Postulates for provenance: Instance-based provenance for first-order logic

Anonymous Author(s)

ABSTRACT

Instance-based provenance is an explanation for a query result in the form of a subinstance of the database. We investigate different desiderata one may want to impose on these subinstances. Concretely we consider seven basic postulates for provenance. Six of them relate subinstances to provenance polynomials, three-valued semantics, and Halpern-Pearl causality. Determinism of the provenance mechanism is the seventh basic postulate. Moreover, we consider the postulate of minimality, which can be imposed with respect to any set of basic postulates. Our main technical contribution is an analysis and characterisation of which combinations of postulates are jointly satisfiable. Our main conceptual contribution is an approach to instance-based provenance through three-valued instances, which makes it applicable to first-order logic queries involving negation.

CCS CONCEPTS

• **Theory of computation** → **Data provenance**.

KEYWORDS

First-order logic, query explanation, three-valued instance, provenance polynomial, causality, determinism, minimality

ACM Reference Format:
Anonymous Author(s). 2024. Postulates for provenance: Instance-based provenance for first-order logic. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

One of the main goals in the field of data provenance [8, 14] is to provide explanations for the results of database queries: Where do the resulting data come from? Why are they results? How were they produced? In this paper, we focus on the "why" and "where"; the "how" ties into much broader fields, such as process and workflow provenance [3, 26, 28], or self-explaining computation [7].

Two forms of explanations for query results can be distinguished, which we call *proof-based* and *instance-based* data provenance, respectively. Proof-based provenance presents a proof, or derivation, that a given result indeed satisfies a given query evaluated in a given database. A popular approach in this category is the use of provenance polynomials [16], which can be viewed as a compact representation of all proofs for unions of conjunctive queries. Provenance polynomials were later extended to full first-order logic by Grädel and Tannen [15, 32].

Instance-based provenance, on the other hand, explains query results by presenting a *subinstance* of the database. In the present paper, we investigate instance-based provenance from a broad perspective. Intuitively, the explaining subinstance should list the relevant information in the database that caused the query result under consideration. The question then becomes how to formalize this intuition. More broadly, what desiderata do we reasonably want to require of instance-based provenance? Also, are there requirements that are mutually incompatible?

Sufficiency, provenance relations, and 3-valued subinstances. The most evident requirement for instance-based provenance is that the subinstance should still satisfy the query it purports to explain. This requirement dates back to the lineage work [9] and was coined *sufficiency* by Glavic [14]. We take sufficiency as the defining property of a *provenance relation*: a total, generic mechanism that relates query results to sufficient subinstances. Totality means that every query result should get an explanation (possibly several, so nondeterminism is allowed), and genericity means that the mechanism should not interpret relation names.

In order to support queries involving negation, we need a proper notion of subinstance that treats the presence of tuples in relations on equal footing with the absence of tuples. For example, to explain that a tuple t belongs to the difference $R - S$ of two relations, the positive fact $R(t)$ is equally important as the negative fact $\neg S(t)$. We will thus define subinstances as *3-valued instances* that are consistent with the database instance under consideration. In our example, the 3-valued instance $I = \{R(t), \neg S(t)\}$ would then be a sufficient subinstance to explain that $t \in R - S$ in some database D. Here, I is 3-valued because it omits (and interprets as unknown) all other facts and non-facts from D, which are indeed irrelevant. Also, I is sufficient, because regardless of how we complete it to a total instance, $t \in R - S$ will be true. We thus adopt a certain-answer semantics, known as *supervaluation semantics*, for first-order logic on 3-valued instances [31].

Provenance polynomials and causality for first-order logic. While we use certain-answer semantics to define sufficiency, it is appropriate to take a more syntactic approach in the definition of provenance polynomials over 3-valued instances, as they are a proof-theoretic notion. We evaluate tokens corresponding to unknown facts as zero, and show that the polynomial for a formula φ in I is nonzero if and only if φ evaluates to true in I using *Kleene semantics*. It follows that the subinstance obtained from all tokens in the provenance polynomial constitutes a sufficient explanation. We thus generalize the connection that was known for UCQs to full first-order logic.

In order to define causes for first-order queries in 3-valued instances, we follow the most recent definition of Halpern-Pearl

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

causality [17]. We will characterize sufficient subinstances as those that intersect with all possible causes. It follows that the set of causal facts is another sufficient explanation, albeit quite different from the one obtained through the provenance polynomial.

Postulates for provenance. Towards our investigation of desiderata for instance-based provenance, we formulate seven *basic* postulates for provenance relations:

Proof containment (K): Let I be a subinstance returned as explanation for a first-order query φ on database D. Proof containment requires that the provenance polynomial for φ on I contain a monomial from the provenance polynomial for φ on D. Intuitively, this means that at least one of the proofs that φ holds in D still works to prove that φ holds in I. This postulate is indicated by K simply because it is equivalent to requiring that φ is Kleene-true in I (a stronger requirement than sufficiency).

Proof preservation (PP): Postulate PP is stronger than K and requires that the provenance polynomials for φ on I and on D be the same. Intuitively, I is fully representative for D in the sense that the different possible proofs for showing that φ holds are the same in I and in D.

Cause containment (CC), cause preservation (CP): Parallel to the previous two, these postulates deal with causes instead of provenance polynomials.

Proof relevance (PR), causal relevance (CR): These postulates upper-bound I by requiring that it should only contain facts from the provenance polynomial of φ on D (postulate PR), or facts from causes of φ on D (postulate CR).

Determinism (D): The seventh postulate is of a different nature and requires that the provenance relation be deterministic: for each query result a unique explanation can be given, without violating genericity.

On top of these, we consider postulates of *minimality*, $\textsc{Min}(X)$, for any subset X of $\{\mathrm{K}, \mathrm{PP}, \mathrm{CC}, \mathrm{CP}, \mathrm{PR}, \mathrm{CR}\}$. This postulate requires that the returned explanation be a minimal subinstance satisfying all properties in X.

While each basic postulate by itself is certainly reasonable, different postulates may be incompatible. For example, no provenance relation can be both K and CR, simply because a tautology (always-true query) has no causes. Also $\{\mathrm{PR}, \mathrm{CC}\}$ is unsatisfiable, since causes may require facts that do not show up in the provenance polynomial, as we will show. One more unsatisfiable example is $\{\textsc{Min}(\emptyset), \mathrm{D}\}$: there may be several minimal sufficient subinstances, and we cannot deterministically pick one in a generic manner.

On the other hand, $\{\mathrm{PR}, \mathrm{CR}, \mathrm{D}\}$ can be satisfied by returning the intersection of the set of facts from the provenance polynomial with the set of causal facts. Returning just the facts in the polynomial, or just the causal facts, satisfies $\{\mathrm{PP}, \mathrm{D}\}$ and $\{\mathrm{CC}, \mathrm{D}\}$, respectively. For another example, $\{\mathrm{CR}, \textsc{Min}(\mathrm{PR})\}$ can be satisfied by returning a minimally sufficient subset of facts from the tokens of the polynomial. Adding D to the latter set of postulates, however, renders it unsatisfiable again.

In this paper we will present a complete analysis of satisfiability for all possible combinations of postulates.

Positive formulas. Since much of provenance research has restricted attention to positive queries, or even just UCQs, it is natural to ask how the postulates behave in restriction to positive formulas. Note that we still allow universal quantifiers, something that seems to have been neglected in earlier work. The main effect of restricting to positive formulas, as we will show, is that causal facts necessarily must appear in the provenance polynomial (something that was already known for Meliou cases in the UCQ case). Consequently, some previously unsatisfiable combinations of postulates become satisfiable in this setting. The other combinations remain unsatisfiable, which is now more difficult to prove since we can only use positive formulas as counterexamples.

Related work. Besides provenance polynomials, graph-based proof representations for data provenance have been proposed as well [23, 24].

A version of Halpern-Pearl causality [18] was first applied to provenance by Meliou et al. [27]. They consider singleton subinstances to explain answers and nonanswers to conjunctive queries. Our definition conservatively extends the approach by Meliou et al., in the sense that, for positive formulas, a fact belongs to a cause if and only if it is a Meliou cause. Also, our result that causal facts for positive queries must appear in the provenance polynomial was already known for Meliou causes in the conjunctive-query case.

To our knowledge, our work is the first to consider instance-based provenance for full first-order logic (FO). For unions of conjunctive queries (UCQ, positive-existential formulas, or relational algebra without difference), well-known initial approaches to instance-based provenance are the notions of lineage [9] and witness [5]. Indeed, witnesses are basically defined to be sufficient subinstances [8]. Thus, our definition of provenance result is the generalization of witness to full FO. For UCQs, lineage has been shown to be sufficient [8]. Note that Cui and Widom have defined lineage in the presence of difference, but, then, lineage is no longer sufficient [10].

Still for UCQs, there are already known connections between proof-based and instance-based provenance [14]. In that case, the set of all tuples occurring as tokens in the provenance polynomial yields the lineage, and minimal monomials in the polynomial correspond similarly to minimal witnesses. Also, the tuples in these minimal monomials are the Meliou causes mentioned above.

Previous works have investigated postulates for other complex tasks, such as belief revision [19], clustering [20], or ensuring fairness [21]. Cheney [6] explores desiderata for provenance traces of program executions. Bourgaux et al. consider postulates for Datalog semantics over annotated databases [4]. While some of their postulates are specific to Datalog and/or annotation semantics, other ideas are relevant to our framework. For example, their "necessary" facts are counterfactual causes. Also, they define "usable" facts which are related to the tokens in the provenance polynomial.

2 PRELIMINARIES

We fix an infinite set **dom** called the domain and also assume an infinite supply of variables **var**. A *schema* Σ is a finite set of relation names, each with an associated arity. A *term* is either a variable or an element from **dom** (in which case the term is called a *constant*).

A Σ-*formula* φ is given by the following grammar:

$$\varphi ::= t_1 = t_2 \mid R(t_1, \dots, t_i) \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi \mid \forall x\, \varphi$$

where x is a variable from **var**, $t_1, t_2, \dots, t_i$ are terms, and R is a relation name from Σ of arity i. Atomic formulas of the form $R(t_1, \dots, t_i)$ are called *relation atoms*.

Remark 2.1. In many examples we will use propositional schemas, i.e., schemas consisting of proposition symbols, i.e., nullary relation names. Nullary atoms $P()$ will be simply written as P for clarity.

A *positive fact* over Σ is a statement of the form $R(a_1, \dots, a_i)$ where R is a relation name from Σ, of arity i, and $a_1, \dots, a_i$ are elements from **dom**. A *negative fact* over Σ is a statement of the form $\neg R(a_1, \dots, a_i)$ where $R(a_1, \dots, a_i)$ is a positive fact. We refer to both positive and negative facts simply as *facts*. The negation of a positive fact f is defined to be $\neg f$, and the negation of $\neg f$ is defined to be f. We define *flipping* a set of facts A, denoted by $\neg A$, as $\{\neg f \mid f \in A\}$. We also define the flipping of a subset of facts $D \subseteq A$ as $A[\neg D] = (A - D) \cup \neg D$. A set of facts is called *consistent* if it does not contain both a fact and its negation.

A Σ-*instance* is a finite consistent set of facts over Σ. We call B a *subinstance* of instance A simply if B is a subset of A.

Remark 2.2. What we call an *instance* is what is often called a "three-valued" instance. Standard, *total* instances will be formally defined shortly.

A *valuation* of a formula φ is a partial mapping ν from **var** to **dom**, defined at least on all free variables in φ. We also agree that every valuation is extended to **dom** as the identity: so $\nu(a) = a$ for every $a \in \mathbf{dom}$. We also write $\nu(\varphi)$ to denote the formula that substitutes the free variables of φ with the corresponding domain elements. The empty valuation (used when evaluating formulas without free variables) will be denoted in this paper by ε.

The *active domain* of an instance A, denoted *adom*(A), is the set of all domain elements that occur in A. The active domain of a formula φ, denoted *adom*(φ) is the set of domain elements that occur as constants in φ.

