Semiring Provenance for First-Order Model Checking

Erich Grädel
RWTH Aachen University

Val Tannen
Univ. of Pennsylvania

December 7, 2017

Abstract

Given a first-order sentence, a model-checking computation tests whether the sentence holds true in a given finite structure. Data provenance extracts from this computation an abstraction of the manner in which its result depends on the data items that describe the model. Previous work on provenance was, to a large extent, restricted to the negation-free fragment of first-order logic and showed how provenance abstractions can be usefully described as elements of commutative semirings — most generally as multivariate polynomials with positive integer coefficients.

In this paper we introduce a novel approach to dealing with negation and a corresponding commutative semiring of polynomials with dual indeterminates. These polynomials are used to perform reverse provenance analysis, i.e., finding models that satisfy various properties under given provenance tracking assumptions.

1 Introduction

Semiring provenance was originally developed for positive database query languages [16]. From this baseline, we have recently started to investigate an approach to the provenance analysis of model checking for full first-order logic (FOL). We propose a novel approach to dealing with negation in provenance formulation and a corresponding commutative semiring of polynomials with dual indeterminates. A preliminary account of this joint work was given by the second author in [24].

Data provenance is extremely useful in many computational disciplines. Suppose that a computational process is applied to a complex input consisting of multiple items. Provenance analysis allows us to understand how these different input items affect the output of the process. It can be used to answer questions of the following type:

(1) Which ones of input items are actually used in the computation of the output?

(2) Can the same output be obtained from different combinations of input items?

(3) In how many different ways can the same output be computed?

As a consequence, provenance can be further applied to issues such as deciding how much to trust the output, assuming that we may trust some input items more than others, deciding what clearance level is required for accessing the output, assuming that we know the clearance levels for the input items, or, assuming that one has to pay for the input items, how to minimize the cost of obtaining the output. More generally, *reverse* provenance analysis allows us to find input data (here first-order models) that satisfies various properties

under given provenance tracking assumptions. This is also closely related to reverse data management [20, 21].

It turns out that the questions listed above, as well as several other questions of interest, can be answered for database transformations (queries and views) via interpretations in commutative semirings. In past work, the semiring provenance approach has been applied to query and view languages such as the positive relational algebra [16, 13], nested relations/complex values (objects) [10, 23], Datalog [16, 7], XQuery (for unordered XML) [10] full relational algebra (on $\mathbb{Z}$-annotated relations) [14], SQL aggregates [3], workflows with map-reduce modules [1], and languages for data-centric (data-dependent) processes [8]. Moreover, the semiring approach has been successfully implemented in two software systems, Orchestra [15, 17, 18] and Propolis [8].

There exists a well-known tight connection between conjunctive queries in databases and constraint satisfaction problems in AI [19]. In this light, and despite a number of technical differences, there exists an interesting connection (that needs more exploration) between the semiring provenance framework applied to conjunctive queries and the semiring framework for *soft constraint satisfaction* [6, 5].

The reader may have noticed that the bulk of the work on provenance for database transformations was concerned with *positive* query languages. Indeed, trying to add to the commutative semiring structure operations that capture difference of relations has led to interesting and algebraically challenging, but divergent approaches [11, 14, 3, 2, 12]. In particular there is no separate account of tracking *negative* information, an aspect that we hope to remedy here.

1.1 Provenance Semantics

We shall consider certain non-standard semantics for FOL that will help us to understand how a sentence φ ends up being true in a finite structure $\mathfrak{A}$, i.e., whether $\mathfrak{A} \models \varphi$ holds or not (we call this *provenance in model checking*). The non-standard semantics that we champion involves various *commutative semirings*. Here we strive to justify this choice.

First of all, the standard semantics for first-order logic maps formulae to truth values in $\mathbb{B} = \{\bot, \top\}$, which form a commutative semiring with respect to the operations of disjunction and conjunction, with units $\bot$ and $\top$.

Second, in a provenance semantics we want to understand the connections between the facts (positive or negative) that are embodied in a model $\mathfrak{A}$ and their use in a justification that $\mathfrak{A} \models \varphi$. Since the model is finite, we can think of such a justification as an alternating disjunction-conjunction *proof tree* (an example appears in 3.2). In any case, these justifications are definitely not proofs in some axiomatization of FOL. If we had a provenance semantics for model checking, it would, in particular, help us to count proof trees. This particular case suffices to suggest the semiring structure as well as some ways in which such non-standard semantics can be quite different from the standard one.

Notice that a semiring semantics refines the classical Boolean semantics, and formulae that are classically equivalent may become non-equivalent under a semantics that counts proof trees. Indeed, already a sentence $\varphi \vee \varphi$ has in general more proof trees than φ. We further illustrate with the failure of some of the usual logical equivalences invoked in transforming sentences to *prenex form*.

Let $\rho \equiv (\forall x\, \varphi) \wedge \psi$ and $\sigma \equiv \forall x\, (\varphi \wedge \psi)$. Every proof tree of ρ can be transformed into a proof tree of σ by making copies of the subtree rooted at ψ. However, when ψ has two or more distinct proof trees we see that σ can have strictly more proof trees than ρ. Similarly we can argue that $\forall x\, (\varphi \vee \psi)$ can have strictly more proof trees than $(\forall x\, \varphi) \vee \psi$.

Now consider $\rho \equiv (\exists x\, \varphi) \vee \psi$ and $\sigma \equiv \exists x\, (\varphi \vee \psi)$. Let's write $\varphi(x)$ to show occurrences of x in φ. For simplicity suppose that the model has exactly two elements, a and b, and that each of $\varphi(a)$, $\varphi(b)$, and ψ has exactly one proof tree. Then, ρ will have 3 proof trees but σ will have 4.

Finally, we note that $(\exists x\, \varphi) \wedge \psi)$ and $\exists x\, (\varphi \wedge \psi)$ have exactly the same number of proof trees and this reflects the fact that multiplication distributes over addition.

For other sentences, we can see that the number-of-proof-trees constitutes a non-standard semantics for FOL sentences constructed using disjunction, conjunction, existentials and universals, because, moreover, addition and multiplication are associative and commutative.

This discussion provides some partial justification for considering commutative semirings as semantic domains. The rest of the justification will follow from the subsequent development.

Remark. Instead of thinking about proof trees for $\mathfrak{A} \models \varphi$, we could equivalently consider *winning strategies* in $\mathcal{G}(\mathfrak{A}, \varphi)$, the model checking game for $\mathfrak{A}$ and φ (see e.g. [4]). We do not pursue this aspect in this paper, but we remark that a provenance analysis in commutative semirings can also be developed for more general models of finite and infinite games, beyond the acyclic and always terminating first-order model-checking games. Also beyond the applications to query evaluation and logic, a provenance analysis of games provides insights into more subtle game-theoretic questions than just who wins the game, concerning for instance the number or costs of winning strategies, or issues like confidence and trust in game-theoretic settings. This approach will be developed in more detail in a forthcoming paper.

1.2 Intermezzo: Examples of Commutative Semirings

Definition 1 *An algebraic structure $(K, +, \cdot, 0, 1)$, with $0 \neq 1$, is a* semiring *when $(K, +, 0)$ is a commutative monoid, $(K, \cdot, 1)$ is a monoid, $\cdot$ distributes over $+$ and $0 \cdot a = a \cdot 0 = 0$. The semiring is* commutative *when $\cdot$ is commutative, and it is* idempotent *when $+$ is idempotent.*

Any distributive lattice is an idempotent commutative semiring. Here are some commutative semirings of interest to us:

1. The Boolean semiring $\mathbb{B} = (\mathbb{B}, \vee, \wedge, \bot, \top)$ is the standard habitat of logical truth. It is a distributive lattice.
2. $\mathbb{N} = (\mathbb{N}, +, \cdot, 0, 1)$ is used for *bag semantics* in databases and we use it here for counting proof trees.
3. $\mathbb{T} = (\mathbb{R}^{\infty}_{+}, \min, +, \infty, 0)$ is called the *tropical* semiring and is idempotent but not a distributive lattice. Its elements and operations appear in *min-cost* interpretations (e.g., shortest paths) and it plays a surprising role in connecting certain dynamic programming algorithms in statistics with certain methods of algebraic geometry [22] (see also next item).
4. $\mathbb{V} = ([0, 1], \max, \cdot, 0, 1)$ is called the *Viterbi* semiring and is isomorhic to $\mathbb{T}$ via $x \mapsto e^{-x}$ and $y \mapsto -\ln y$. When interpreted as probabilities, its elements and operations appear in *statistical model* interpretations (e.g., maximum probability trajectories in Hidden Markov Models). We will think of the elements of $\mathbb{V}$ as *confidence scores*.
5. $\mathbb{F} = ([0, 1], \max, \min, 0, 1)$, is called the *fuzzy* semiring. It is a distributive lattice.
6. $\mathbb{A} = (\{\mathsf{P} < \mathsf{C} < \mathsf{S} < \mathsf{T} < 0\}, \min, \max, 0, \mathsf{P})$ is the *access control* semiring, where P is "public", C is "confidential", S is "secret", T is "top secret", and 0 is "so secret that nobody can access it!". This is a distributive lattice (beware! the lattice order is the opposite of the one we used in the definition).

