglio, M.E.: Combining logics. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy, Fall, 2020th edn. Stanford University, Metaphysics Research Lab (2020)
12. Clarkson, M.R., Schneider, F.B.: Hyperproperties. J Comput Secur **18**(6), 1157–1210 (2010). https://doi.org/10.3233/JCS-2009-0393
13. Coenen, N., Finkbeiner, B., Hahn, C., et al.: The hierarchy of hyperlogics. In: 34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019. IEEE, pp 1–13, (2019). https://doi.org/10.1109/LICS.2019.8785713
14. Coenen, N., Dachselt, R., Finkbeiner, B., et al.: Explaining hyperproperty violations. In: Shoham S, Vizel Y (eds) Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part I, Lecture Notes in Computer Science, vol 13371. Springer, pp 407–429, , (2022a). https://doi.org/10.1007/978-3-031-13185-1_20
15. Coenen, N., Finkbeiner, B., Frenkel, H., et al.: Temporal causality in reactive systems. In: Bouajjani A, Holík L, Wu Z (eds) Automated Technology for Verification and Analysis - 20th International Symposium, ATVA 2022, Virtual Event, October 25-28, 2022, Proceedings, Lecture Notes in Computer Science, vol 13505. Springer, pp 208–224, (2022b). https://doi.org/10.1007/978-3-031-19992-9_13
16. Darwiche, A., Ji, C.: x 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. AAAI Press, pp 5582–5591, (2022). https://ojs.aaai.org/index.php/AAAI/article/view/20498
17. Fagin, R., Halpern, J.Y., Moses, Y., et al.: Reasoning about knowledge. MIT Press (1995). https://doi.org/10.7551/mitpress/5803.001.0001
18. Finkbeiner, B., Siber, J.: Counterfactuals modulo temporal logics. In: Piskac R, Voronkov A (eds) LPAR 2023: Proceedings of 24th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, Manizales, Colombia, 4-9th June 2023, EPiC Series in Computing, vol 94. EasyChair, pp 181–204, https://doi.org/10.29007/qtw7, (2023)
19. Finkbeiner, B., Zimmermann, M.: The first-order logic of hyperproperties. In: Vollmer H, Vallée B (eds) 34th Symposium on Theoretical Aspects of Computer Science, STACS 2017, March 8-11, 2017, Hannover, Germany, LIPIcs, vol 66. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp 30:1–30:14, (2017). https://doi.org/10.4230/LIPIcs.STACS.2017.30
20. Finkbeiner, B., Frenkel, H., Metzger, N., et al.: Synthesis of temporal causality. In: Gurfinkel A, Ganesh V (eds) Computer Aided Verification - 36th International Conference, CAV 2024, Montreal, QC, Canada, July 24-27, 2024, Proceedings, Part III, Lecture Notes in Computer Science, vol 14683. Springer, pp 87–111, (2024). https://doi.org/10.1007/978-3-031-65633-0_5
21. Finkbeiner, B., Frenkel, H., Siber, J.: An information-flow perspective on explainability requirements: Specification and verification. In: 22nd International Conference on Principles of Knowledge Representation and Reasoning, KR 2025, Melbourne, Australia, November 11-17, 2025, (to appear) (2025)
22. Goyal, Y., Wu, Z., Ernst, J., et al.: Counterfactual visual explanations. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, Proceedings of Machine Learning Research, vol 97. PMLR, pp 2376–2384, (2019). http://proceedings.mlr.press/v97/goyal19a.html
23. Halpern, J.Y.: Hypothetical knowledge and counterfactual reasoning. Int. J. Game Theory **28**(3), 315–330 (1999). https://doi.org/10.1007/s001820050113
24. Halpern, J.Y., O'Neill, K.R.: Secrecy in multiagent systems. ACM Trans Inf Syst Secur **12**(1), 5:1–5:47 (2008). https://doi.org/10.1145/1410234.1410239
25. Halpern JY, Pearl J (2005a) Causes and explanations: A structural-model approach. part i:Causes. The British Journal for the Philosophy of Science **56**(4):843–887. http://www.jstor.org/stable/3541870
26. Halpern, J.Y., Pearl, J.: Causes and explanations A structural-model approach. Explanations part ii: The British Journal for the Philosophy of Science **56**(4), 889–911 (2005). (http://www.jstor.org/stable/3541871)
27. Halpern, J.Y., van der Meyden, R., Vardi, M.Y.: Complete axiomatizations for reasoning about knowledge and time. SIAM J. Comput. **33**(3), 674–703 (2004). https://doi.org/10.1137/S0097539797320906