A *relativized instance* is an instance on an explicit domain [1]. Formally, a relativized instance of a schema Σ is a tuple $(\mathbf{d}, A)$ where A is a Σ-instance; $adom(A) \subseteq \mathbf{d} \subseteq \mathbf{dom}$; and $\mathbf{d}$ is finite. A Σ-formula φ is said to be *interpretable* in $(\mathbf{d}, A)$ if $adom(\varphi) \subseteq \mathbf{d}$.

A *completion* of $(\mathbf{d}, A)$ is a *total* relativized instance $(\mathbf{d}, B)$, on the same domain, such that $A \subseteq B$. Here, totality means that $(\mathbf{d}, B)$ is "two-valued", i.e., B contains either f or $\neg f$ for every fact f over Σ with constants from $\mathbf{d}$. Total relativized instances will henceforth be simply referred to as total instances. A formal convenience of our definition is that every total instance has a unique schema, determined by the relation names in its facts.

Let $(\mathbf{d}, A)$ be a total instance, let φ be a formula that is interpretable in $(\mathbf{d}, A)$, and let ν be a valuation of φ in $\mathbf{d}$. The notion that $(\mathbf{d}, A)$ satisfies φ under ν, denoted $(\mathbf{d}, A), \nu \models \varphi$, is well known and we omit the formal definition [1].

That is for total instances. For general, three-valued, instances, however, various semantics are in use [31]. In this paper, we will work with two of them. Supervaluation is a natural certain-answer semantics; Kleene semantics is the most classical of three-valued logics.

Let $(\mathbf{d}, A)$ be a relativized instance and let φ and ν be as above.

Supervaluation semantics. The supervaluation of φ in $(\mathbf{d}, A)$, denoted by $[\![\varphi]\!]^{(\mathbf{d},A),\nu}_{\text{super}}$, is defined to be $\mathbf{t}$ if $(\mathbf{d}, B), \nu \models \varphi$ for every completion $(\mathbf{d}, B)$ of $(\mathbf{d}, A)$; it is $\mathbf{f}$ if $(\mathbf{d}, B), \nu \not\models \varphi$ for every completion $(\mathbf{d}, B)$ of $(\mathbf{d}, A)$; and it is $\mathbf{u}$ otherwise. Here, $\mathbf{t}$, $\mathbf{f}$ and $\mathbf{u}$ stand for true, false and unknown.

Kleene semantics. The Kleene value, denoted by $[\![\varphi]\!]^{(\mathbf{d},A),\nu}_{\text{K}}$, is defined as follows. For relation atoms α, we define $[\![\alpha]\!]^{(\mathbf{d},A),\nu}_{\text{K}}$ to be $\mathbf{t}$ if $\nu(\alpha) \in A$; it is $\mathbf{f}$ if $\nu(\neg\alpha) \in A$; and it is $\mathbf{u}$ otherwise.

For equalities, $[\![t_1 = t_2]\!]^{(\mathbf{d},A),\nu}_{\text{K}}$ is $\mathbf{t}$ if $\nu(t_1)$ and $\nu(t_2)$ are the same element; otherwise it is $\mathbf{f}$.

The boolean operators follow the well-known 3-valued truth tables. Recall that $v_1 \vee v_2 = \mathbf{t}$ if at least one of v_1 and v_2 is $\mathbf{t}$; it is $\mathbf{f}$ if both are $\mathbf{f}$; and it is $\mathbf{u}$ otherwise. Likewise, $v_1 \wedge v_2 = \mathbf{f}$ if at least one of v_1 and v_2 is $\mathbf{f}$; it is $\mathbf{t}$ if both are $\mathbf{t}$; and it is $\mathbf{u}$ otherwise. Also, $\neg\mathbf{t} = \mathbf{f}$; $\neg\mathbf{f} = \mathbf{t}$; and $\neg\mathbf{u} = \mathbf{u}$.

Existential and universal quantifiers $\exists x\, \varphi_1$ and $\forall x\, \varphi_1$ are treated as disjunctions $\bigvee_{c \in \mathbf{d}} \varphi_1[x/c]$ and conjunctions $\bigwedge_{c \in \mathbf{d}} \varphi_1[x/c]$, respectively. Here $\varphi_1[x/c]$ denotes φ_1 where c is substituted for all free occurrences of x.

The advantage of Kleene semantics is that it is defined in a syntactical, compositional manner. The advantage of supervaluation is that it is more precise, in the sense that if a formula is Kleene-true or Kleene-false, then it also has that value under supervaluation.[1] On total instances, both semantics coincide with the standard one.

Example 2.3. As a simple example that supervaluation can be strictly more precise than Kleene, consider the propositional tautology $\varphi = P \vee \neg P$ with P nullary. It is always true on total instances, and also always true under supervaluation. Formally, on the empty relativized instance, we have $[\![\varphi]\!]^{(\emptyset,\emptyset),\varepsilon}_{\text{super}} = \mathbf{t}$. In contrast, $[\![\varphi]\!]^{(\emptyset,\emptyset),\varepsilon}_{\text{K}} = \mathbf{u}$ since $\mathbf{u} \vee \mathbf{u} = \mathbf{u}$.

Query results and potential query results. Let, as before, be $(\mathbf{d}, A)$ be a relativized instance of some schema Σ, let φ be a Σ-formula interpretable on $(\mathbf{d}, A)$, and let ν be a valuation of φ in $\mathbf{d}$. We refer to the tuple $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ as a *potential query result*; when indeed $[\![\varphi]\!]^{(\mathbf{d},A),\nu}_{\text{super}} = \mathbf{t}$, we call $\mathbf{r}$ plainly a *query result*. When $(\mathbf{d}, A)$ is total, we also call $\mathbf{r}$ a total query result.

3 PROVENANCE POLYNOMIALS AND KLEENE

We recall provenance polynomials for first-order logic [15], used here over the boolean semiring. We adapt them to three-valued instances, the simple idea being that $\mathbf{u}$ collapses to 0.

Let $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ be a potential query result, with φ in negation normal form. The provenance polynomial *pol*$(\mathbf{r})$ is defined in Figure 1. It is a polynomial over the boolean semiring, with facts (positive or negative) from A playing the role of indeterminates.[2] The indeterminates are referred to as *tokens*, and a negative fact used as token is written in the polynomial as $\overline{f}$ instead of $\neg f$.

[1] A similar property has been exploited in the context of instances with null values [25].

[2] The boolean semiring has two elements 0 and 1 with *logical or* as addition and *logical and* as multiplication. Note that, since $1 + 1 = 1$, any polynomial p is equal to $p + p$.

$$pol(\mathbf{d}, A, \nu, \alpha) = \begin{cases} \nu(\alpha) & \text{if } \nu(\alpha) \in A \\ 0 & \text{otherwise} \end{cases}$$

$$pol(\mathbf{d}, A, \nu, \neg\alpha) = \begin{cases} \overline{\nu(\alpha)} & \text{if } \nu(\neg\alpha) \in A \\ 0 & \text{otherwise} \end{cases}$$

$$pol(\mathbf{d}, A, \nu, t_1 = t_2) = \begin{cases} 1 & \text{if } \nu(t_1) = \nu(t_2) \\ 0 & \text{otherwise} \end{cases}$$

$$pol(\mathbf{d}, A, \nu, t_1 \neq t_2) = \begin{cases} 1 & \text{if } \nu(t_1) \neq \nu(t_2) \\ 0 & \text{otherwise} \end{cases}$$

$$pol(\mathbf{d}, A, \nu, \varphi_1 \vee \varphi_2) = pol(\mathbf{d}, A, \nu, \varphi_1) + pol(\mathbf{d}, A, \nu, \varphi_2)$$

$$pol(\mathbf{d}, A, \nu, \varphi_1 \wedge \varphi_2) = pol(\mathbf{d}, A, \nu, \varphi_1) \cdot pol(\mathbf{d}, A, \nu, \varphi_2)$$

$$pol(\mathbf{d}, A, \nu, \forall x\, \varphi_1) = \prod_{a \in \mathbf{d}} pol(\mathbf{d}, A, \nu[x \mapsto a], \varphi_1)$$

$$pol(\mathbf{d}, A, \nu, \exists x\, \varphi_1) = \sum_{a \in \mathbf{d}} pol(\mathbf{d}, A, \nu[x \mapsto a], \varphi_1)$$

Figure 1: Provenance polynomial of a potential query result for a formula in negation normal form. In the first two lines, α stands for a relation atom.

Example 3.1. Let $\mathbf{d} = \{a, b, c\}$, let φ be $\exists x(P(x) \wedge \neg Q(x))$, and consider the three instances

$$A = \{P(a), P(b), P(c)\}$$
$$B = \{P(a), P(b), P(c), \neg Q(b)\}$$
$$C = \{P(a), P(b), P(c), \neg Q(b), \neg Q(c)\}.$$

We have:

$$pol(\mathbf{d}, A, \varepsilon, \varphi) = 0$$
$$pol(\mathbf{d}, B, \varepsilon, \varphi) = P(b)\overline{Q(b)}$$
$$pol(\mathbf{d}, C, \varepsilon, \varphi) = P(b)\overline{Q(b)} + P(c)\overline{Q(c)}.$$ □

The polynomial of a potential query result tells us something about its Kleene-truth value. Specifically, we have the following generalization of Proposition 9 from Grädel and Tannen [15] to the three-valued setting. The proof is straightforward.

Proposition 3.2. *Let* $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ *be a potential query result. Then* $pol(\mathbf{r}) \neq 0$ *iff* $[\![\varphi]\!]_{\mathrm{K}}^{(\mathbf{d},A),\nu} = \mathbf{t}$.

Because supervaluation is more precise than Kleene, we have:

Corollary 3.3. *Let* $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ *be a potential query result. If* $pol(\mathbf{r}) \neq 0$ *then* $[\![\varphi]\!]_{\mathrm{super}}^{(\mathbf{d},A),\nu} = \mathbf{t}$.

The converse direction of the corollary does not hold for the same reason already illustrated in Example 2.3.

4 CAUSES IN FIRST-ORDER LOGIC

Causality is a large subject in the philosophy of science [29]. An influential proposal to define causality was made by Halpern and Pearl [18]. Their definition was updated a few times; here we follow the most recent definition [17].

Halpern and Pearl consider so-called *structural models* built on a set of *endogenous variables.* (There are also exogenous variables, which we do not consider here.) A structural model assigns to every such variable X a function F and a tuple $(X_1, \dots, X_m)$ of other variables to which F can be applied. The assignment to X, denoted by $X = F(X_1, \dots, X_m)$, is called a *structural equation.* The dependency graph on variables described by all the structural equations should be acyclic.

An *actual cause* for the values of certain variables is then defined to be a setting of values to some other variables, that satisfies a number of conditions [17] which we do not repeat here; below we will give a self-contained definition directly applied to our purpose.

Our purpose is, of course, the explanation of query results. We can straightforwardly view a query result $(\mathbf{d}, A, \nu, \varphi)$ as a structural model. As endogenous variables, we take all possible positive facts on $\mathbf{d}$, plus the pair (φ, ν), which we also view as an endogenous variable. The structural equations for positive facts f involve simple constant functions and take the forms $f = \mathbf{t}$ or $f = \mathbf{f}$ or $f = \mathbf{u}$, depending on whether f is in A, or $\neg f$ is in A, or neither is in A. The structural equation for (φ, ν) is given by the function that determines the value of $[\![\varphi]\!]_{\mathrm{super}}^{(\mathbf{d},A),\nu}$ from the values of the facts.

Under the above view, the notion of actual cause for the value $[\![\varphi]\!]_{\mathrm{super}}^{(\mathbf{d},A),\nu} = \mathbf{t}$ boils down to the following. The notation $A[\neg C]$ for flipping C in A was defined in Section 2.