7. For any set X, the semiring $\mathbb{N}[X] = (\mathbb{N}[X], +, \cdot, 0, 1)$ consist of the multivariate polynomials in indeterminates from X and with coefficients from $\mathbb{N}$. This is the commutative semiring freely generated by the set X. It's used for a general form of provenance.

8. $\mathsf{PosBool}(X) = (\mathsf{PosBool}(X), \vee, \wedge, \bot, \top)$ is the semiring whose elements are classes of equivalent positive (monotone) boolean expressions with boolean variables from X (its elements are in bijection with the positive boolean expressions in irredundant disjunctive normal form). This is the distributive lattice freely generated by the set X. It is also used for provenance, e.g., in probabilistic databases.

2 First-Order Logic Interpreted in Commutative Semirings

We are interested in the provenance analysis of the model checking computation of first-order sentences. Such a computation is nicely and *declaratively* driven by the structure of the sentence, and thus amounts to a non-standard semantics for FOL. In its simplest form model checking takes as input a finite structure and the input items are the various facts (positive or negative) which hold in the model. We have found however that it pays to take a more general approach and specify not a structure but just its (finite) universe. This way we can track the use of positive and negative facts in checking a sentence under multiple possible models on that universe. This allows a certain amount of *reverse analysis*: finding models that satisfy useful constraints.

2.1 K-Interpretations

Consider a finite relational vocabulary: $\mathscr{V} = \{R, S, \ldots\}$. From this vocabulary and a finite *non-empty* universe A of *ground values* we construct the set Facts_A of all ground relational atoms (facts) $R(\mathbf{a})$, the set $\mathsf{NegFacts}_A$ of all negated facts $\neg R(\mathbf{a})$ and thus the set $\mathsf{Lit}_A = \mathsf{Facts}_A \cup \mathsf{NegFacts}_A$ of all *literals*, positive and negative facts, over $\mathscr{V}$ and A. By convention we will identify $\neg\neg R(\mathbf{a}) \equiv R(\mathbf{a})$) so the negation of a literal is again a literal.

Any finite structure $\mathfrak{A} = (A, R^{\mathfrak{A}}, S^{\mathfrak{A}}, \ldots)$ with universe A makes some of these literals true and the remaining ones false. Note, however, that much of the development does not assume a specific model, and this can be usefully exploited.

Let $(K, +, \cdot, 0, 1)$ be a commutative semiring. Very roughly speaking, $0 \in K$ is intended to interpret false assertions, while an element $a \neq 0$ in K provides a "nuanced" interpretation for true assertions (call them "a-true").

Next, K-interpretations will map literals to elements of K and are then extended to all formulae. Disjunction and existential quantification are interpreted by the addition operation of K. Conjunction and universal quantification are interpreted by the multiplication operation of K. For quantifiers, the finiteness of the universe A of ground values will be essential. For negation we use the well-known syntactic transformation to *negation normal form (NNF)*, denoted $\psi \mapsto \mathsf{nnf}(\psi)$. Note that $\mathsf{nnf}(\psi)$ is a formula constructed from literals (positive and negative facts) and equality/inequality atoms using just $\wedge, \vee, \exists, \forall$.

Definition 2 *A K-**interpretation** is a mapping $\pi : \mathsf{Lit}_A \to K$. This is extended to FO formulae given*

valuations $\nu : \mathsf{Vars} \to A$*:*

$$\begin{array}{rclcrcl}
\pi[\![R(\mathbf{x})]\!]_\nu &=& \pi(R(\nu(\mathbf{x})) & & \pi[\![\neg R(\mathbf{x})]\!]_\nu &=& \pi(\neg R(\nu(\mathbf{x})) \\
\pi[\![x \mathsf{\ op\ } y]\!]_\nu &=& \text{if } \nu(x) \mathsf{\ op\ } \nu(y) \text{ then } 1 \text{ else } 0 & & \pi[\![\varphi \wedge \psi]\!]_\nu &=& \pi[\![\varphi]\!]_\nu \cdot \pi[\![\psi]\!]_\nu \\
\pi[\![\varphi \vee \psi]\!]_\nu &=& \pi[\![\varphi]\!]_\nu + \pi[\![\psi]\!]_\nu & & \pi[\![\exists x\, \varphi]\!]_\nu &=& \sum_{a \in A} \pi[\![\varphi]\!]_{\nu[x \mapsto a]} \\
\pi[\![\forall x\, \varphi]\!]_\nu &=& \prod_{a \in A} \pi[\![\varphi]\!]_{\nu[x \mapsto a]} & & \pi[\![\neg\varphi]\!]_\nu &=& \pi[\![\mathsf{nnf}(\neg\varphi)]\!]_\nu
\end{array}$$

The symbol $\mathsf{\ op\ }$ stands for either $=$ or $\neq$. As you can see from the definition, the equality and inequality atoms are interpreted in K as 0 or 1, i.e., their provenance is not tracked. One could give a similar treatment to other such relations with "fixed" meaning, e.g., assuming an ordering on A, however, we omit this here.

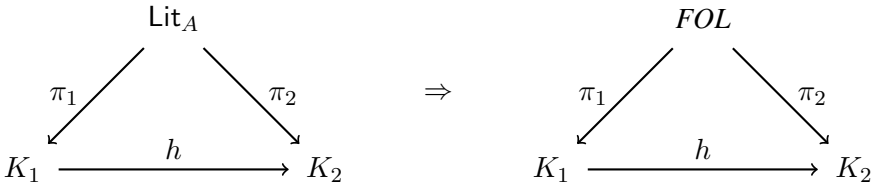
As intended, it suffices to consider formulae in NNF:

Proposition 3 $\pi[\![\varphi]\!]_\nu = \pi[\![\mathsf{nnf}(\varphi)]\!]_\nu$

Corollary 4
$$\begin{array}{rcl}
\pi[\![\neg(\varphi \wedge \psi)]\!]_\nu &=& \pi[\![\neg\varphi \vee \neg\psi)]\!]_\nu \\
\pi[\![\neg(\varphi \vee \psi)]\!]_\nu &=& \pi[\![\neg\varphi \wedge \neg\psi)]\!]_\nu \\
\pi[\![\neg(\forall x\, \varphi)]\!]_\nu &=& \pi[\![\exists x\, \neg\varphi]\!]_\nu \\
\pi[\![\neg(\exists x\, \varphi)]\!]_\nu &=& \pi[\![\forall x\, \neg\varphi]\!]_\nu \\
\pi[\![\neg\neg\varphi]\!]_\nu &=& \pi[\![\varphi]\!]_\nu
\end{array}$$

A useful consequence of Proposition 3 is that we can prove further results by induction on formulas in NNF, and hence avoid the negation connective. When φ is a sentence we write just $\pi[\![\varphi]\!]$.

Proposition 5 (Fundamental Property) *Let* $h : K_1 \to K_2$ *be a semiring homomorphism and let* $\pi_1 : \mathsf{Lit}_A \to K_1$ *and* $\pi_2 : \mathsf{Lit}_A \to K_2$ *be interpretations such that* $h \circ \pi_1 = \pi_2$*. Then, for any FOL sentence* φ *we have* $h(\pi_1[\![\varphi]\!]) = \pi_2[\![\varphi]\!]$*. As diagrams*



Proof: By Proposition 3 the proof can proceed by induction on formulae in NNF. For example $h(\pi_1[\![\varphi \wedge \psi]\!]_\nu) = h(\pi_1[\![\varphi]\!]_\nu \cdot_1 \pi_1[\![\psi]\!]_\nu) = h(\pi_1[\![\varphi]\!]_\nu) \cdot_2 h(\pi_1[\![\psi]\!]_\nu) = \pi_2[\![\varphi]\!]_\nu \cdot_2 \pi_2[\![\psi]\!]_\nu = \pi_2[\![\varphi \wedge \psi]\!]_\nu$. ■

The somewhat bombastic name "fundamental property" is motivated by two observations. First, the property checks that the definition of our semantics is nicely compositional. Second, the property plays a central role in a strategy that we have widely applied with query languages in databases: compute provenance as generally as (computationally) feasible, then specialize via homomorphisms to coarser-grain provenance, or to specific domains, e.g., count, trust, cost or access control.