28. Hofmann J.: Logical methods for the hierarchy of hyperlogics. PhD thesis, Saarland University, Saarbrücken, Germany, (2022). https://publikationen.sulb.uni-saarland.de/handle/20.500.11880/35154
29. Horak, T., Coenen, N., Metzger, N., et al.: Visual analysis of hyperproperties for understanding model checking results. IEEE Trans Vis Comput Graph **28**(1), 357–367 (2022). https://doi.org/10.1109/TVCG.2021.3114866
30. Kamp, J.A.W.: Tense logic and the theory of linear order. University of California, Los Angeles (1968)
31. Khan, S.M., Lespérance, Y.: Knowing why - on the dynamics of knowledge about actual causes in the situation calculus. In: Dignum F, Lomuscio A, Endriss U, et al (eds) AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021. ACM, pp 701–709, (2021). https://doi.org/10.5555/3463952.3464037, https://www.ifaamas.org/Proceedings/aamas2021/pdfs/p701.pdf
32. Khan, S.M., Rostamigiv, M.: On explaining agent behaviour via root cause analysis: A formal account grounded in theory of mind. In: Gal K, Nowé A, Nalepa GJ, et al (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), Frontiers in Artificial Intelligence and Applications, vol 372. IOS Press, pp 1239–1247, (2023). https://doi.org/10.3233/FAIA230401
33. Köhl, M.A., Baum, K., Langer, M., et al.: Explainability as a non-functional requirement. In: Damian DE, Perini A, Lee S (eds) 27th IEEE International Requirements Engineering Conference, RE 2019, Jeju Island, Korea (South), September 23-27, 2019. IEEE, pp 363–368, (2019). https://doi.org/10.1109/RE.2019.00046
34. Ladner, R.E., Reif, J.H.: The logic of distributed protocols. In: Halpern JY (ed) Proceedings of the 1st Conference on Theoretical Aspects of Reasoning about Knowledge, Monterey, CA, USA, March 1986. Morgan Kaufmann, pp 207–222 (1986)
35. Langer, M., Oster, D., Speith, T., et al.: 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 (2021). https://doi.org/10.1016/j.artint.2021.103473
36. Lewis, D.K.: Counterfactuals. Blackwell, Cambridge, MA, USA (1973)
37. Lewis, D.K.: Causal explanation. In: Lewis D (ed) Philosophical Papers Vol. II. Oxford University Press, p 214–240 (1986)
38. Lichtenstein, O., Pnueli, A., Zuck, L.D.: The glory of the past. In: Parikh R (ed) Logics of Programs, Conference, Brooklyn College, New York, NY, USA, June 17-19, 1985, Proceedings, Lecture Notes in Computer Science, vol 193. Springer, pp 196–218, (1985). https://doi.org/10.1007/3-540-15648-8_16
39. Liu, X., Lorini, E.: A unified logical framework for explanations in classifier systems. J. Log. Comput. **33**(2), 485–515 (2023). https://doi.org/10.1093/logcom/exac102
40. van der Meyden, R., Shilov, N.V.: Model checking knowledge and time in systems with perfect recall (extended abstract). In: Rangan,C.P., Raman, V., Ramanujam, R. (eds) Foundations of Software Technology and Theoretical Computer Science, 19th Conference, Chennai, India, December 13-15, 1999, Proceedings, Lecture Notes in Computer Science, vol 1738. Springer, pp 432–445, (1999). https://doi.org/10.1007/3-540-46691-6_35
41. van der Meyden, R., Su. K.: Symbolic model checking the knowledge of the dining cryptographers. In: 17th IEEE Computer Security Foundations Workshop, (CSFW-17 2004), 28-30 June 2004, Pacific Grove, CA, USA. IEEE Computer Society, p 280, (2004). https://doi.org/10.1109/CSFW.2004.19, https://doi.ieeecomputersociety.org/10.1109/CSFW.2004.19
42. van der Meyden, R., Vardi, M.Y.: Synthesis from knowledge-based specifications (extended abstract). In: Sangiorgi D, de Simone R (eds) CONCUR '98: Concurrency Theory, 9th International Conference, Nice, France, September 8-11, 1998, Proceedings, Lecture Notes in Computer Science, vol 1466. Springer, pp 34–49, (1998). https://doi.org/10.1007/BFb0055614
43. Miller, T.: Explanation in artificial intelligence: insights from the social sciences. Artif. Intell. **267**, 1–38 (2019). https://doi.org/10.1016/j.artint.2018.07.007
44. Pnueli, A.: The temporal logic of programs. In: 18th Annual Symposium on Foundations of Computer Science, Providence, Rhode Island, USA, 31 October - 1 November 1977. IEEE Computer Society, pp 46–57, https://doi.org/10.1109/SFCS.1977.32
45. Poyiadzi, R., Sokol, K., Santos-Rodríguez, R., et al.: FACE: feasible and actionable counterfactual explanations. In: Markham AN, Powles J, Walsh T, et al (eds) AIES '20: AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, February 7-8, 2020. ACM, pp 344–350, (2020). https://doi.org/10.1145/3375627.3375850
46. Rabe, M.N.: A temporal logic approach to information-flow control. PhD thesis, Saarland University, (2016). http://scidok.sulb.uni-saarland.de/volltexte/2016/6387/
47. Rosenfeld, A., Richardson, A.: Explainability in human-agent systems. Auton Agents Multi Agent Syst **33**(6), 673–705 (2019). https://doi.org/10.1007/s10458-019-09408-y

48. Sakama, C.: Deception in epistemic causal logic. In: Sarkadi, S., Wright, B., Masters, P., et al. (eds.) Deceptive AI, pp. 105–123. Springer International Publishing, Cham (2021)
49. Sandu, A.S.: Knowledge of counterfactuals. PhD thesis, Cornell University (2021)
50. Schlicker, N., Langer, M., Ötting, S.K., et al.: What to expect from opening up 'black boxes'? comparing perceptions of justice between human and automated agents. Comput. Hum. Behav. **122**, 106837 (2021). https://doi.org/10.1016/j.chb.2021.106837
51. Stalnaker, R.: A Theory of Conditionals, Springer Netherlands, Dordrecht, pp 41–55, (1981). https://doi.org/10.1007/978-94-009-9117-0_2
52. Stalnaker, R.: On logics of knowledge and belief. Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition **128**(1), 169–199 (2006). (http://www.jstor.org/stable/4321718)
53. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: automated decisions and the gdpr. Harvard Journal of Law and Technology **31**(2), 841–887 (2018)
54. Ziemek, R., Piribauer, J., Funke, F., et al.: Probabilistic causes in markov chains. Innov. Syst. Softw. Eng. **18**(3), 347–367 (2022). https://doi.org/10.1007/s11334-022-00452-8

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.