Definition 4.1. A *supercause* of a query result $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ is a subinstance C of A such that $[\![\varphi]\!]_{\mathrm{super}}^{(\mathbf{d},A[\neg C]),\nu} \neq \mathbf{t}$. An *actual cause* (or simply *cause*) is a minimal supercause.

An earlier version of Halpern-Pearl causality was applied to conjunctive query results by Meliou et al. [27]. In their setting, actual causes are always single facts, which is not true here, as we will see in the following example. Nevertheless, we will see later that in the case of positive formulas, the Meliou causes are exactly the facts that appear in causes as defined here.

Example 4.2. Let $\mathbf{r} = (\emptyset, A, \varepsilon, \varphi\})$ with $A = \{P, Q, R\}$ and φ the propositional formula $(P \wedge Q) \vee (R \wedge S)$. There are two causes, namely, $\{P\}$ and $\{Q\}$. The query result $\mathbf{r}' = (\emptyset, B, \varepsilon, \varphi)$ with $B = A \cup \{S\}$ has four causes, namely, $\{P, R\}$, $\{P, S\}$, $\{Q, R\}$, and $\{Q, S\}$.

Our terminology of supercause and cause is inspired by similar terminology in dependency theory, where a key is a minimal superkey [1]. This analogy is not perfect, however. In dependency theory, every superset of a key is a superkey, but here, not every superset of a cause is a supercause. It is not even true that if C_1 and C_2 are supercauses and $C_1 \subseteq C_3 \subseteq C_2$, then C_3 must also be a supercause. For example, over proposition symbols P, Q and R, consider the formula φ that states that an odd number of said propositions is true. The supercauses of φ being true in $A = \{P, Q, R\}$ are all subsets of A of odd cardinality.

Remark 4.3. It is possible for a query result to have no causes at all. This can only happen when the formula says something purely about the domain. For example, let $\mathbf{d} = \{a, b, c\}$ and let φ be $\exists x_1 \exists x_2 \exists x_3 \forall y(y = x_1 \vee y = x_2 \vee y = x_3)$. This formula is supervaluation-true on *every* instance A with $adom(A) \subseteq \mathbf{d}$. Hence, for any such A, the query result $(\mathbf{d}, A, \varepsilon, \varphi)$ has no causes. In particular, this holds for formulas that are tautologies.

5 INSTANCE-BASED PROVENANCE AND SUFFICIENCY

Instance-based provenance attempts to explain a query result by providing a subinstance that is "sufficient". The query result may be assumed to be total, since that is the standard database setting. The subinstances serving as provenance, however, are typically not total. We are going to assess these three-valued instances with respect to the query they purport to explain. This is the reason why we needed to set up everything for 3-valued logic in the preceding sections.

Formally, let $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ be a total query result, and let B be a subinstance of A. We take sufficiency as the defining property of instance-based provenance:

Definition 5.1. Subinstance B is called *sufficient* for $\mathbf{r}$ if $(\mathbf{d}, B, \nu, \varphi)$ is also a query result, i.e., if $[\![\varphi]\!]^{(\mathbf{d},B),\nu}_{\text{super}} = \mathbf{t}$. A *provenance result* is a pair $(\mathbf{r}, B)$ such that B is sufficient for $\mathbf{r}$.

It is important to note that sufficiency is upward closed: if $B \subseteq A$ is sufficient and $B \subseteq B' \subseteq A$, then also B' is sufficient. This holds because 3-valued logic semantics (both Kleene and supervaluation) are monotone in information order (where $\mathbf{u} < \mathbf{t}$ and $\mathbf{u} < \mathbf{f}$ and $\mathbf{t}$ and $\mathbf{f}$ are incomparable). Intuitively, "giving more information does not hurt", although later in the paper we will pay attention to *minimality* as a desideratum for provenance.

We next explore how provenance results can be obtained from provenance polynomials and from causality.

5.1 Provenance from polynomials

In the case of unions of conjunctive queries, it is well known that the lineage or why-provenance of a query result is provided by the tokens in the provenance polynomial [16]. We generalize this connection here to full first-order logic.

Formally, let $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ be a total query result with provenance polynomial $p = pol(\mathbf{r})$. Since $\mathbf{r}$ is total, $[\![\varphi]\!]^{(\mathbf{d},A),\nu}_{\text{K}}$ equals $[\![\varphi]\!]^{(\mathbf{d},A),\nu}_{\text{super}} = \mathbf{t}$. Hence, by Proposition 3.2, the polynomial p is nonzero. For any monomial m of p, we write $tokens(m)$ to denote the set of facts (positive or negative) occurring as tokens in p. From the definition of p it is readily verified that $tokens(m)$ is a subinstance of A. We also write $tokens(\mathbf{r})$ for the union of all $tokens(m)$, i.e., the set of all facts occurring in p.

In a sense, every monomial of the provenance polynomial encodes a proof for the query result. In accordance, we establish:

Theorem 5.2. *Let $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ be a total query result, and let m be a monomial of $pol(\mathbf{r})$. Then $B = tokens(m)$ is sufficient for $\mathbf{r}$. Actually, B is even Kleene-sufficient, meaning that $[\![\varphi]\!]^{(\mathbf{d},B),\nu}_{\text{K}} = \mathbf{t}$.*

Proof sketch. For any $B \subseteq A$, the polynomial on B is a quotient of the polynomial on A; formally, $pol(\mathbf{d}, B, \nu, \varphi) = pol(\mathbf{r})/(A - B)$. Hence, if B contains $tokens(m)$, the polynomial on B still has m as a monomial. In particular, the polynomial on B is not zero. Proposition 3.2 then yields $[\![\varphi]\!]^{(\mathbf{d},B),\nu}_{\text{K}} = \mathbf{t}$ as desired. □

5.2 Provenance from causality

The following "hitting-set lemma" establishes a close connection between sufficiency and causality:

Lemma 5.3. *Let $\mathbf{r}$ be a total query result with instance A, and let $B \subseteq A$. Then B is sufficient for $\mathbf{r}$ if and only if B intersects every cause of $\mathbf{r}$.*

This is a good place to note that causes need not be sufficient, as illustrated by the propositional formula $P \wedge Q$ on instance $\{P, Q\}$ with causes $\{P\}$ and $\{Q\}$. The above lemma still implies a rather strong sufficiency result. Similar to $tokens(\mathbf{r})$, which contains all facts in the provenance polynomial, we define $cf(\mathbf{r})$ (causal facts) as the union of all causes of $\mathbf{r}$. We establish:

Theorem 5.4. *For any total query result $\mathbf{r}$, the intersection $cf(\mathbf{r}) \cap tokens(\mathbf{r})$ is sufficient for $\mathbf{r}$.*

Proof. From Theorem 5.2 and the upward-closedness of sufficiency, we know that $tokens(\mathbf{r})$ is sufficient, so by the above Lemma it intersects with all causes. Then certainly $cf(\mathbf{r}) \cap tokens(\mathbf{r})$ also intersects with all causes, so the same lemma yields sufficiency. □

The sets $cf(\mathbf{r})$ and $tokens(\mathbf{r})$ are in general incomparable. Intuitively this is because the notion of cause is *syntax-independent* [14]: it is the same for equivalent formulas. Provenance polynomials are syntax-dependent.

Example 5.5. Consider the propositional formula $\varphi = \psi \vee (\psi \wedge R)$ where $\psi = P \vee (\neg P \wedge Q)$. Note that ψ is equivalent to $P \vee Q$. Over instance $A = \{P, Q, R\}$, the provenance polynomial for φ is $P + PR$, so $tokens(\mathbf{r}) = \{P, R\}$. In contrast, the only cause of φ being true in A is $\{P, Q\}$.

6 PROPERTIES OF PROVENANCE RESULTS

We introduce a number of natural properties that one may want to require of provenance results. They will form the basis for the postulates in the next section.

Definition 6.1 (Properties of provenance results). Let $\mathbf{p} = (\mathbf{r}, B)$ be a provenance result, with $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$.

- $\mathbf{p}$ is *proof preserving* (pp) if $pol(\mathbf{d}, A, \nu, \varphi) = pol(\mathbf{d}, B, \nu, \varphi)$.
- $\mathbf{p}$ is *proof containing* (pc) if $tokens(m) \subseteq B$ for some monomial m in $pol(\mathbf{d}, A, \nu, \varphi)$.
- $\mathbf{p}$ is *proof-relevant* (pr) if $B \subseteq tokens(\mathbf{r})$.
- $\mathbf{p}$ satisfies *Kleene* (k) if $[\![\varphi]\!]^{(\mathbf{d},B),\nu}_{\text{K}} = \mathbf{t}$.
- $\mathbf{p}$ is *cause preserving* (cp) if $\mathbf{r}$ and $(\mathbf{d}, B, \nu, \varphi)$ have exactly the same causes.
- $\mathbf{p}$ is *cause containing* (cc) if B contains a cause for $\mathbf{r}$, on condition that a cause exists; otherwise, cc is trivially satisfied.
- $\mathbf{p}$ is *cause-relevant* (cr) if $B \subseteq cf(\mathbf{r})$.

We call the above *basic properties.* Let X be any set of basic properties.

- $\mathbf{p}$ is *minimal for X* ($min(X)$) if B is minimal such that $\mathbf{p}$ satisfies all basic properties in X.

We use the term 'proof' in the properties regarding the provenance polynomial because the monomials in the polynomial encode the different proofs for a query result. Apart from property Kleene (see below), there is a clear *symmetry* in the set of basic properties. On the one hand we have preservation, containment, and relevance for proofs; on the other hand, we have the same for causes. Here, the two relevance properties express *upper bounds* on the facts that appear in the provenance, while the preservation and containment properties express *lower bounds*.

One may wonder why proof containment is not defined in another way, requiring that the polynomials on A and on B have a monomial in common. Also, one may wonder why property Kleene is in the list. This is answered in the following:

Proposition 6.2. *Provenance result* $\mathbf{p}$ *as above is proof-containing, iff it satisfies Kleene, iff* $pol(\mathbf{d}, A, v, \varphi)$ *and* $pol(\mathbf{d}, B, v, \varphi)$ *have a monomial in common.*

We will continue with property k and omit the equivalent property pc. We can also give equivalent formulations for proof and cause preservation:

Proposition 6.3. *Let provenance result* $\mathbf{p} = (\mathbf{r}, B)$ *be as above. Then* $\mathbf{p}$ *is* pp *iff* B *contains* $tokens(\mathbf{r})$*, and* $\mathbf{p}$ *is* cp *iff* B *contains* $cf(\mathbf{r})$.

The properties we consider here are not all independent. For example, proof (cause) preservation is stronger than proof (cause) containment. Also, since proof and cause relevance are upper bound properties, it is plausible (we prove it formally) that $min(\text{pp})$ and $min(\text{k})$ imply pr, and similarly that $min(\text{cp})$ and $min(\text{cc})$ imply cr. Interestingly, also $min(\emptyset)$ implies cr. We list some useful implications in the following, where we denote logical implication by $\Rightarrow$. (We do not claim the list is complete.)

Proposition 6.4. *(1)* $\text{pp} \Rightarrow \text{k}$ *(2)* $\text{cp} \Rightarrow \text{cc}$ *(3)* $min(\text{pp}) \Rightarrow \text{pr}$ *(4)* $min(\text{k}) \Rightarrow \text{pr}$ *(5)* $min(\text{cp}) \Rightarrow \text{cr}$ *(6)* $min(\emptyset) \Rightarrow \text{cr}$ *(7)* $min(\text{cc}) \Rightarrow \text{cr}$ *(8)* $min(\text{pr}) \Rightarrow min(\emptyset)$ *(9)* $min(\text{cr}) \Leftrightarrow min(\emptyset)$

7 POSTULATES FOR PROVENANCE RELATIONS

Provenance relations are our proposed abstraction of mechanisms for instance-based provenance.

Definition 7.1. A *provenance relation* is an infinite set Π of provenance results that is *total* and *generic*.

By **totality** we mean that Π has at least one provenance result for each total query result (over all schemas). There may be several provenance results for one total query result, i.e., provenance relations may be nondeterministic.