2.2 Intermezzo: Positive Semirings

We say that a semiring K has *divisors of 0* if there exist $a, b \in K$ such that $a \neq 0$, $b \neq 0$ but $ab = 0$. None of the semirings described in Sect. 1.2 has divisors of 0. The classical examples of such are rings that are not integral domains, e.g., $\mathbb{Z}_6$, as well as boolean algebras.

A semiring K is *+-positive* if $a + b = 0$ implies $a = 0$ and $b = 0$. Rings, e.g., $\mathbb{Z}$, or the boolean ring $\mathbb{Z}_2$, are not $+$-positive. Finally, a semiring is (simply) *positive* [9] if it is $+$-positive and has no divisors of 0. All the semirings described in Sect. 1.2 are positive.

Proposition 6 *A semiring K is positive if, and only if, $\dagger_K : K \to \mathbb{B}$ defined by*

$$\dagger_K(a) = \begin{cases} \top & \text{if } a \neq 0 \\ \bot & \text{if } a = 0 \end{cases} \qquad \textit{is a homomorphism.}$$

2.3 Sanity Checks

Let $\mathfrak{A} = (A, R^{\mathfrak{A}}, S^{\mathfrak{A}}, \ldots)$ be a (finite) $\mathscr{V}$-model.

The **canonical truth interpretation** for $\mathfrak{A}$ is, of course, $\pi_{\mathfrak{A}} : \mathsf{Lit}_A \to \mathbb{B}$ where

$$\pi_{\mathfrak{A}}(L) \;=\; \begin{cases} \top & \text{if } \mathfrak{A} \models L \\ \bot & \text{otherwise} \end{cases}$$

Earlier we have discussed "number of proof trees" as a non-standard semantics for FOL model-checking. This is also captured by interpretations in a semiring.

The **canonical counting interpretation** for $\mathfrak{A}$ is $\pi_{\#\mathfrak{A}} : \mathsf{Lit}_A \to \mathbb{N}$ where

$$\pi_{\#\mathfrak{A}}(L) \;=\; \begin{cases} 1 & \text{if } \mathfrak{A} \models L \\ 0 & \text{otherwise} \end{cases}$$

Proposition 7 (sanity checks) *For any FOL sentence φ we have $\mathfrak{A} \models \varphi$ if, and only if, $\pi_{\mathfrak{A}}[\![\varphi]\!] = \top$. Moreover, $\pi_{\#\mathfrak{A}}[\![\varphi]\!]$ is the number of proof trees that witness $\mathfrak{A} \models \varphi$.*

Now, let K be a commutative semiring, and let π be a K-interpretation. As we have indicated, for a sentence φ we intend to interpret $\pi[\![\varphi]\!] = 0$ as "φ is false in K", while $\pi[\![\varphi]\!] = k \neq 0$ is interpreted as "φ is k-true in K", i.e., as offering "shades of truth". We examine how this meshes with standard logical truth in a model.

Definition 8 *A K-interpretation $\pi : \mathsf{Lit}_A \to K$ is **model-defining** when, for each fact, one of $\pi(R(\mathbf{a}))$ and $\pi(\neg R(\mathbf{a}))$ is 0 and the other one is $\neq 0$.*

Indeed, every model-defining interpretation π uniquely defines a $\mathscr{V}$-model $\mathfrak{A}_\pi$ with universe A such that for any literal L we have $\mathfrak{A}_\pi \models L$ if, and only if, $\pi(L) \neq 0$.

Both $\pi_{\mathfrak{A}}$ and $\pi_{\#\mathfrak{A}}$ shown above are model-defining and the model they define is $\mathfrak{A}$. If K is not $\mathbb{B}$ then several model-defining interpretations may define the same model. It is also clear that any finite model can be defined by such an interpretation, for any K.

Proposition 9 (another sanity check) *Let K be* positive*, and let π be a model-defining K-interpretation. Then for any FOL sentence*

$$\mathfrak{A}_\pi \models \varphi \;\;\Leftrightarrow\;\; \pi[\![\varphi]\!] \neq 0$$

Proof: By Proposition 6, since K is positive, $\dagger_K$ is a homomorphism. Since π is model-defining let $\mathfrak{A}$ be the model defined by π. Clearly, $\dagger_K \circ \pi$ is the canonical truth interpretation $\pi_{\mathfrak{A}}$. Applying Proposition 5 we get $\dagger_K(\pi[\![\varphi]\!]) = \pi_{\mathfrak{A}}[\![\varphi]\!]$. Now the result follows from Proposition 7. ■

In fact, we can refine the previous proposition as follows.

Proposition 10 (refinement of Proposition 9)

(a) *For any semiring K (positive or not!), for any model-defining K-interpretation π, and for any FOL sentence φ we have*

$$\pi[\![\varphi]\!] \neq 0 \;\;\Rightarrow\;\; \mathfrak{A}_\pi \models \varphi.$$

(b) *Moreover, a semiring K is positive if, and only if, for any model-defining K-interpretation π and any FOL sentence φ we have*

$$\mathfrak{A}_\pi \models \varphi \;\;\Rightarrow\;\; \pi[\![\varphi]\!] \neq 0.$$

Proof: Part (a) of the proposition is by induction on φ.

The left to right implication in part (b) follows from Proposition 9. For the right to left implication we first prove that K has no divisors of 0. Suppose that $a, b \in K$ are such that $a \neq 0$, $b \neq 0$ but $ab = 0$. Consider $A = \{\{c_1, c_2\}$ and the model-defining interpretation defined by $\pi(\neg R(c_1)) = \pi(\neg R(c_2)) = 0$, $\pi(R(c_1)) = a, \pi(R(c_2)) = b$ as well as the sentence $\varphi = R(c_1) \wedge R(c_2)$. We have $\mathfrak{A}_\pi \models \varphi$ hence $\pi[\![\varphi]\!] \neq 0$, contradiction.

Next we prove that K is $+$-positive. Let $a, b \in K$ be such that $a \neq 0$ and $b \neq 0$. Consider the same interpretation π as above, with the sentence $\psi = R(c_1) \vee R(c_2)$. We have $\mathfrak{A}_\pi \models \psi$ hence $0 \neq \pi[\![\psi]\!] = a + b$. ■

2.4 "Consistency" and "completeness" for K-interpretations

In the study of provenance we shall also have occasion to consider interpretations that do not correspond to a single specific model (as formalized in Definition 8). Additional issues arise for such interpretations.

An interpretation in which both $\pi[\![\varphi]\!] \neq 0$ and $\pi[\![\neg\varphi]\!] \neq 0$ for some sentence φ is seemingly "inconsistent". On the other hand, an interpretation in which both $\pi[\![\varphi]\!] = 0$ and $\pi[\![\neg\varphi]\!] = 0$ for some sentence φ seems to to be "incomplete". [1] Of course, neither of these situations arises for a model-defining K-interpretation when K is positive (by Proposition 9). We analyze each of these issues in turn for general interpretations.

First we note that we have the following:

Proposition 11 *Let $\pi : \mathsf{Lit}_A \to K$ be a K-interpretation. If for every $L \in \mathsf{Lit}_A$ at least one of $\pi(L)$ and $\pi(\neg L)$ is 0 then there exists no sentence φ for which both $\pi[\![\varphi]\!] \neq 0$ and $\pi[\![\neg\varphi]\!] \neq 0$.*

[1] The same terminology is used for logical theories.

Observe that if at least one of $\pi[\![\varphi]\!]$ or $\pi[\![\neg\varphi]\!]$ is 0 then $\pi[\![\varphi]\!] \cdot \pi[\![\neg\varphi]\!] = 0$. If K has no divisors of 0 the converse holds as well. Although the examples described in 1.2 are positive semirings, we are about to introduce, in 3.1, a semiring for FOL provenance that *does* have divisors of 0. For this reason we note also the following:

Proposition 12 *Let $\pi : \mathsf{Lit}_A \to K$ be a K-interpretation. If for every $L \in \mathsf{Lit}_A$ we have $\pi(L) \cdot \pi(\neg L) = 0$ then for any sentence φ we have $\pi[\![\varphi]\!] \cdot \pi[\![\neg\varphi]\!] = 0$.*

Propositions 11 and 12 hold in arbitrary K and each supports a kind of "consistency", with the two kinds coinciding when K has no divisors of 0.

Turning to "completeness", note that if both $\pi[\![\varphi]\!]$ and $\pi[\![\neg\varphi]\!]$ are 0 then $\pi[\![\varphi]\!] + \pi[\![\neg\varphi]\!] = 0$. If K is +-positive then the converse holds as well. However, for arbitrary K, neither an analog of Proposition 11 nor one of Proposition12 holds. Indeed, let $K = \mathbb{Z}_4$. Consider the vocabulary consisting of one unary relation symbol R and let $A = \{c_1, c_2\}$. For the interpretation given by $\pi(\neg R(c_1)) = \pi(\neg R(c_2)) = \pi(R(c_1)) = \pi(R(c_2)) = 2$ and the sentence $\varphi = R(c_1) \wedge R(c_2)$ we have $\pi[\![\varphi]\!] = \pi[\![\neg\varphi]\!] = 0$.

Instead, we have the following for positive semirings.

Proposition 13 *Assume that K is positive. Let $\pi : \mathsf{Lit}_A \to K$ be a K-interpretation. If for every $L \in \mathsf{Lit}_A$ we have $\pi(L) \neq 0$ or $\pi(\neg L) \neq 0$ (equivalently, $\pi(L) + \pi(\neg L) \neq 0$) then for any sentence φ we have $\pi[\![\varphi]\!] \neq 0$ or $\pi[\![\neg\varphi]\!] \neq 0$ (equivalently, $\pi[\![\varphi]\!] + \pi[\![\neg\varphi]\!] \neq 0$).*

3 A Provenance Semiring for FOL

We have claimed Sect. 1.2 that $\mathbb{N}[Y]$, the commutative semiring freely generated by a set Y is used for provenance tracking. The elements of Y label the information whose propagation we wish to capture in provenance. This works fine for *positive* database query languages [16] but difference/negation cause problems. Here we shall use a variation on the idea of polynomials in order to deal with negated facts in provenance analysis.

We construct a semiring whose elements can be identified with certain polynomials that describe the provenance of FOL model checking. The main insight is the use of indeterminates in "positive-negative pairs". We show that the resulting polynomials provide a nicely dual interpretation for provenance that captures model-checking proofs. We illustrate with a running example.

3.1 Dual-Indeterminate Polynomials

Let $X, \bar{X}$ be two disjoint sets together with a one-to-one correspondence $X \longleftrightarrow \bar{X}$. We denote by $p \in X$ and $\bar{p} \in \bar{X}$ two elements that are in this correspondence. We refer to the elements of $X \cup \bar{X}$ as **provenance tokens** as they will be used to label/annotate some of the "data", i.e., literals over some ground values, via the concept of K-interpretation that we defined previously. Indeed, if, as before, we fix a finite non-empty set A and consider $\mathsf{Lit}_A = \mathsf{Facts}_A \cup \mathsf{NegFacts}_A$ then we shall use X for Facts_A and $\bar{X}$ for $\mathsf{NegFacts}_A$. By convention, if we annotate $R(\mathbf{a})$ with the "positive" token p then the "negative" token $\bar{p}$ can only be used to annotate $\neg R(\mathbf{a})$, and vice versa. We refer to p and $\bar{p}$ as *complementary* tokens.

Further, we denote by $\mathbb{N}[X, \bar{X}]$ the quotient of the semiring of polynomials $\mathbb{N}[X \cup \bar{X}]$ by the congruence

generated by the equalities $p \cdot \bar{p} = 0$ for all $p \in X$.[2] Observe that two polynomials $\mathfrak{p}, \mathfrak{q} \in \mathbb{N}[X \cup \bar{X}]$ are congruent if, and only if, they become identical after deleting from each of them the monomials that contain complementary tokens. Hence, the congruence classes in $\mathbb{N}[X, \bar{X}]$ are in one-to-one correspondence with the polynomials in $\mathbb{N}[X \cup \bar{X}]$ such that none of their monomials contain complementary tokens. We shall call these **dual-indeterminate polynomials** although we might often omit "-indeterminate" just use "dual polynomials".

The following is the universality property of the semiring of dual polynomials:

Proposition 14 *For any commutative semiring K and for any $f : X \cup \bar{X} \to K$ such that $\forall p \in X\ f(p) \cdot f(\bar{p}) = 0$ there exists a unique semiring homomorphism $h : \mathbb{N}[X, \bar{X}] \to K$ such that $\forall x \in X \cup \bar{X}\ h(x) = f(x)$.*

We note that $\mathbb{N}[X, \bar{X}]$ is $+$-positive, but not positive, since it has divisors of 0. Examples:

$$p \cdot \bar{p} \ = \ 0, \quad (p + \bar{q})\bar{p}q \ = \ 0, \quad (p\bar{q} + \bar{p}q)(pq + \bar{p}\bar{q}) \ = \ 0.$$

However, keeping both p and $\bar{p}$ around and even using them in certain "inconsistent" $\mathbb{N}[X, \bar{X}]$-interpretations can be very useful in provenance analysis, as we shall see in Sect. 4.1.

Definition 15 *A **provenance-tracking** interpretation is a $\mathbb{N}[X, \bar{X}]$-interpretation $\pi : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ such that $\pi(\mathsf{Facts}_A) \subseteq X \cup \{0, 1\}$ and $\pi(\mathsf{NegFacts}_A) \subseteq \bar{X} \cup \{0, 1\}$.*

The idea is that if π annotates a positive or negative fact with a token, then we wish to track that fact through the model-checking computation. On the other hand annotating with 0 or 1 is done when we do not track the fact, yet we need to recall whether it holds or not in the model.

3.2 An Example and a Characterization

The vocabulary of directed graphs consists one binary predicate E denoting directed edges. Consider, over this vocabulary, the following formula and sentence

$$\mathsf{dominant}(x) \ \equiv \ \forall y \, \big(x = y \vee (E(x, y) \wedge \neg E(y, x))\big), \qquad \varphi \ \equiv \ \forall x \, \neg \mathsf{dominant}(x).$$

$\mathsf{dominant}(x)$ says that in a digraph with edge relation E the vertex x is "dominant" while φ says that the digraph does not have a dominant vertex.

Consider also the digraph $\mathfrak{G}$ depicted in Figure 1 with vertices a, b, c. The edges of the digraph are the solid arrows and we wish to track their presence through model-checking. The dashed arrows corresponds to absent edges, whose absence, however, we also wish to track. We do this with the provenance-tracking $\mathbb{N}[X, \bar{X}]$-interpretation $\beta : \mathsf{Lit}_V \to X \cup \bar{X} \cup \{0, 1\}$ defined by

$$\beta(L) \ = \ \begin{cases} p & \text{if } L = E(a, b) \\ 0 & \text{if } L = \neg E(a, b) \\ q & \text{if } L = E(b, c) \\ 0 & \text{if } L = \neg E(b, c) \\ 0 & \text{if } L = E(a, c) \\ \bar{r} & \text{if } L = \neg E(a, c) \end{cases} \quad = \quad \begin{cases} 0 & \text{if } L = E(c, b) \\ \bar{s} & \text{if } L = \neg E(c, b) \\ t & \text{if } L = E(b, a) \\ 0 & \text{if } L = \neg E(b, a) \\ 0 & \text{for the other positive facts} \\ 1 & \text{for the other negative facts.} \end{cases}$$

[2] This is the same as quotienting by the ideal generated by the polynomials $p\bar{p}$ for all $p \in X$.

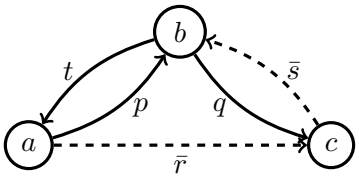


Figure 1: The model $\mathfrak{G}$

So, for example, $\beta(E(c,a)) = \beta(E(c,c)) = \ldots = 0$ and also $\beta(\neg E(c,a)) = \beta(\neg E(c,c)) = \ldots = 1$. Note that β is model-defining in the sense of Definition 8 and that the model it defines is precisely $\mathfrak{G}$.

The assumptions made in the definition of β indicate that we choose to track positive facts like $E(b,c)$ and negative facts like $\neg E(a,b)$, etc., as they are used in establishing the truth of some sentence in $\mathfrak{G}$. They also indicate that we accept, and thus do not track, the absence of the other potential edges such as $E(c,a)$. We think of data annotated with 0 as being "forget-about-it" absent and of data annotated with 1 as "available for free" present.

Clearly, $\mathfrak{G} \models \varphi$, but how can we justify this in terms of the facts, negative or positive, that hold in the model? By computing the semantics of the sentence φ under the interpretation β we will obtain *provenance information* for the result $\mathfrak{G} \models \varphi$. Clearly

$$\mathsf{nnf}(\varphi) \;\equiv\; \forall x\, \exists y \left(x \neq y \wedge (\neg E(x,y) \vee E(y,x))\right)$$

and therefore

$$\beta[\![\varphi]\!] \;=\; \beta[\![\mathsf{nnf}(\varphi)]\!] \;=\; (\bar{r} + t) \cdot p \cdot (1 + q + \bar{s}) \;=\; p\bar{r} + pt + pq\bar{r} + pqt + p\bar{r}\bar{s} + p\bar{s}t.$$

Each of the monomials of the dual polynomial $\beta[\![\varphi]\!]$ has coefficient 1 [3] and each corresponds to a different (model-checking) proof tree of φ from the literals described by the monomial. For example, the monomial pt corresponds to a proof tree of φ in which the fact $E(a,b)$ is used to deny the dominance of b, the fact $E(b,a)$ is used to deny the dominance of a, and the negative fact $\neg E(c,a)$, which is accepted without tracking—it has provenance 1—is used to deny the dominance of c.