By **genericity** we mean two things. First, Π should not interpret relation names, i.e., should not provide different provenance for situations that are identical except for the names of relations. Second, Π should not make a difference between a disjunction $\psi_1 \vee \psi_2$ and the same disjunction $\psi_2 \vee \psi_1$ in the reverse order, and similarly for conjunctions. When two formulas differ only in the way disjunctions and conjunctions are ordered, we call them isomorphic.

Formally, Π is generic if Π is invariant under *vocabulary renaming* and under *formula isomorphism*. We omit the formal definition of these invariances and give an example instead.

Example 7.2. Over proposition symbols P and Q, consider the formula $\varphi = P \vee Q$ and the total instance $\{P, Q\}$. Clearly, (A, φ) is a total query result; for simplicity in this example we omit the domain and the valuation from the notation for query and provenance results. Consider subinstances $B_1 = \{P\}$ and $B_2 = \{Q\}$. Both are sufficient for φ being true on A. Intuitively, there is no reason to prefer symbol P over symbol Q. Accordingly, if a provenance relation Π would relate (A, φ) to B_1, we would expect Π to relate (A, φ) also to B_2, and thus to be nondeterministic. We can see this formally using genericity. Assume $(A, \varphi, B_1) \in \Pi$. Formally, let ρ be the vocabulary renaming that swaps P and Q. By invariance under renaming, $(A, \varphi', B_2) \in \Pi$, where $\varphi' = Q \vee P$. Then by invariance under isomorphism, also $(A, \varphi, B_2) \in \Pi$.

Remark 7.3. Genericity is similar in spirit, but formally different, from the notion of genericity for database queries [1]. The latter notion is about invariance under isomorphism of instances, whereas our notion is about invariance on a simple syntactic level (schemas and formulas). □

The properties of provenance results from Definition 6.1 now give rise to properties on the level of provenance relations, simply by requiring them pointwise. We refer to these properties as *postulates* that one may want to impose on a provenance relation. We used lowercase letters for the properties; we now use uppercase letters for the postulates. In addition to postulates obtained from properties, we also consider the natural postulate of determinism.

The postulates on a provenance relation Π are as follows.

Polynomial preservation (PP): every $\mathbf{p} \in \Pi$ is proof preserving.
Kleene (K): every $\mathbf{p} \in \Pi$ satisfies Kleene.
Proof relevance (PR): every $\mathbf{p} \in \Pi$ is proof-relevant.
Cause Preservation (CP): every $\mathbf{p} \in \Pi$ is cause preserving.
Cause Containing (CC): every $\mathbf{p} \in \Pi$ is cause containing.
Causal Relevance (CR): every $\mathbf{p} \in \Pi$ is cause-relevant.
Determinism (D): for every total query result $\mathbf{r}$, there is exactly one provenance result $(\mathbf{r}, B)$ in Π.

Let X be any set of basic properties from Definition 6.1.

Minimal for X (Min(X)): every $\mathbf{p} \in \Pi$ is $min(X)$.

We can illustrate some postulates using five "canonical" provenance relations. All five are deterministic. Let $\mathbf{r}$ be an arbitrary total query result about instance A.

- Π^{id} relates $\mathbf{r}$ with A, as a subinstance of itself. Indeed, the entire instance is the trivial instance-based provenance. It satisfies the postulates PP, K, CP, CC, and D.
- Π^{tok} relates $\mathbf{r}$ with $tokens(\mathbf{r})$. It satisfies Min(pp), K, PR, and D.
- Π^{cf} relates $\mathbf{r}$ with $cf(\mathbf{r})$. It satisfies Min(cp), CC, CR, and D.
- $\Pi^{tokcf}_{\cap}$ relates $\mathbf{r}$ with $tokens(\mathbf{r}) \cap cf(\mathbf{r})$. It satisfies PR, CR, and D.
- $\Pi^{tokcf}_{\cup}$ relates $\mathbf{r}$ with $tokens(\mathbf{r}) \cup cf(\mathbf{r})$. It satisfies K, CC, D, and Min(pp, cp).

The above claims follow directly from the definitions, results and remarks in the preceding sections. For example, that Π^{cf} is a well-defined provenance relation, i.e., that $cf(\mathbf{r})$ is sufficient for any total query result $\mathbf{r}$, follows from Theorem 5.4 and the upward-closedness of sufficiency. That Π^{tok} satisfies K follows from Theorem 5.2 and the monotonicity of Kleene semantics in information order. That Π^{tok} satisfies Min(pp) is immediate from Proposition 6.3. And so on.

8 YOU CAN'T HAVE IT ALL: SATISFIABILITY

In this section we systematically analyze the *satisfiability* of different combinations of postulates. Here, a set of postulates $\mathcal{X}$ is called satisfiable if there exists a provenance relation that satisfies all postulates in $\mathcal{X}$.

We start by considering sets of *basic* postulates; these are the postulates PP, K, PR, CP, CC and CR corresponding to the corresponding basic properties pp, k, pr, cp, cc and cr of individual provenance results. For any set X of the latter properties, we denote the corresponding set of basic postulates by $\mathit{Postulates}(X)$. We also write $\overline{X}$ for the closure of X under the implications stated in Proposition 6.4.

It turns out that when a set of basic postulates is satisfiable, it is also satisfiable by a deterministic provenance relation. Recall the canonical provenance relations discussed at the end of the previous section. We show:

Theorem 8.1. *Let X be a set of basic properties. If $\overline{X}$ contains $\{\mathrm{k}, \mathrm{cr}\}$ or $\{\mathrm{pr}, \mathrm{cc}\}$, then $\mathit{Postulates}(X)$ is unsatisfiable. Otherwise, $\mathit{Postulates}(X) \cup \{D\}$ is satisfiable by one of the provenance relations Π^{id}, Π^{tok}, Π^{cf}, or $\Pi^{tokcf}_{\cap}$.*

Proof. To show $\{\mathrm{K}, \mathrm{CR}\}$ is not satisfiable consider any provenance relation Π and the query result $\mathbf{r}_1 = (\emptyset, \{P\}, \varepsilon, P \vee \neg P)$ over the proposition symbol P. Because of totality, there must be a provenance result $(\mathbf{r}_1, B) \in \Pi$. Since $P \vee \neg P$ is a tautology, $\mathbf{r}_1$ has no causes, so $\mathit{cf}(\mathbf{r}_1) = \emptyset$. Therefore, for Π to satisfy CR, the only possible value for B is $\emptyset$. However, $(\mathbf{r}_1, \emptyset)$ does not satisfy property k, so Π cannot satisfy K.

To show $\{\mathrm{PR}, \mathrm{CC}\}$ is not satisfiable consider $\mathbf{r}_2 = (\emptyset, \{P, Q\}, \varepsilon, \varphi)$ where φ is $P \vee (\neg P \wedge Q)$, for proposition symbols P and Q. Let $(\mathbf{r}_2, B) \in \Pi$. We can verify $\mathit{pol}(\mathbf{r}_2) = P$. Also, since φ is equivalent to $P \vee Q$, there is only one cause $\{P, Q\}$. For $\mathbf{r}$ to satisfy pr, we should have $B \subseteq \{P\}$, but then $(\mathbf{r}_2, B)$ cannot satisfy property cc, so Π cannot satisfy CC.

The remaining sets of postulates are satisfiable. We consider all maximal sets of basic postulates that are not supersets of $\{\mathrm{K}, \mathrm{CR}\}$ or $\{\mathrm{PR}, \mathrm{CC}\}$. The set $\{\mathrm{PP}, \mathrm{K}, \mathrm{CP}, \mathrm{CC}\}$ is satisfied by Π^{id}. The set $\{\mathrm{PP}, \mathrm{K}, \mathrm{PR}\}$ is satisfied by Π^{tok}. The set $\{\mathrm{CR}, \mathrm{CP}\}$ is satisfied by Π^{cf}. The set $\{\mathrm{PR}, \mathrm{CR}\}$ is satisfied by $\Pi^{tokcf}_{\cap}$. □

Note how the two unsatisfiable combinations show a symmetry between proofs and causes. Indeed, $\{\mathrm{K}, \mathrm{CR}\}$ requires proof containment (which we have seen is the same as property k) but causal relevance, while $\{\mathrm{CC}, \mathrm{PR}\}$ requires causal containment but proof relevance.

We next consider minimality. A single minimality postulate $\textsc{Min}(X)$ is satisfiable if and only if $\mathit{Postulates}(X)$ is. However, as already hinted by Example 7.2, determinism is no longer for free, so we obtain additional unsatisfiable combinations.

Example 8.2. Example 7.2 basically showed that $\{\textsc{Min}(\emptyset), \mathrm{D}\}$ is unsatisfiable. A different example is given by $\{\textsc{Min}(\mathrm{pp}, \mathrm{cc}), \mathrm{D}\}$. To see that this is unsatisfiable, consider $\mathbf{r} = (\varphi, A)$ with the propositional formula $\varphi = (P \wedge Q \wedge \neg R) \vee R$ and the instance $A = \{P, Q, R\}$ (we omit domain and valuation in propositional logic). The polynomial is R, and there are two causes, namely $B_1 = \{P, R\}$ and $B_2 = \{Q, R\}$. Crucially, B_1 and B_2 are also the only two minimal sufficient subinstances that are proof preserving and cause containing. Reasoning as in Example 7.2, swapping symbols P and Q, we see that any provenance relation Π containing $(\mathbf{r}, B_1)$ must also contain $(\mathbf{r}, B_2)$, and vice versa.

We can show the following.

Theorem 8.3. *For X a set of basic properties, $\{\textsc{Min}(X), D\}$ is satisfiable if and only if $\overline{X}$ equals the closure of one of $\{\mathrm{pp}\}$, $\{\mathrm{cp}\}$, $\{\mathrm{pp}, \mathrm{pr}\}$, $\{\mathrm{cp}, \mathrm{cr}\}$, or $\{\mathrm{pp}, \mathrm{cp}\}$.*

Note again the symmetry between proof and cause. Interestingly, the satisfiable combinations in the above result are "categorical", in the sense that each of them is satisfied by exactly one provenance relation. Indeed, $\{\textsc{Min}(\mathrm{pp}), \mathrm{D}\}$ and $\{\textsc{Min}(\mathrm{pp}, \mathrm{pr}), \mathrm{D}\}$ are equivalent and satisfied only by Π^{tok}; symmetrically, $\{\textsc{Min}(\mathrm{cp}), \mathrm{D}\}$ and $\{\textsc{Min}(\mathrm{cp}, \mathrm{cr}), \mathrm{D}\}$ are equivalent and satisfied only by Π^{cf}. Finally $\{\textsc{Min}(\mathrm{pp}, \mathrm{cp}), \mathrm{D}\}$ is satisfied only by $\Pi^{tokcf}_{\cup}$.

Next, we investigate combining a minimality postulate with extra basic postulates. For $\mathit{Postulates}(Y) \cup \{\textsc{Min}(X)\}$ to be satisfiable, at the very least $\mathit{Postulates}(X \cup Y)$ must be satisfiable, but, it turns out that many combinations become unsatisfiable.

Example 8.4. Consider $\{\textsc{Min}(\mathrm{k}), \mathrm{PP}\}$. The first postulate restricts the subinstance to come from just one monomial of the provenance polynomial; the second forces the subinstance to contain all provenance tokens. Intuitively, the two postulates are opposing each other and indeed their combination is unsatisfiable. (Consider, for example, a propositional formula $P \vee Q$ on the instance $\{P, Q\}$.)

For another example, consider $\{\textsc{Min}(\mathrm{pp}), \mathrm{CC}\}$, and consider the propositional formula $P \vee (\neg P \wedge Q)$ on the instance $\{P, Q\}$. The polynomial is P, but the only cause is $\{P, Q\}$, again showing that the two postulate are opposing each other.

We can characterize the combinations that remain satisfiable as follows.