Note that what we call proof tree here involves formulae in NNF and has inference rules corresponding to model checking conjunction, disjunction, universal and existential quantifiers. We illustrate with the proof tree corresponding to another monomial, $p\bar{r}\bar{s}$, using the following formula abbreviations:

$$\begin{aligned}
\mathsf{denydom}(x,y) &\equiv \left(x \neq y \wedge (\neg E(x,y) \vee E(y,x))\right) && y \text{ denies dominance of } x \\
\mathsf{notdom}(x) &\equiv \exists y \left(x \neq y \wedge (\neg E(x,y) \vee E(y,x))\right) && x \text{ is not dominant} \\
\mathsf{noVdom} &\equiv \forall x\, \exists y \left(x \neq y \wedge (\neg E(x,y) \vee E(y,x))\right) && \text{no vertex is dominant}
\end{aligned}$$

With these, the proof tree corresponding to $p\bar{r}\bar{s}$ is:

[3] In this example all the monomial coefficients and all the exponents are 1. This is certainly not the case in general. In fact, it is possible to show that any dual polynomial can be computed as some provenance, with suitable choices of sentence, model, and interpretation.

$$\dfrac{\dfrac{a \neq b \qquad \dfrac{\neg E(a,c) \quad [\bar{r}]}{\neg E(a,c) \vee E(c,a)}}{\mathsf{denydom}(a,c)}}{\mathsf{notdom}(a)} \qquad \dfrac{\dfrac{b \neq c \qquad \dfrac{E(a,b) \quad [p]}{\neg E(b,a) \vee E(a,b)}}{\mathsf{denydom}(b,a)}}{\mathsf{notdom}(b)} \qquad \dfrac{\dfrac{c \neq a \qquad \dfrac{\neg E(c,b) \quad [\bar{s}]}{\neg E(c,b) \vee E(b,c)}}{\mathsf{denydom}(c,b)}}{\mathsf{notdom}(c)}$$

$$\dfrac{\mathsf{notdom}(a) \qquad \mathsf{notdom}(b) \qquad \mathsf{notdom}(c)}{\mathsf{noVdom}}$$

The following proposition summarizes the situation.

Proposition 16 *Let $\beta : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ be a provenance-tracking model-defining interpretation, and let φ be an FOL sentence. Then, the dual polynomial $\beta[\![\varphi]\!]$ describes* all *the proof trees that verify φ using premises from among the literals that that β maps to provenance tokens or to 1 (i.e., from the literals that hold in $\mathfrak{A}_\beta$). Specifically, each monomial $m\, x_1^{m_1} \cdots x_k^{m_k}$ corresponds to m distinct proof trees that use m_1 times a literal that β annotates by x_1, ..., and m_k times a literal annotated by x_k, as well as any number of the literals annotated with 1. In particular, $\beta[\![\varphi]\!] \neq 0$ if, and only if, some proof tree exists, and if, and only if, $\mathfrak{A}_\beta \models \varphi$.*

Note that since $\mathbb{N}[X, \bar{X}]$ is not positive this proposition does not follow from Proposition 9. (Nor does this contradict Proposition 10 (b) because provenance-tracking interpretations have a special form.) Nonetheless, albeit not positive, $\mathbb{N}[X, \bar{X}]$ has many remarkable properties and this proposition is a corollary of a more general one that we shall state in Sect. 4.2.

3.3 From Provenance to Confidence

Recall from Sect. 1.2 the Viterbi semiring $\mathbb{V}$. We think of the elements of $\mathbb{V}$ as confidence scores. Going back to the example in Sect. 3.2, and assuming specific confidence scores for the literals that $\mathfrak{G}$ makes true, and that we track, we wish to compute a confidence score for $\mathfrak{G} \models \varphi$.

Specifically, consider the $\mathbb{V}$-interpretation $\gamma : \mathsf{Lit}_V \to [0, 1]$ defined by

$$\gamma(E(a,b)) = \gamma(E(b,c)) = 0.9, \qquad \gamma(E(b,a)) = 0.2, \qquad \gamma(\neg E(a,c)) = \gamma(\neg E(c,b)) = 0.6,$$

and in addition, for any *other* positive fact we have $\gamma(E(\_,\_)) = 0$ and for any *other* negative fact we have $\gamma(\neg E(\_,\_)) = 1$.

With this we could use Definition 2 to compute $\gamma[\![\varphi]\!] \in [0, 1]$, which is the desired confidence score.

However, since we have already computed in Sect. 3.2 the provenance $\beta[\![\varphi]\!]$ we can take advantage of the Fundamental Property (Proposition 5) via a homomorphism whose existence is guaranteed by Proposition 14.

We define $f : X \cup \bar{X} \to [0, 1]$ by

$$f(p) = f(q) = 0.9, \qquad f(t) = 0.2, \qquad f(\bar{r}) = f(\bar{s}) = 0.6,$$

by $f(x) = 0$ for $x \notin \{p, q, t\}$, and by $f(\bar{x}) = 1$ for $\bar{x} \notin \{\bar{r}, \bar{s}\}$. The condition on f in Proposition 14 is satisfied, hence f can be extended to a homomorphism $h : \mathbb{N}[X, \bar{X}] \to \mathbb{V}$. From the definition of f we have $h \circ \beta = \gamma$. By the Fundamental Property

$$\gamma[\![\varphi]\!] \;=\; h(\beta[\![\varphi]\!]).$$

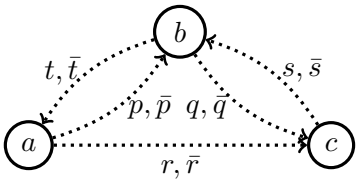


Figure 2: Provenance tracking assumptions

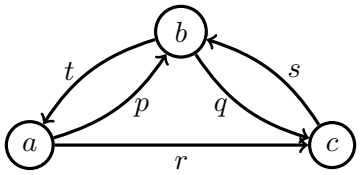


Figure 3: The model $\mathfrak{F}$

Hence the score we wish to compute can be obtained by applying the homomorphism h to the dual polynomial $\beta[\![\varphi]\!] = p\bar{r} + pt + pq\bar{r} + pqt + p\bar{r}\bar{s} + p\bar{s}t$. It is easier to use the factored form of $\beta[\![\varphi]\!]$:

$$h(p(\bar{r}+t)(1+q+\bar{s})) \;=\; 0.9 \cdot \max(0.6,\ 0.2) \cdot \max(1,\ 0.9,\ 0.6) \;=\; 0.54.$$

In general, confidence calculation may be only one of the analyses that we wish to perform. When these analyses are based on semiring calculations we can compute the provenance just once and then evaluate it in multiple semiring and under multiple valuations, by virtue of the Fundamental Property.

3.4 Detailed Provenance Analysis: Top-Secret Proofs

We describe here another kind of provenance analysis that we can perform on in conjunction with interpretation in various semiring. Recall from Sect. 1.2 the access control semiring $\mathbb{A}$. Its elements are interpreted as *clearance levels*, from lowest to highest $\mathsf{P} < \mathsf{C} < \mathsf{S} < \mathsf{T} < 0$. For example, administrators would assign clearance levels to the different items in the input data. The resulting clearance level for the output of a computation determines which users get to access that output. In the context of this paper there would be an assignment of clearance levels to literals.

Going back to the example in Sect. 3.2, consider the $\mathbb{A}$-interpretation $\alpha : \mathsf{Lit}_V \to \mathbb{A}$ defined by

$$\alpha(E(a,b)) = \alpha(E(b,c)) = \alpha(E(b,a)) = \mathsf{P}, \qquad \alpha(\neg E(a,c)) = \alpha(\neg E(c,b)) = \mathsf{T},$$

and in addition, for any *other* positive fact we have $\alpha(E(\_,\_)) = 0$ and for any *other* negative fact we have $\alpha(\neg E(\_,\_)) = \mathsf{P}$.

As in Sect. 3.3 we have $\alpha[\![\varphi]\!] = h(p\bar{r} + pt + pq\bar{r} + pqt + p\bar{r}\bar{s} + p\bar{s}t)$, where h is the unique homomorphism $\mathbb{N}[X, \bar{X}] \to \mathbb{A}$ such that $h(p) = h(q) = h(t) = \mathsf{P}$, $h(\bar{r}) = h(\bar{s}) = \mathsf{T}$, and otherwise equals 0 on the rest of X and equals P on the rest of $\bar{X}$.

We can see that $\alpha[\![\varphi]\!] = \mathsf{P}$ but we can also perform a more detailed analysis in which we can associate clearance levels to individual proof trees Thus, while it will be publicly known that $\mathfrak{G} \models \varphi$, those with top-secret clearance can also know that $p\bar{r}$ describes a proof of the assertion $\mathfrak{G} \models \varphi$. This may become relevant if we have particularly high confidence (as described above in Sect. 3.3) in the literals that p and $\bar{r}$ annotate, that is, in the presence of the edge from a to b and in the absence of an edge from a to c.

4 Reverse Provenance Analysis

There are limitations to what we can do with the provenance of a model-checking assertion $\mathfrak{A} \models \varphi$ for a given $\mathfrak{A}$. It is even more interesting to consider provenance-tracking interpretations that allow us to *choose*, from among multiple models, the ones that fulfill various desiderata.

4.1 A Reverse Analysis Example

Let $V = \{a, b, c\}$ be a set of ground values. As before, these will eventually play the role of the vertices of a digraph. However, we do not yet specify a set of edges, i.e., we do not specify a finite model with universe V. Instead, as illustrated by the dotted edges in Figure 2, we supply a set of provenance tokens $X = \{p, q, r, s, t\}$ that corresponds to the *potential presence* of some edges that we wish to track. Therefore, $\bar{X} = \{\bar{p}, \bar{q}, \bar{r}, \bar{s}, \bar{t}\}$ are the provenance tokens allowing us to track the *potential absence* of the same edges. These **provenance tracking assumptions** can be formalized via a provenance-tracking $\mathbb{N}[X, \bar{X}]$-interpretation.

Define $\pi : \mathsf{Lit}_V \to X \cup \bar{X} \cup \{0, 1\}$ by

$$\pi(L) \;=\; \begin{cases} p & \text{if } L = E(a,b) \\ \bar{p} & \text{if } L = \neg E(a,b) \\ q & \text{if } L = E(b,c) \\ \bar{q} & \text{if } L = \neg E(b,c) \\ r & \text{if } L = E(a,c) \\ \bar{r} & \text{if } L = \neg E(a,c) \end{cases} \quad = \quad \begin{cases} s & \text{if } L = E(c,b) \\ \bar{s} & \text{if } L = \neg E(c,b) \\ t & \text{if } L = E(b,a) \\ \bar{t} & \text{if } L = \neg E(b,a) \\ 0 & \text{for the other positive facts} \\ 1 & \text{for the other negative facts.} \end{cases}$$

So, for example, $\pi(E(c,a)) = \pi(E(c,c)) = \ldots = 0$ and also $\pi(\neg E(c,a)) = \pi(\neg E(c,c)) = \ldots = 1$. This particular interpretation does not feature a positive fact annotated with 1 but we could have just as well had $\pi(E(a,b)) = 1$ and $\pi(\neg E(a,b)) = 0$ if we chose to assume that edge without tracking it.

Note that π is not model-defining (in the sense of Definition 8), i.e., it does not correspond to any *single* model. As we shall see, this is not a bug but a feature (!), as it will allow us to consider, under the given provenance assumptions, multiple models that can satisfy a sentence.

Now we compute the semantics of the sentence φ from Sect. 3.2, under this interpretation and we obtain

$$\pi[\![\varphi]\!] \;=\; (\bar{p} + \bar{r} + t) \cdot (p + \bar{q} + s + \bar{t}) \cdot (1 + q + r + \bar{s}).$$

If we multiply these three expressions and we apply $p\bar{p} = q\bar{q} = r\bar{r} = s\bar{s} = 0$ we get a polynomial with $48 - 4 - 3 - 3 - 4 = 34$ monomials (the reader shall be spared the trouble of admiring it). As in Sect. 3.2, each of these monomials has coefficient 1 and (as shown in Sect. 4.2) each corresponds to a different proof tree of φ from the literals described by the monomial.

For example, the monomial pqt corresponds to a proof tree of φ in which the fact $E(b,a)$ is used to deny the dominance of a, the fact $E(a,b)$ is used to deny the dominance of b, and the fact $E(b,c)$ is used to deny the dominance of c. Recalling the notations from Sect. 3.2, note that the same monomial is part of the dual polynomial $\beta[\![\varphi]\!]$ and that the same proof tree justifies $\mathfrak{G} \models \varphi$. Note also that setting $r = s = \bar{p} = \bar{q} = \bar{t} = 0$ in the definition of π gives the definition of β. Doing the same in $\pi[\![\varphi]\!]$ gives

$$(0 + \bar{r} + t) \cdot (p + 0 + 0 + 0) \cdot (1 + q + 0 + \bar{s}) \;=\; (\bar{r} + t) \cdot p \cdot (1 + q + \bar{s}),$$

which is the same as the polynomial $\beta[\![\varphi]\!]$ obtained with the model-defining interpretation β which corresponds to the model $\mathfrak{G}$. In this sense, π is a "generalization" of β, or, β can be obtained by *specializing* π. All this will be made precise in full generality in Sect. 4.2 while here we explore two other interesting specializations of π.

One of the monomials in $\pi[\![\varphi]\!]$ is $\bar{p}\bar{q}$. This means that we can find a specialization of π that is model-defining and that defines, in fact, a model with *no* positive information, namely the digraph with vertices V and no edges. Hence, denoting with $\mathfrak{E}$ this no-edge model, we have $\mathfrak{E} \models \varphi$. How many proof trees verify that

$\mathfrak{E} \models \varphi$? The specialization π_1 that we are after corresponds to setting $p = q = r = s = t = 0$. This gives

$$\pi_1[\![\varphi]\!] \;=\; (\bar{p} + \bar{r}) \cdot (\bar{q} + \bar{t}) \cdot (1 + \bar{s}),$$

which is a polynomial with 8 monomials, each with coefficient 1. It follows that there are 8 distinct proof trees for $\mathfrak{E} \models \varphi$.

One can also figure out that pqt, prt, qst, rst are among the monomials in $\pi[\![\varphi]\!]$. This means that we can find another specialization of π that is also model-defining and that defines a model with *maximum* positive information (allowed by π), namely the digraph with vertices V and edges $E(a,b), E(b,c), E(a,c), E(c,b)$ and $E(b,a)$. Let's denote with $\mathfrak{F}$ this all-allowed-edges model (see Figure 3). How many proof trees verify that $\mathfrak{F} \models \varphi$? The specialization π_2 that we look for here corresponds to setting $\bar{p} = \bar{q} = \bar{r} = \bar{s} = \bar{t} = 0$. This gives

$$\pi_2[\![\varphi]\!] \;=\; t \cdot (p + s) \cdot (1 + q + r),$$

which is a polynomial with 6 monomials, each with coefficient 1, hence there are 6 proof trees for this.

Finally, we also wish to consider for this example the provenance of the *negation* of the sentence φ considered above, i.e., the sentence $\neg\varphi$ that says that the digraph *has* a dominant vertex:

$$\neg\varphi \;\equiv\; \neg\forall x\, \neg\mathsf{dominant}(x).$$

Since $\mathsf{dominant}(x) \;\equiv\; \forall y\, \big(x \;=\; y \vee (E(x,y) \wedge \neg E(y,x))\big)$ is already in NNF, we have $\mathsf{nnf}(\neg\varphi) \;\equiv\; \exists x\, \mathsf{dominant}(x)$. We compute the semantics of this sentence under the same interpretation:

$$\pi[\![\neg\varphi]\!] \;=\; pr\bar{t} + \bar{p}q\bar{s}t + s\bar{q}\bar{r} \cdot 0 \;=\; pr\bar{t} + \bar{p}q\bar{s}t.$$

Thus, under the provenance tracking assumptions we have made, there are only two proof trees for $\neg\varphi$. [4]

4.2 Properties of Provenance

In this subsection all interpretations are provenance-tracking, unless another semiring is specified. The interpretation exhibited in Sect. 4.1 belongs to a class that merits its own definition.

Definition 17 *A provenance-tracking interpretation $\pi : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ is said to be **model-compatible** if for each fact $R(\mathbf{a})$ one of the following three holds:*

1. *$\exists z \in X$ s.t $\pi(R(\mathbf{a})) = z$ and $\pi(\neg R(\mathbf{a})) = \bar{z}$, or*
2. *$\pi(R(\mathbf{a})) = 0$ and $\pi(\neg R(\mathbf{a})) = 1$, or*
3. *$\pi(R(\mathbf{a})) = 1$ and $\pi(\neg R(\mathbf{a})) = 0$*

As promised, we state a more powerful version of Proposition 16 (which was about provenance-tracking model-defining interpretations).