Theorem 8.5. *For sets X and Y of basic properties, $\mathit{Postulates}(Y) \cup \{\textsc{Min}(X)\}$ is satisfiable if and only if*

(1) $Y \subseteq \overline{X}$ and $\mathit{Postulates}(X)$ is satisfiable; or
(2) X and Y fall in the following table:

X	Y	*Reason*
$\emptyset$	pr *or* cr *or* $\{\mathrm{pr}, \mathrm{cr}\}$	$\textsc{Min}(\mathrm{pr})$
pr	cr	$\textsc{Min}(\mathrm{pr})$
cr	pr	$\textsc{Min}(\mathrm{pr})$
k	pr	$\textsc{Min}(\mathrm{k})$
pp	pr	$\textsc{Min}(\mathrm{pp})$
cc	cr	$\textsc{Min}(\mathrm{cc})$
cp	cr	$\textsc{Min}(\mathrm{cp})$

We see that when $\mathit{Postulates}(Y) \cup \{\textsc{Min}(X)\}$ is satisfiable it is almost always equivalent to $\textsc{Min}(X)$, except in the first line in the table, where it is $\textsc{Min}(\mathrm{pr})$. This postulate can be satisfied by the provenance relation $\Pi^{tok}_{\min}$ that relates each total query result $\mathbf{r}$ to the minimal sufficient subinstances that are contained in $\mathit{tokens}(\mathbf{r})$. To satisfy postulate $\textsc{Min}(\mathrm{k})$ we can return the minimal monomials in $\mathit{pol}(\mathbf{r})$. Postulate $\textsc{Min}(\mathrm{cc})$ is satisfiable simply because CC is, but a simple description of a satisfying provenance relation seems elusive.

Of course, the discussed provenance relations are nondeterministic. So what happens when we combine a minimality postulate with both extra basic postulates and determinism? It turns out that essentially no satisfiable combinations remain, beyond those that are equivalent to a combination of a type already seen before. The same holds when we combine two different minimality postulates. Thus, our analysis of satisfiability is concluded.

Theorem 8.6. *Let X and Y be sets of basic properties.*

(1) *If $\{\text{Min}(X), D\} \cup \text{Postulates}(Y)$ is satisfiable, then it is equivalent to $\{\text{Min}(X), D\}$.*
(2) *If $\{\text{Min}(X), \text{Min}(Y)\}$ is satisfiable, then it is equivalent to $\text{Min}(X)$ or $\text{Min}(Y)$.*

9 POSITIVE QUERIES

A lot of past research on provenance has focused on positive relational algebra, or unions of conjunctive queries (UCQs). It is therefore interesting to look at the postulates in the absence of negation. We focus slightly more generally on *positive* first-order logic formulas, adapting the grammar from Section 2 as follows:

$$\varphi ::= t_1 = t_2 \mid t_1 \neq t_2 \mid R(t_1, \dots, t_i) \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi \mid \forall x\, \varphi$$

Thus, negation is absent, but universal quantification is still allowed.

In this setting, supervaluation semantics and Kleene semantics coincide. Also, it is no longer necessary to work with three-valued instances, since positive formulas are well-known to be monotone on total instances ($D \leq D'$ if every positive fact in D is also a positive fact in D'). For relativized three-valued instances we can now write $(\mathbf{d}, A), v \vDash \varphi$ to mean that $(\mathbf{d}, D), v \vDash \varphi$ with D the negative completion that adds unknown facts as negative.

Causes for positive formulas never contain negative facts. Moreover, our Definition 4.1 of cause is compatible with the definition by Meliou et al. [27] of causes for UCQs:

Definition 9.1. Let $\mathbf{r} = (\mathbf{d}, A, v, \varphi)$ be a total query result, with φ positive, and let f be a positive fact in A. We call f a *Meliou cause* for $\mathbf{r}$ if there exists a subinstance $B \subseteq A$ (called a contingency) such that $(\mathbf{d}, A - B), v \vDash \varphi$ but $(\mathbf{d}, A - B - \{f\}), v \nvDash \varphi$.

Proposition 9.2. *The set of Meliou causes for $\mathbf{r}$ equals $cf(\mathbf{r})$.*

Note that Meliou causes, being singletons, are not necessarily causes in themselves. For example, the only cause of $P \vee Q$ being true on instance $\{P, Q\}$ is $\{P, Q\}$ itself. This is because we use the modified Halpern-Pearl definition which is known for its better treatment of disjunction.

What makes the positive case simpler (compare Example 5.5) is that causal facts always appear in the provenance polynomial. The proof exploits that flipping positive facts simplifies to deleting them.

Proposition 9.3. *$cf(\mathbf{r}) \subseteq tokens(\mathbf{r})$ for positive query results $\mathbf{r}$.*

As a consequence, the closure $\overline{X}$ of a set X of basic properties is now done with respect to three implications: pp $\Rightarrow$ cp and cr $\Rightarrow$ pr by the above result, and cp $\Rightarrow$ cc as before. The implication pp $\Rightarrow$ k becomes moot in the positive case since k is always satisfied. The effect is that the combination of postulates {K, CR}, which was unsatisfiable (Theorem 8.1), simplifies to CR which is satisfied by Π^{cf}. Also {PR, CC} becomes satisfiable and is satisfied by Π^{tok}. Indeed, by Proposition 9.3, cc and pr are no longer conflicting properties.

Only one unsatisfiable combination remains in the positive case. Intuitively, properties pp and cr are still conflicting since the causal facts can be a strict subset of the provenance tokens.

Theorem 9.4. *For a set X of basic properties, Postulates(X) is unsatisfiable in the positive case iff $\overline{X}$ contains {pp, cr}.*

For minimality and determinism, compared to Theorem 8.3, there is one new satisfiable case: since cp and pr are no longer conflicting, $\{\text{Min}(\text{cp}, \text{pr}), \text{D}\}$ is now satisfied by Π^{cf}.

Theorem 9.5. *For X a set of basic properties, $\{\text{Min}(X), D\}$ is satisfiable in the positive case if and only if $\overline{X}$ equals the closure of one of {pp}, {cp}, {pp, pr}, {cp, cr}, {cp, pr}.*

We omit the treatment of combining minimality with extra basic postulates. We can again show that the only satisfiable cases are equivalent to a single minimality postulate. Also, Theorem 8.6 remains verbatim true in the positive case.

10 CONCLUSION

We have reported on a systematic investigation of instance-based provenance, in its relation to provenance polynomials and causality, in the setting of first-order queries with negation. We encountered a number of interesting provenance relations: the deterministic relations Π^{tok}, Π^{cf}, $\Pi^{tokcf}_{\cup}$ and $\Pi^{tokcf}_{\cap}$, and nondeterministic provenance such as the minimal monomials, the minimally sufficient subinstances, or the minimally sufficient subinstances that contain a cause. For example, for $P \vee (\neg P \wedge Q)$ on $\{P, Q\}$, subinstance $\{Q\}$ is minimally sufficient but does not appear in the polynomial.

Given the available variety of combinations of postulates, it would be interesting to conduct an empirical study on real-life queries, asking domain experts which provenance relations are the most practical and useful in different application scenarios. Moreover, such scenarios may suggest new postulates. Complexity requirements, or the "non-usable fact" postulate from Bourgaux et al. which requires that query results with the same polynomial should have the same provenance [4], are examples of other postulates.

Also, a novel application made possible by instance-based provenance is to return data in response to integrity constraints, i.e., boolean queries. This avenue is beginning to be explored in the context of RDF constraint languages [10, 22]

The complexity of Halpern-Pearl causality is already well studied [2, 12, 17]. Nevertheless, our particular instantiation of it for first-order logic query results may have different complexities, and it is a natural topic for further research to investigate data and combined complexity. (As done for Meliou causes [27].) Also the complexity of various properties of provenance results considered in this paper, such as cp or *min*(cc), merits further investigation.

Another natural direction for further research is to explore instance-based provenance, and the application of Halpern-Pearl causality, for classes of queries beyond first-order, e.g., queries involving aggregation or recursion. There exist proposals for Kleene semantics in such settings [13, 30]. Proof-based provenance results are already available through the work on provenance circuits for Datalog [11].

REFERENCES

[1] S. Abiteboul, R. Hull, and V. Vianu. 1995. *Foundations of Databases*. Addison-Wesley.

[2] G. Aleksandrowicz, H. Chockler, J.Y. Halpern, and A. Ivrii. 2014. The computational complexity of structure-based causality. In *Proceedings 28th AAAI Conference*, C.E. Brodley and P. Stone (Eds.). 974–980.

[3] Y. Amsterdamer, S.B. Davidson, D. Deutch, T. Milo, J. Stoyanovich, and V. Tannen. 2011. Putting lipstick on Pig: Enabling database-style workflow provenance. *Proceedings of the VLDB Endowment* 5, 4 (2011), 346–357.

[4] C. Bourgaux, P. Bourhis, L. Peterfreund, and M. Thomazo. 2022. Revisiting semiring provenance for Datalog. In *Proceedings 19th International Conference on Principles of Knowledge Representation and Reasoning*, G. Kern-Isberner, G. Lakemeyer, and T. Meyer (Eds.). IJACI Organization, 91–101.

[5] P. Buneman, S. Khanna, and W.C. Tan. 2001. Why and where: A characterization of data provenance. In *Database Theory—ICDT 2001 (Lecture Notes in Computer Science, Vol. 1973)*, J. Van den Bussche and V. Vianu (Eds.). Springer, 316–330.

[6] J. Cheney. 2010. Causality and the semantics of provenance. In *Proceedings 6th Workshop on Developments in Computational Models (EPTCS, Vol. 26)*, S.B. Cooper, E. Kashefi, and P. Panangaden (Eds.). Open Publishing Association, 63–74.

[7] J. Cheney, U.A. Acar, and R. Perera. 2013. Toward a theory of self-explaining computation. In *In Search of Elegance in the Theory and Practice of Computation*, V. Tannen, L. Wong, et al. (Eds.). Lecture Notes in Computer Science, Vol. 8000. Springer, 193–216.

[8] J. Cheney, L. Chiticariu, and W.-C. Tan. 2009. Provenance in Databases: why, how and where. *Foundations and Trends in Databases* 1, 4 (2009), 379–474.

[9] Y. Cui, J. Widom, and J.L. Wiener. 2000. Tracing the lineage of view data in a warehousing environment. *ACM Transactions on Database Systems* 25, 2 (2000), 179–227.

[10] T. Delva, A. Dimou, M. Jakubowski, and J. Van den Bussche. 2023. Data provenance for SHACL. In *Proceedings 26th International Conference on Extending Database Technology*, J. Stoyanovich, J. Teubner, et al. (Eds.). OpenProceedings.org, 285–297.

[11] D. Deutch, T. Milo, S. Roy, and V. Tannen. 2014. Circuits for Datalog provenance. In *Proceedings 17th International Conference on Database Theory*, N. Schweikardt et al. (Eds.). OpenProceedings.org, 201–212.

[12] T. Eiter and T. Lukasiewicz. 2002. Complexity results for structure-based causality. *Artificial Intelligence* 142, 1 (2002), 53–89.

[13] M. Fitting. 1985. A Kripke-Kleene semantics for logic programs. *Journal of Logic Programming* 2, 4 (1985), 295–312.

[14] B. Glavic. 2021. Data provenance: Origins, Applications, Algorithms, and Models. *Foundations and Trends in Databases* 9, 3–4 (2021), 209–441.

[15] E. Grädel and V. Tannen. 2017. Semiring provenance for first-order model checking. arXiv:1712.01980.

[16] T.J. Green, G. Karvounarakis, and V. Tannen. 2007. Provenance semirings. In *Proceedings 26th ACM Symposium on Principles of Database Systems*. ACM, 31–40.

[17] J.Y. Halpern. 2015. A modification of the Halpern-Pearl definition of causality. In *Proceedings 24th International Joint Conference on Artificial Intelligence*, Q. Yang and M.J. Wooldridge (Eds.). AAAI Press, 3022–3033.

[18] J.Y. Halpern and J. Pearl. 2005. Causes and explanations: A structural-model approach. Part I: Causes. *British Journal for the Philosophy of Science* 56 (2005), 843–887.

[19] H. Katsuno and A.O. Mendelzon. 1991. Propositional knowledge base revision and minimal change. *Artificial Intelligence* 52, 3 (1991).

[20] J.M. Kleinberg. 2003. An impossibility theorem for clustering. In *Proceedings 2002 Neural Information Processing Systems Conference*, S. Becker, S. Thrun, and K. Obermayer (Eds.). MIT Press, 446–453.

[21] J.M. Kleinberg, S. Mullainathan, and M. Raghavan. 2017. Inherent trade-offs in the fair determination of risk scores. In *8th Innovations in Theoretical Computer Science Conference (Leibniz International Proceedings in Informatics, Vol. 67)*, Ch.H. Papadimitriou (Ed.). Schloss Dagstuhl–Leibniz Center for Informatics, 43:1–43:23.

[22] J.E. Labra Gayo. 2021. Creating knowledge graph subsets using shape expressions. arXiv:2110.11709.

[23] S. Lee, B. Ludäscher, and B. Glavic. 2019. PUG: A framework and practical implementation for why and why-not provenance. *The VLDB Journal* 28 (2019), 47–71.

[24] C. Li, Z. Miao, Q. Zeng, B. Glavic, and S. Roy. 2021. Putting things into context: Rich explanations for query answers using join graphs. In *Proceedings 2021 International Conference on Management of Data*. ACM, 1051–1063.

[25] L. Libkin. 2016. SQL's three-valued logic and certain answers. *ACM Transactions on Database Systems* 41, 1 (2016), 1:1–1:28.

[26] B. Ludäscher, M. Weske, et al. 2009. Scientific workflows: Business as usual?. In *Proceedings 7th International Conference on Business Process Management (Lecture Notes in Computer Science, Vol. 5701)*, U. Dayal, J. Eder, J. Koehler, and H.A. Reijers (Eds.). Springer, 31–47.

[27] A. Meliou, W. Gatterbauer, K.M. Moore, and D. Suciu. 2010. The complexity of causality and responsibility for query answers and non-answers. *Proceedings of the VLDB Endowment* 4, 1 (2010), 34–45.

[28] L. Moreau. 2010. The foundations for provenance on the Web. *Foundations and Trends in Web Science* 2, 2–3 (2010), 99–241.

[29] J. Pearl. 2009. *Causality: Models, Reasoning, and Inference.* Cambridge University Press.

[30] N. Pelov, M. Denecker, and M. Bruynooghe. 2007. Well-founded and stable semantics of logic programs with aggregates. *Theory and Practice of Logic Programming* 7, 3 (2007), 301–353.

[31] G. Priest. 2008. *An Introduction to Non-Classical Logic: From If to Is.* Cambridge University Press.

[32] V. Tannen. 2017. Provenance analysis for FOL model checking. *ACM SIGLOG News* 4, 1 (2017), 24–36.

A PROOFS

Theorem 5.2

We fill in the details of the proof sketched in the main body of the paper.

Recall that the quotient of a polynomial p by a set Z of indeterminates, denoted by p/Z, is the polynomial obtained from p by setting the elements in Z to zero.

Lemma A.1. *Let* $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ *be a query result. Let* $B \subseteq A$. *We have* $pol(\mathbf{d}, B, \nu, \varphi) = pol(\mathbf{r})/(A - B)$.

Proof. By induction on the structure of φ in negation normal form. When φ is relation atom α, then $pol(\mathbf{r}) = \nu(\alpha)$. Clearly, if $\nu(\alpha) \in B$, we have $pol(\mathbf{d}, B, \nu, \varphi) = \nu(\alpha) = pol(\mathbf{r})$. Otherwise, if $\nu(\alpha) \notin B$, then $pol(\mathbf{d}, B, \nu, \varphi) = 0 = pol(\mathbf{r})/\{\nu(\alpha)\}$. The case when φ is a negated relation atom $\neg\alpha$, is analogous. When φ is $t_1 = t_2$ or $t_1 \neq t_2$, then $pol(\mathbf{r})$ and $pol(\mathbf{d}, B, \nu, \varphi)$ are the same and do not contain any tokens. When φ is $\varphi_1 \wedge \varphi_2$, we have the following:

$$\begin{aligned} pol(\mathbf{d}, B, \nu, \varphi_1 \wedge \varphi_2) &= pol(\mathbf{d}, B, \nu, \varphi_1) \cdot pol(\mathbf{d}, B, \nu, \varphi_2) \\ &= (pol(\mathbf{d}, A, \nu, \varphi_1)/B) \cdot (pol(\mathbf{d}, A, \nu, \varphi_2)/B) \\ &= (pol(\mathbf{d}, A, \nu, \varphi_1) \cdot pol(\mathbf{d}, A, \nu, \varphi_2))/B \\ &= pol(\mathbf{r})/B \end{aligned}$$

The case where φ is $\varphi_1 \vee \varphi_2$, is analogous to the previous case, where the definition of the polynomial uses addition instead of multiplication. Also the cases when φ is $\forall x\, \varphi_1$, or $\exists x\, \varphi_1$, are straightforward variants of the above calculations. □

Proof of the hitting-set lemma

Proof of Lemma 5.3. We will prove the only-if direction by contradiction. Suppose $[\![\varphi]\!]^{(\mathbf{d},B),\nu}_{\text{super}} \neq \mathbf{t}$. By definition, there exists a total instance $D \supseteq B$ such that $(\mathbf{d}, D), \nu \nvDash \varphi$. Since $D = A[\neg(A - D)]$, the set $C := A - D$ is a supercause for $\mathbf{r}$. Since $B \subseteq A \cap D$, it is disjoint from C. Since C contains a cause, B therefore does not intersect with all causes, which is a contradiction.

For the if-direction, towards a contradiction, suppose there exists a cause C for $\mathbf{r}$ such that $B \cap C = \emptyset$, i.e., $B \subseteq A - C$. Then $B \subseteq A[\neg C]$, whence $(\mathbf{d}, A[\neg C]), \nu \vDash \varphi$. This contradicts that C is a cause for $\mathbf{r}$. □

Proposition 6.2

Let us denote having a monomial in common by 'mp' (monomial preservation). Then mp $\Rightarrow$ pc is trivial, and pc $\Rightarrow$ k is given by Theorem 5.2. We next show k $\Rightarrow$ mp. From k and Proposition 3.2, $pol(\mathbf{d}, B, \nu, \varphi)$ has a monomial m. By Lemma A.1 this polynomial is $pol(\mathbf{r})/(B - A)$.

Proposition 6.3

We prove the two nontrivial implications of the proposition.

Lemma A.2. *Let* $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ *and* $\mathbf{r}' = (\mathbf{d}, B, \nu, \varphi)$ *be potential query results with* $tokens(\mathbf{r}) \subseteq B \subseteq A$. *Then the polynomials for* $\mathbf{r}$ *and* $\mathbf{r}'$ *are the same.*

Proof. By Lemma A.1, we know $pol(\mathbf{r}') = pol(\mathbf{r})/(A - B)$. However, because $tokens(\mathbf{r}) \subseteq B$, we know $A - B$ is disjoint from $tokens(\mathbf{r})$. Therefore $pol(\mathbf{r})/(A - B) = pol(\mathbf{r})$. □

Lemma A.3. *Let* $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ *and* $\mathbf{r}' = (\mathbf{d}, B, \nu, \varphi)$ *be potential query results with* $cf(\mathbf{r}) \subseteq B \subseteq A$. *Then* $\mathbf{r}$ *and* $\mathbf{r}'$ *have the same causes.*

Proof. For the only if-direction, we observe that $B[\neg C] \subseteq A[\neg C]$. So, because $[\![\varphi]\!]^{(\mathbf{d}, A[\neg C]), \nu}_{\text{super}} \neq \mathbf{t}$, by monotonicity of supervaluation also $[\![\varphi]\!]^{(\mathbf{d}, B[\neg C]), \nu}_{\text{super}} \neq \mathbf{t}$, whence C is a supercause for $\mathbf{r}'$. It remains to show that C is a minimal supercause for $\mathbf{r}'$. Let $C' \subsetneq C$. We must show that $[\![\varphi]\!]^{(\mathbf{d}, B[\neg C']), \nu}_{\text{super}} = \mathbf{t}$. First, we observe that $B - C'$ intersects with all causes for $\mathbf{r}$. Indeed, let E be a cause for $\mathbf{r}$. Since B contains $cf(\mathbf{r})$, we have $E \subseteq B$. Now suppose $B - C'$ would be disjoint from E. This can only happen if $E \subseteq C'$, whence $E \subsetneq C$, which is impossible since C is a cause for $\mathbf{r}$. So, $B - C'$ intersects with all causes for $\mathbf{r}$, and by Lemma 5.3, $[\![\varphi]\!]^{(\mathbf{d}, B - C'), \nu}_{\text{super}} = \mathbf{t}$. Hence, $[\![\varphi]\!]^{(\mathbf{d}, B[\neg C']), \nu}_{\text{super}} = \mathbf{t}$.

Next, for the if-direction, we are given that $[\![\varphi]\!]^{(\mathbf{d}, B[\neg C]), \nu}_{\text{super}} \neq \mathbf{t}$. Hence, by Lemma 5.3 we know $B[\neg C]$ is disjoint with at least one cause C' for $\mathbf{r}$. By definition, $B[\neg C] = (B - C) \cup \neg C$, so $B - C$ is also disjoint from C'. Furthermore, because $cf(\mathbf{r}) \subseteq B$, also $C' \subseteq B$. Therefore, C' must be a subset of C. We will show that $C = C'$ and thus that C is a cause for $\mathbf{r}$, as desired. Towards a contradiction, suppose $C' \subsetneq C$. Because C is minimal for $\mathbf{r}'$, $[\![\varphi]\!]^{(\mathbf{d}, B[\neg C']), \nu}_{\text{super}} = \mathbf{t}$. In particular $A[\neg C']$ is a completion for $B[\neg C']$, so $(\mathbf{d}, A[\neg C']), \nu \models \varphi$. This is a contradiction, as C' is a cause for $\mathbf{r}$. □

Proposition 6.4

Lemma A.4. *If* $\mathbf{p}$ *is* $min(\text{pp})$, *then* $B = tokens(\mathbf{r})$.

Proof. Assume $\mathbf{p}$ is minimal such that it satisfies pp. Then, by Lemma A.2, $tokens(\mathbf{r}) \subseteq B$. Moreover, $(\mathbf{r}, tokens(\mathbf{r}))$ is a provenance result by Lemma 5.2. Hence, by minimality, $B = tokens(\mathbf{r})$. □

Analogously, we also have such a lemma in terms of causality:

Lemma A.5. *If* $\mathbf{p}$ *is* $min(\text{cp})$, *then* $B = cf(\mathbf{r})$.

Proof. Assume $\mathbf{p}$ is minimal such that it satisfies cp. Then, by Lemma A.3, $cf(\mathbf{r}) \subseteq B$. Moreover, $(\mathbf{r}, cf(\mathbf{r}))$ is a provenance result by Theorem 5.4 and upwards closedness of sufficiency. Hence, by minimality, $B = cf(\mathbf{r})$. □

Proof of Proposition 6.4. We show every implication separately:

(1) We have pp $\Rightarrow$ pc immediately from the definitions, and pc $\Leftrightarrow$ k by Proposition 6.2.
(2) Directly from the definitions of the properties.
(3) Follows immediately from Lemma A.4.