[4] In the polynomials featured in this example all the monomial coefficients and all the exponents are 1. This is certainly not the case in general. In fact, it can be shown that any dual polynomial results from suitably chosen sentences and interpretations. We shall come back later to this.

Proposition 18 *Let $\pi : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ be a model-compatible interpretation and let φ be an FOL sentence. Then, $\pi[\![\varphi]\!]$ describes* all *the proof trees that verify φ using premises from among the literals that π maps to provenance tokens or to 1. Specifically, each monomial $m\, x_1^{m_1} \cdots x_k^{m_k}$ corresponds to m distinct proof trees that use m_1 times a literal annotated by x_1, ..., and m_k times a literal annotated by x_k., where $x_1, \ldots, x_k \in X \cup \bar{X}$. In particular, when $\pi[\![\varphi]\!] = 0$ no proof tree exists.*

Corollary 19 *Let π be a model-compatible interpretation. Then, the sum of the monomial coefficients in $\pi[\![\varphi]\!]$ counts the number of proof trees that verify φ using premises from among the literals that π maps to provenance tokens or to 1. The same count can be obtained from an $\mathbb{N}$-interpretation as $(h \circ \pi)[\![\varphi]\!] \in \mathbb{N}$ where $h : X \cup \bar{X} \cup \{0, 1\} \to \mathbb{N}$ is defined by $h(0) = 0$ and $h(p) = h(\bar{p}) = h(1) = 1$.*

A model-compatible interpretation may allow the tracking of both a literal and its negation. Therefore, model-compatible interpretations are not model-defining unless they do not make use of provenance tokens at all (in which case they are essentially canonical truth interpretations). Hence, Proposition 16 is not a simple particular case of Proposition 18. Nonetheless, we shall see how model-defining interpretations can be seen as *specializations* of model-compatible interpretations with respect to models that "agree" (i.e., are *compatible*) with them, as defined below.

Definition 20 *Let $\pi : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ be a model-compatible interpretation and let $\mathfrak{A}$ be a model with universe A (same A). We say that $\mathfrak{A}$ is **compatible** with π if $\mathfrak{A} \models L$ for any literal L such that $\pi(L) = 1$. Further, let $\mathsf{Mod}_\pi := \{\mathfrak{A} \mid \mathfrak{A} \text{ is compatible with } \pi\}$.*

For instance, the models shown in Figures 1 and 3 are compatible with the interpretation defined in Sect. 4.1.

Now we can talk about satisfiability and validity *restricted to the class of models that agree with the provenance tracking assumptions* made by an interpretation.

Corollary 21 (to Proposition 18) *Let $\pi : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ be a model-compatible interpretation and let φ be a first-order sentence. Then, φ is Mod_π-satisfiable if, and only if, $\pi[\![\varphi]\!] \neq 0$, and φ is Mod_π-valid if, and only if, $\pi[\![\neg\varphi]\!] = 0$.*

This is not finite satisfiability (shown undecidable by Trakhtenbrot), of course. Even if we map every possible literal to a different provenance token we only decide satisfiability in a model with exactly $|A|$ elements, which is easily in NP (without talking about provenance).

Example 22 *With the same (digraph) vocabulary as in Sect. 3.2 and 4.1 consider the sentence*

$$\tau := \exists x\, \forall y\, E(x, y) \;\to\; \forall y\, \exists x\, E(x, y).$$

This is a well-known tautology (holding in all models, not just in finite ones). Obviously, $\mathsf{nnf}(\neg\tau) = \exists x\, \forall y\, E(x, y)) \wedge \exists y\, \forall x\, \neg E(x, y)$. Now consider $V = \{a, b\}$ and a truth-compatible interpretation π that annotates $E(a, b), E(b, a), E(a, a), E(b, b)$ with p, q, r, s respectively, and the corresponding negated facts with $\bar{p}, \bar{q}, \bar{r}, \bar{s}$. Then

$$\pi[\![\neg\tau]\!] \;=\; (pr + qs)(\bar{q}\bar{r} + \bar{p}\bar{s}) = 0,$$

verifying that τ is Mod_π-valid.

From the provenance analysis of (provenance-restricted) validity/satisfiability that is enabled by Corollary 21 we can obtain a provenance analysis of model checking, for each model of a given sentence, as follows.

Definition 23 *Let π be model-compatible and let $\mathfrak{A} \in \mathsf{Mod}_\pi$. The **specialization** of π with respect to $\mathfrak{A}$ is the $\mathbb{N}[X, \bar{X}]$-interpretation $\pi|_{\mathfrak{A}} : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ defined by*

$$\pi|_{\mathfrak{A}}(L) \;=\; \begin{cases} \pi(L) & \text{if } \mathfrak{A} \models L \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\pi|_{\mathfrak{A}}$ is always model-defining and the model it defines is, of course, $\mathfrak{A}$.

The model-defining interpretation β in Sect. 3.2 is the specialization with respect to the model $\mathfrak{G}$ of the model-compatible interpretation π in Sect. 4.1, $\beta = \pi|_{\mathfrak{G}}$. Other specializations of π are given in Sect. 4.1. The next corollary finally justifies Proposition 16.

Corollary 24 (to Proposition 18) *Let $\pi : \mathsf{Lit}_A \to \mathbb{N}[X, \bar{X}]$ be a model-compatible interpretation, let $\mathfrak{A}$ be structure that is compatible with π, and let φ be a first-order sentence such that $\mathfrak{A} \models \varphi$ (hence, by Corollary 21, $\pi[\![\varphi]\!] \neq 0$).*

Then, $\pi|_{\mathfrak{A}}[\![\varphi]\!] \neq 0$ and every monomial in $\pi|_{\mathfrak{A}}[\![\varphi]\!]$ also appears in $\pi[\![\varphi]\!]$, with the same coefficient.

Moreover, $\pi|_{\mathfrak{A}}[\![\varphi]\!]$ describes all *the proof trees that verify $\mathfrak{A} \models \varphi$. In particular, the sum of all the monomial coefficients in $\pi|_{\mathfrak{A}}[\![\varphi]\!]$ counts the number of distinct such proof trees (as in Corollary 19, the same count can be obtained from an $\mathbb{N}$-interpretation).*

While $\pi|_{\mathfrak{A}}[\![\varphi]\!]$ analyzes the provenance of checking in a specific model, the more general $\pi[\![\varphi]\!]$ allows for a form *reverse analysis*. Indeed, to each monomial $m\, x_1^{m_1} \cdots x_k^{m_k}$ in $\pi[\![\varphi]\!] \neq 0$ we can associate a model from Mod_π that makes true the literals that are annotated by $x_1, \ldots, x_k$ (and possibly more literals) and, as we have seen, every model $\mathfrak{A} \in \mathsf{Mod}_\pi$ such that $\mathfrak{A} \models \varphi$ can be obtained this way.

Example 25 (Example 22 cont'd) *Let us also compute the provenance of the tautology τ itself:*

$$\pi[\![\tau]\!] = (\bar{p} + \bar{r})(\bar{q} + \bar{s}) + (q + r)(p + s).$$

Here Mod_π consists of all possible structures with universe $\{a, b\}$ and, for any such $\mathfrak{A}$, the model-refinement $\pi|_{\mathfrak{A}}$ sets to 0 exactly one of the two tokens in a complementary pair. No matter how this is done, observe that $\pi|_{\mathfrak{A}}[\![\tau]\!] \neq 0$.

4.3 Confidence Maximization

As in Sect. 3.3 we use the Viterbi semiring $\mathbb{V}$ from Sect. 1.2 interpreting its values as confidence scores. Interestingly, we can reverse analyze the provenance polynomials and use confidence scores to find a model in which confidence is maximized.

In the context of the example in Sect. 4.1, suppose that we have confidence $1/3$ in all the literals that the model-compatible interpretation π maps to a (positive or negative) provenance token. This yields a $\mathbb{V}$-interpretation π' which, by Propositions 5 and 14, factors as $\pi' = h \circ \pi$ where h is the unique semiring

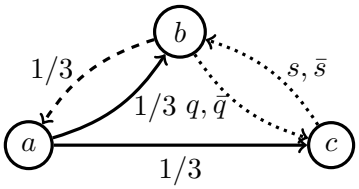


Figure 4: Maximum confidence model with dominant vertex

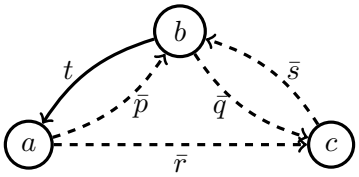


Figure 5: The model $\mathfrak{H}$

homomorphism $\mathbb{N}[X \cup \bar{X}] \to \mathbb{V}$ that maps all the tokens $p, \ldots, \bar{p}, \ldots$ to $1/3$ (this is perfectly plausible, as confidence is *not* probability).