Table 1: Counterexamples

Tag	φ	B_1	B_2	$pol(\mathbf{r})$	Causes for $\mathbf{r}$
(a)	$P \vee Q$	$\{P\}$	$\{Q\}$	$P + Q$	$\{P, Q\}$
(b)	$(P \wedge Q) \vee R$	$\{P, R\}$	$\{Q, R\}$	$PQ + R$	$\{P, R\}, \{Q, R\}$
(c)	$(P \wedge Q \wedge \neg R) \vee R$	$\{P, R\}$	$\{Q, R\}$	R	$\{P, R\}, \{Q, R\}$
(d)	$(P \vee \neg P) \vee (Q \vee \neg Q)$	$\{P\}$	$\{Q\}$	$P + Q$	no causes

(4) Assume $\mathbf{p}$ minimally satisfies k. By Proposition 6.2, $\mathbf{p}$ minimally satisfies pc. Let m be a monomial from $pol(\mathbf{r})$ such that $tokens(m) \subseteq B$. By Theorem 5.2, $(\mathbf{r}, tokens(m))$ is a provenance result. Hence, by minimality, $B = tokens(m)$, so $\mathbf{p}$ trivially satisfies pr.
(5) Follows immediately from Lemma A.5.
(6) Towards a contradiction, suppose there exists a fact $f \in B$ such that $f \notin cf(\mathbf{r})$. By Lemma 5.3, we know $(\mathbf{r}, B - \{f\})$ is still a provenance result because $B = \{f\}$ intersects with every cause for $\mathbf{r}$. This is in contradiction with the minimality of B.
(7) First, assume $\mathbf{r}$ has no causes. Then, by Remark 4.3 $(\mathbf{r}, \emptyset)$ is a provenance result, and voidlessly satisfies cc. Hence, if $\mathbf{p}$ satisfies $min(\text{cc})$, $B = \emptyset$ in this case, so $\mathbf{p}$ trivially satisfies cr.
Now assume $\mathbf{r}$ has a cause, and assume $\mathbf{p}$ minimally satisfies cc. Then, B contains such a cause C. Towards a contradiction, assume there exists $f \in B - cf(\mathbf{r})$. By Lemma 5.3, $B - \{f\}$ is still a provenance result, and still contains C since $C \subseteq cf(\mathbf{r})$. This contradicts the minimality of B.
(8) Immediate from the definition of pr.
(9) The if-direction is immediate from the definition of cr. The only-if direction follows from (6). □

Theorem 8.3

Clearly, for sets of properties X such that $Postulates(X)$ is not satisfiable, also $\textsc{Min}(X), \mathrm{D}$ is not satisfiable. For all other sets X, we need to investigate whether our claim holds. We start by enumerating these satisfiable sets. Figure 2 depicts this process. Table 2 lists the satisfiable sets $\overline{X}$. When $\textsc{Min}(\overline{X}), \mathrm{D}$ is satisfiable, we give a provenance relation that satisfies it. Otherwise, we give a counterexample.

It turns out that we can use just four counterexamples that have a similar structure. Generally, we can show $\textsc{Min}(\overline{X}), \mathrm{D}$ is not satisfiable by giving a query result $\mathbf{r} = (\mathbf{d}, A, \nu, \varphi)$ such that for every subinstance $B \subseteq A$ such that $(\mathbf{r}, B)$ satisfies $min(\overline{X})$ there exists a vocabulary renaming ρ such that $\rho(A) = A$, $\rho(\varphi) \cong \varphi$, but $\rho(B) \neq B$. Then, by genericity, both $(\mathbf{r}, B), (\mathbf{r}, \rho(B)) \in \Pi$ for any Π satisfying $\textsc{Min}(\overline{X})$. Thus, determinism is impossible.

In this proof, we will use particular counterexamples $\mathbf{r} = (\emptyset, \{P, Q, R, S\}, \varepsilon, \varphi)$ with φ a propositional formula. In our counterexamples, there will always be exactly two subinstances B such that $(\mathbf{r}, B)$ satisfies $min(\overline{X})$; we call them B_1 and B_2. Let ρ be the vocabulary renaming that swaps predicate names P and Q. We have, for every one of our counterexamples, $\rho(B_1) = B_2$. The counterexamples are listed in Table 1. The reader should verify, for every row in Table 2 that mentions a counterexample, that $(\mathbf{r}, B_1)$ and $(\mathbf{r}, B_2)$ indeed satisfy $min(\overline{X})$.

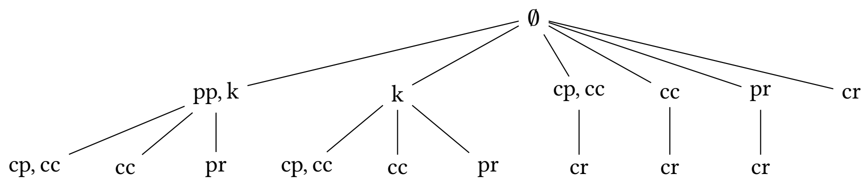

Figure 2: Starting with the empty set of properties, we add systematically add properties in the order: pp, k, cp, cc, pr, cr. Every time we add a property, we take the closure of the set. We leave out the sets X that are in violation of Theorem 8.1, i.e., the sets $Postulates(X)$ that are not satisfiable. We also leave out repeated sets. Every path in the tree from root to a node represents a satisfiable set (taking the union of the nodes on the path).

Table 2: Satisfiability of $\textsc{Min}(\overline{X})$, D. The Reason column (when not satisfiable) refers to Table 1.

$\overline{X}$	$\textsc{Min}(\overline{X})$, D	Reason
$\emptyset$	no	(a)
pp, k	yes	Π^{tok}
k	no	(a)
cp, cc	yes	Π^{cf}
cc	no	(b)
pr	no	(a)
cr	no	(a)
pp, k, cp, cc	yes	$\Pi^{tokcf}_{\cup}$
pp, k, cc	no	(c)
pp, k, pr	yes	Π^{tok}
k, cc	no	(b)
k, cp, cc	no	(d)
k, pr	no	(a)
cp, cc, cr	yes	Π^{cf}
cc, cr	no	(b)
pr, cr	no	(a)

Table 3: Satisfiability of $\textsc{Min}(\overline{X})$, $Postulates(\overline{Y})$.

$\overline{X}$	Y	sat?	Reason
$\emptyset$	k	no	(b)
	cc	no	(c)
	pr	yes	(a)
	cr	yes	$min(\emptyset) \Rightarrow min(\text{cr})$
	pr, cr	yes	(a)
pp, k	cc	no	(d)
	pr	yes	$min(\text{pp}) \Rightarrow \text{pr}$
k	pp	no	(c)
	cc	no	(c)
	pr	yes	$min(\text{k}) \Rightarrow \text{pr}$
cp, cc	k	no	(b)
	cr	yes	$min(\text{cp}) \Rightarrow \text{cr}$
cc	k	no	(b)
	cp	no	(e)
	cr	yes	$min(\text{cc}) \Rightarrow \text{cr}$
pr	k	no	(b)
	cr	yes	$min(\text{pr}) \Rightarrow min(\emptyset) \Rightarrow \text{cr}$
cr	cc	no	(c)
	pr	yes	(a)
pp, k, cc	cp	no	(f)
k, cc	pp	no	(e)
	cp	no	(e)
k, cp, cc	pp	no	(g)
k, pr	pp	no	(c)
cc, cr	cp	no	(e)

Theorem 8.5

Table 3 lists all combinations of $\overline{X}$ and Y of interest. The first column enumerates most sets $\overline{X}$ such that $Postulates(\overline{X})$ is satisfiable. This is the same listing as in the Proof of Theorem 8.3, obtained using the method from Figure 2, but omitting $Y \subseteq \overline{X}$. If there is an unsatisfiable $\overline{X}$–Y combination not present in the table, then it is not satisfiable because either:

- $Postulates(X, Y)$ is not satisfiable; or
- there exists a set of basic postulates Z such that $Z \subseteq \overline{Y}$, and $\textsc{Min}(\overline{X})$, $Postulates(\overline{Z})$ is not satisfiable. (This $\overline{X}$–Z combination is then present in the table.)

We now discuss the Reason column from Table 3. We start with the satisfiable sets of postulates:

(a) Let Π be the provenance relation that associates with any total query result $\mathbf{r}$ the minimal sufficient subinstances $B \subseteq tokens(\mathbf{r})$. By Theorem 5.4, such subinstances exist, and by Proposition 6.4 satisfy $min(\text{pr}, \text{cr})$. Hence, Π satisfies $\textsc{Min}(\text{pr}, \text{cr})$. Furthermore, $min(\text{pr}, \text{cr}) \implies min(\emptyset)$, as well as $min(\text{pr})$, $min(\text{cr})$, pr, and cr. Thus, our claims with Reason (a) in Table 3 follow.

Now, to show that some $\overline{X}$–Y combinations are not satisfiable, we will provide a query result $\mathbf{r}$ that serves as a counterexample in the following sense: for every subinstance B, if $\mathbf{p} = (\mathbf{r}, B)$ satisfies $min(\overline{X})$, then $\mathbf{p}$ does not satisfy Y. Below, for each reason, we cover all rows in Table 3 where that reason is invoked.

(b) Let $\mathbf{r} = (\emptyset, \{P\}, \varepsilon, P \vee \neg P)$. Only the provenance result $\mathbf{p} = (\mathbf{r}, \emptyset)$ satisfies $min(\emptyset)$. Furthermore $tokens(\mathbf{r}) = \{P\}$, so $\mathbf{p}$ also satisfies $min(\text{pr})$, and, as there are no causes, $\mathbf{p}$ also satisfies $min(\text{cp}, \text{cc})$ and $min(\text{cc})$. However, it is clear that $\mathbf{p}$ does not satisfy k.

(c) Let $\mathbf{r} = (\emptyset, \{P\}, \varepsilon, P \vee Q)$. There are two minimally sufficient subinstances: $\{P\}$ and $\{Q\}$. In other words $(\mathbf{r}, \{P\})$ and $(\mathbf{r}, \{Q\})$ are the only provenance results that satisfy $min(\overline{X})$ for $\overline{X} = \emptyset$. The same holds for the other relevant rows in Table 3:
   - $\overline{X} = \{\text{k}\}$. Clear.
   - $\overline{X} = \{\text{cr}\}$. The only cause is $\{P, Q\}$.
   - $\overline{X} = \{\text{k}, \text{pr}\}$. Indeed, $pol(\mathbf{r}) = P + Q$.

   The relevant rows in Table 3 are about $Y = \{\text{cc}\}$ or $Y = \{\text{pp}\}$. We can easily verify that neither $(\mathbf{r}, \{P\})$ nor $(\mathbf{r}, \{Q\})$ satisfy cc or pp.

(d) Let $\mathbf{r} = (\emptyset, \{P, Q\}, \varepsilon, P \vee (\neg P \wedge Q))$. Recall that pp $\Leftrightarrow$ pp, k. Because $pol(\mathbf{r}) = P$, the only provenance result that satisfies $min(\text{pp})$ is $\mathbf{p} = (\mathbf{r}, \{P\})$. Since there is only one cause, namely $\{P, Q\}$, we see that $\mathbf{p}$ does not satisfy cc, as desired.

(e) Let $\mathbf{r} = (\emptyset, \{P, Q, R\}, \varepsilon, (P \wedge Q) \vee R)$. There are two causes: $\{P, R\}$ and $\{Q, R\}$. These two causes are also sufficient subinstances. Let $\mathbf{p}$ be the provenance result $(\mathbf{r}, \{P, R\})$ or $(\mathbf{r}, \{Q, R\})$. Clearly $\mathbf{p}$ satisfies $min(\mathrm{cc})$. As $\mathbf{p}$ satisfies cr, it also satisfies $min(\mathrm{cc}, \mathrm{cr})$, and, as $\mathbf{p}$ satisfies k, it also satisfies $min(\mathrm{cc}, \mathrm{k})$.
Table 3 requires us to verify that $\mathbf{p}$ does neither satisfy cp nor pp. For cp this is because the subinstance does not contain $cf(A)$. For pp this is because $pol(\mathbf{r}) = PQ + R$.

(f) Let $\mathbf{r} = (\emptyset, \{P, Q, R\}, \varepsilon, (P \wedge Q \wedge \neg R) \vee R)$. There are two causes: $\{P, R\}$ and $\{Q, R\}$. These two causes are also sufficient subinstances. Let $\mathbf{p}$ be the provenance result $(\mathbf{r}, \{P, R\})$ or $(\mathbf{r}, \{Q, R\})$. Clearly $\mathbf{p}$ satisfies $min(\mathrm{cc})$, and because $pol(\mathbf{r})$ is R, $\mathbf{p}$ also satisfies $min(\mathrm{pp}, \mathrm{k}, \mathrm{cc})$.
As desired by Table 3, it is easy to see $\mathbf{p}$ does not satisfy cp (as the subinstance does not contain all necessary facts).

(g) Let $\mathbf{r} = (\emptyset, \{P, Q\}, \varepsilon, P \vee Q \vee \neg Q)$. As there are no causes, every provenance result $(\mathbf{r}, B)$ voidlessly satisfies cp and cc. There are two provenance results that satisfy $min(\mathrm{k})$: $(\mathbf{r}, \{P\})$ and $(\mathbf{r}, \{Q\})$. Because $pol(\mathbf{r}) = P + Q$, it is clear that neither of these provenance results satisfy pp.

Theorem 8.6

Proposition A.6. *Let X, Y be sets of basic properties. The set of postulates $\textsc{Min}(\overline{X}), Postulates(\overline{Y}), D$ is satisfiable when:*

(1) $Y \subseteq \overline{X}$ and $\textsc{Min}(X), D$ is satisfiable;
(2) $\overline{X} = \{\mathrm{pp}, \mathrm{k}\}$ and $Y = \{\mathrm{pr}\}$; or
(3) $\overline{X} = \{\mathrm{cp}, \mathrm{cc}\}$ and $Y = \{\mathrm{cr}\}$;

Otherwise, $\textsc{Min}(\overline{X}), Postulates(\overline{Y}), D$ is not satisfiable.

Proof. Condition (1) is clear. To show the rest of the claim, we list all X–Y pairs such that at least $\textsc{Min}(\overline{X}), Postulates(Y)$ is satisfiable in Table 4. Clearly, all other X–Y pairs (except for the ones that satisfy condition (1)) are not satisfiable. In Table 4, there are two possible situations for each row. First, the X–Y pair is equivalent to a postulate of the form $\textsc{Min}(\overline{X}), D$ (shown by using implications from Proposition 6.4), and thus we already knew whether it was satisfiable or not by Theorem 8.3. The second possibility is that it is unsatisfiable because of the following counterexample. We will argue that the case where $X = \{\emptyset\}$ and $Y = \{\mathrm{pr}, \mathrm{cr}\}$ is not satisfiable. By Totality, any provenance relation Π must have a provenance result for the total query result $\mathbf{r} = (\emptyset, \{P, Q\}, \varepsilon, \varphi)$ with φ the propositional formula $P \vee Q$. There are clearly two subinstances B such that $\mathbf{p} = (\mathbf{r}, B)$ is a provenance result: $\{P\}$ and $\{Q\}$. As $pol(\mathbf{r}) = P + Q$, $\mathbf{p}$ is pr, and as there is exactly one cause $C = \{P, Q\}$, $\mathbf{p}$ is also cr. If we consider the vocabulary renaming ρ that swaps P and Q, then by genericity both provenance results must be in Π. Therefore, Π cannot satisfy D. It easily verified that this counterexample also holds for the other two cases. ◻

Proposition A.7. *Let X, Y be sets of basic properties. The postulates of the form $\textsc{Min}(\overline{X}), \textsc{Min}(\overline{Y})$ is only satisfiable when X and Y are subsets of $\{\mathrm{pr}, \mathrm{cr}\}$, or when $\overline{X} = \overline{Y}$.*

Proof. We start by observing that if $\textsc{Min}(X), \textsc{Min}(Y)$ is satisfiable, then both $\textsc{Min}(X), Postulates(Y)$ and $\textsc{Min}(Y), Postulates(X)$ must be satisfiable.

Table 4: Satisfiability of $\textsc{Min}(\overline{X})$, $Postulates(\overline{Y})$, D.

$\overline{X}$	Y	sat?	Reason
$\emptyset$	pr	no	counterexample
	cr	no	$min(\emptyset) \Rightarrow$ cr and $\textsc{Min}(\emptyset)$, D is not satisfiable
	pr, cr	no	counterexample
pp, k	pr	yes	$min(\mathrm{pp}) \Rightarrow$ pr
k	pr	no	$min(\mathrm{k}) \Rightarrow$ pr and $\textsc{Min}(\mathrm{k})$, D is not satisfiable
cp, cc	cr	yes	$min(\mathrm{cp}) \Rightarrow$ cr
cc	cr	no	$min(\mathrm{cc}) \Rightarrow$ cr and $\textsc{Min}(\mathrm{cc})$, D is not satisfiable
pr	cr	no	$min(\mathrm{pr}) \Rightarrow min(\mathrm{cr})$ and $\textsc{Min}(\mathrm{pr})$, D is not satisfiable
cr	pr	no	counterexample

Table 5: The pairs of sets of properties X, Y such that both $\textsc{Min}(X), Postulates(Y)$ and $\textsc{Min}(Y), Postulates(X)$ are satisfiable, and the reason why also $\textsc{Min}(X), \textsc{Min}(Y)$ is satisfiable.

X	Y	Reason
$\emptyset$	pr	$min(\mathrm{pr}) \Rightarrow min(\emptyset)$; equivalent to $min(\mathrm{pr})$
	cr	$min(\mathrm{cr}) \Leftrightarrow min(\emptyset)$; equivalent to $min(\emptyset)$
	pr, cr	$min(\mathrm{pr}, \mathrm{cr}) \Rightarrow min(\mathrm{cr}) \Rightarrow min(\emptyset)$; equivalent to $min(\mathrm{pr}, \mathrm{cr})$
pr	cr	$min(\mathrm{pr}) \Rightarrow min(\mathrm{cr})$; equivalent to $min(\mathrm{pr})$

The relevant X–Y pairs can be deduced from Theorem 8.5 and the accompanying Table 3. The pairs where this holds (ignoring the symmetric cases) are: $X = \emptyset$ and $Y \subseteq \{\mathrm{pr}, \mathrm{cr}\}$; and $X = \{\mathrm{pr}\}$ and $Y = \{\mathrm{cr}\}$. It turns out that for each of these pairs, $\textsc{Min}(X), \textsc{Min}(Y)$ is satisfiable because they are equivalent to a single satisfiable minimality postulate. The reasoning can be found in Table 5, we do not list the symmetric cases (where the X and Y values are swapped) or the cases where $X = Y$ as these are trivial. The satisfiable cases are satisfied by provenance relations described in the Proof of Theorem 8.5, Reason (a). ◻

Proposition 9.2

For the if-direction, suppose f is a Meliou cause, with contingency B. W.l.o.g. we may assume that B is a minimal contingency. Clearly, $B \cup \{f\}$ is a supercause. We still need to show that $B \cup \{f\}$ is a minimal supercause. Take any $C \subsetneq B \cup \{f\}$. We consider two possibilities. First, if $f \in C$, then we know $A - C \vDash_{\mathbf{d}} \varphi$ by the minimality of B. Second, if $f \notin C$, then $C \subseteq B$ and because $A - B \vDash_{\mathbf{d}} \varphi$, by monotonicity of positive formulas, $A - C \vDash_{\mathbf{d}} \varphi$.

For the only-if direction, suppose C is a cause for $\mathbf{r}$ and $f \in C$. We show that f is a Meliou cause with contingency $B := C - \{f\}$. Clearly, $A - B \vDash_{\mathbf{d}} \varphi$ as C is a minimal supercause. Furthermore, $A - B - \{f\} \nvDash_{\mathbf{d}} \varphi$ as $A - B - \{f\} = A - C$ and C is a supercause.

Proposition 9.3

Let C be a cause for $\mathbf{r}$. We show $C \subseteq tokens(\mathbf{r})$ by verifying that $C \cap tokens(\mathbf{r})$ is a supercause. Let $\mathbf{r} = (\mathbf{d}, A, v, \varphi)$, let $\mathbf{r}' = (\mathbf{d}, A - (C \cap tokens(\mathbf{r})), v, \varphi)$, and let $\mathbf{r}'' = (\mathbf{d}, A - C, v, \varphi)$. Then

$$\begin{aligned} pol(\mathbf{r}') &= pol(\mathbf{r})/(C \cap tokens(\mathbf{r})) \\ &= pol(\mathbf{r})/C \\ &= pol(\mathbf{r})/A - (A - C) \end{aligned}$$

Table 6: Satisfiability of the postulates $\textsc{Min}(\overline{X})$, D for the positive-existential setting. The Reason column (when not satisfiable) refers to Table 1.

$\overline{X}$	$\textsc{Min}(\overline{X})$, D	Reason
$\emptyset$	no	(a)
pp, cp, cc	yes	Π^{tok}
cp, cc	yes	Π^{cf}
cc	no	(b)
cr, pr	no	(a)
pr	no	(a)
pp, cp, cc, pr	yes	Π^{tok}
cp, cc, cr, pr	yes	Π^{cf}
cp, cc, pr	yes	Π^{cf}
cc, cr, pr	no	(b)
cc, pr	no	(b)

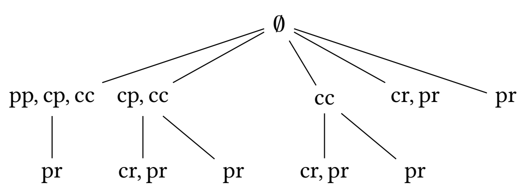

Figure 3: Similar to Figure 2 but now including the implications resulting from Proposition 9.3 and considering the unsatisfiability from Proposition 9.4

$$= pol(\mathbf{r}'')$$
$$= 0,$$

where the first equality holds by Lemma A.1, the second since dividing by non-tokens has no effect, the third since $C \subseteq A$, the fourth again by Lemma A.1, and the last since C is a cause. Thus $pol(\mathbf{r}') = 0$, so by Proposition 3.2, $(\mathbf{d}, A - (C \cap tokens(\mathbf{r}))), \nu \not\models \varphi$, i.e., $C \cap tokens(\mathbf{r})$ is a supercause as desired.

Theorem 9.4

To show {PP, CR} is not satisfiable consider any provenance relation Π and the query result $\mathbf{r} = (\emptyset, \{P, Q\}, \varepsilon, \varphi)$ with φ the formula $P \vee (P \wedge Q)$. Because of Totality, we know there must be a provenance result $(\mathbf{r}_1, B) \in \Pi$. There is one cause $C := \{P\}$ for $\mathbf{r}$, so $cf(\mathbf{r}) = \{P\}$. However, the polynomial for $\mathbf{r}$ is $pol(\mathbf{r}) = P + PQ$. For Π to satisfy PP, B must contain Q, but for Π to satisfy CR, B cannot contain Q.

The remaining sets of postulates are satisfiable. We consider the two maximal sets of postulates that are not supersets of {PP, CR}. Excluding CR, we get the set {PP, CP, CC, PR}, which is satisfied by Π^{tok}. Excluding PP, we get the set {CP, CC, CR, PR}, which is satisfied by Π^{cf}.

Theorem 9.5

Clearly, for sets of properties $\overline{X}$ such that $Postulates(\overline{X})$ is not satisfiable, also $\textsc{Min}(\overline{X})$, D is not satisfiable. For all other sets $\overline{X}$, we need to investigate whether our claim holds. Similar to the proof of Theorem 8.3, we start by enumerating these satisfiable sets. Figure 3 depicts this process. Table 6 lists the satisfiable sets $\overline{X}$. When $\textsc{Min}(\overline{X})$, D is satisfiable, we give a provenance relation that satisfies it. Otherwise, we give a counterexample.

The counterexamples are the same as in the proof of Theorem 8.3. Here, we only refer to counterexamples that do not use negation.