Now recall from Sect. 4.1 the sentence $\neg\varphi$ (which asserts that there exists a dominant vertex). We have computed $\pi[\![\neg\varphi]\!] = pr\bar{t} + \bar{p}q\bar{s}t$. Obviously, π' is inconsistent so further applying $h(pr\bar{t} + \bar{p}q\bar{s}t) = 1/27+1/81 = 4/81$ is not meaningful. However, we know from Corollary 24 that each monomial in $\pi[\![\neg\varphi]\!]$ corresponds to some model of $\neg\varphi$. In this case we have exactly two proof tree choices, corresponding to different models, and they give different confidence to $\neg\varphi$. To maximize confidence we choose the monomial $pr\bar{t}$ therefore a model in which we have an edge $E(a, b)$, an edge $E(a, c)$ and *no* edge $E(b, a)$. This will ensure the dominance of vertex a with confidence $1/27$, in other words, $\neg\varphi$ is $1/27$-true in this model. This model is shown in Figure 4 (the edge $E(b, a)$ is dashed because it is absent but we still wanted to show the confidence 1/3 in this absence). The edges $E(b, c)$ and $E(c, b)$ are dotted because neither their presence nor their absence contradicts the provenance assumptions. We can, in fact, continue with a provenance analysis for these two edges if other properties of the model are of interest.

5 Model Update

In this section we indicate a method for updating provenance polynomials corresponding to a model-defining interpretation when the model associated with the interpretation is updated by inserting or deleting facts.

For example, recall from Sect. 3.2 the interpretation β, the structure $\mathfrak{G}$ that it defines (Figure 1), and the sentence φ asserting "no dominant vertex". We had computed

$$\beta[\![\varphi]\!] = (\bar{r} + t) \cdot p \cdot (1 + q + \bar{s}).$$

First suppose that we update $\mathfrak{G}$ by *deleting* $E(a, b)$ and $E(b, c)$. Keeping the other provenance targets, this results in the model $\mathfrak{H}$ depicted in Figure 5. What is the corresponding update on the dual polynomial $\beta[\![\varphi]\!]$? For the provenance polynomials used for positive queries, as in [16], this update is performed by setting $p = q = 0$. However, this would result in the polynomial 0, which is wrong, because $\mathfrak{H} \models \varphi$.

The right way to perform this update takes advantage of the results in Sect. 4.2. We use the model-compatible interpretation π given in Sect. 4.1 (or any other model-compatible interpretation that both $\mathfrak{G}$ and $\mathfrak{H}$ are compatible with and that specializes with respect to $\mathfrak{G}$ to β). Recall from Sect. 4.1 that

$$\pi[\![\varphi]\!] = (\bar{p} + \bar{r} + t) \cdot (p + \bar{q} + s + \bar{t}) \cdot (1 + q + r + \bar{s}),$$

and therefore

$$\pi|_{\mathfrak{H}}[\![\varphi]\!] \;=\; (\bar{p}+\bar{r}+t)\cdot\bar{q}\cdot(1+\bar{s})$$

is the update we desire. Comparing this with $\beta[\![\varphi]\!]$ shows the need for doing an excursion through π.

Next, suppose that we update $\mathfrak{G}$ by *inserting* $E(a,c)$ and $E(c,b)$ resulting in the model $\mathfrak{F}$ in Figure 3. Then, the update of $\beta[\![\varphi]\!]$ is

$$\pi|_{\mathfrak{F}}[\![\varphi]\!] \;=\; t\cdot(p+s)\cdot(1+q+r).$$

6 Conclusions

The previous work on provenance in databases focused on positive languages, and essentially even on the $\exists, \wedge, \vee$ fragment of first-order logic. But it also focused on Datalog, hence on least fixed points. The presentation in this article should encourage us to extend these studies to the full least fixed-point logic LFP. This will be done in subsequent work, in relationship to games. The model checking games for LFP are parity games (see e.g. [4]), which are much more complicated than the acyclic games with only finite plays that suffice for first-order logic. At this point it is not really clear yet how a provenance analysis for arbitrary parity games can be done, but it is known that, on finite structures, we can restrict LFP to formulae that only make use of positive least fixed-point operators, without losing expressive power. On the game-theoretic side this corresponds to restricting parity games to reachability games (that however may still admit infinite plays), and for these a combination of ω-continuous semirings of formal power series with the idea of dual indeterminates provides a sound mathematical basis for provenance analysis.

Acknowledgements Our collaboration on the topics of this paper started in Fall 2016 as we were both participating in the "Logical Structures in Computation" program at the Simons Institute for the Theory of Computing in Berkeley. We are very grateful to the Institute for support and for the perfect collaborative atmosphere that it fosters. We would like to acknowledge very useful discussions at the Institute with Andreas Blass, Mikołaj Bojańczyk, Thomas Colcombet, Anuj Dawar, Kousha Etessami, Diego Figueira, Phokion Kolaitis, Ugo Montanari, Jaroslav Nešetřil, Daniela Petrişan, and Miguel Romero.

Val Tannen is very grateful to his collaborators in the development over several years of semiring provenance for databases: (in chronological · alphabetical order) T.J. Green, Grigoris Karvounarakis, Zack Ives, Nate Foster, Yael Amsterdamer, Daniel Deutch, Tova Milo, Susan Davidson, Julia Stoyanovich, Sudeepa Roy, and Yuval Moskovitch. He was partially supported by NSF grants 1302212 and 1547360 and by NIH grant U01EB02095401.

References

[1] Y. Amsterdamer, S. B. Davidson, D. Deutch, T. Milo, J. Stoyanovich, and V. Tannen. Putting lipstick on pig: Enabling database-style workflow provenance. *PVLDB*, 5(4):346–357, 2011.

[2] Y. Amsterdamer, D. Deutch, and V. Tannen. On the limitations of provenance for queries with difference. In *3rd Workshop on the Theory and Practice of Provenance, TaPP'11, Heraklion, Crete, Greece, June 20-21, 2011*, 2011. See also CoRR abs/1105.2255.

[3] Y. Amsterdamer, D. Deutch, and V. Tannen. Provenance for aggregate queries. In *Proceedings of the 30th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2011, June 12-16, 2011, Athens, Greece*, pages 153–164, 2011. See also CoRR abs/1101.1110.

[4] K. Apt and E. Grädel, editors. *Lectures in Game Theory for Computer Scientists*. Cambridge University Press, 2011.

[5] S. Bistarelli. *Semirings for Soft Constraint Solving and Programming*, volume 2962 of *Lecture Notes in Computer Science*. Springer, 2004.

[6] S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint satisfaction and optimization. *J. ACM*, 44(2):201–236, 1997.

[7] D. Deutch, T. Milo, S. Roy, and V. Tannen. Circuits for datalog provenance. In *Proc. 17th International Conference on Database Theory (ICDT), Athens, Greece, March 24-28, 2014.*, pages 201–212, 2014.

[8] D. Deutch, Y. Moskovitch, and V. Tannen. Provenance-based analysis of data-centric processes. *VLDB J.*, 24(4):583–607, 2015.

[9] S. Eilenberg. *Automata, Languages, and Machines*. Academic Press, New York, 1974.

[10] J. N. Foster, T. J. Green, and V. Tannen. Annotated XML: queries and provenance. In *Proceedings of the Twenty-Seventh ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2008, June 9-11, 2008, Vancouver, BC, Canada*, pages 271–280, 2008.

[11] F. Geerts and A. Poggi. On database query languages for K-relations. *J. Applied Logic*, 8(2):173–185, 2010.

[12] F. Geerts, T. Unger, G. Karvounarakis, I. Fundulaki, and V. Christophides. Algebraic structures for capturing the provenance of SPARQL queries. *J. ACM*, 63(1):7:1–7:63, 2016.

[13] T. J. Green. Containment of conjunctive queries on annotated relations. *Theory Comput. Syst.*, 49(2):429–459, 2011.

[14] T. J. Green, Z. G. Ives, and V. Tannen. Reconcilable differences. *Theory Comput. Syst.*, 49(2):460–488, 2011.

[15] T. J. Green, G. Karvounarakis, Z. G. Ives, and V. Tannen. Update exchange with mappings and provenance. In *Proceedings of the 33rd International Conference on Very Large Data Bases, University of Vienna, Austria, September 23-27, 2007*, pages 675–686, 2007.

[16] T. J. Green, G. Karvounarakis, and V. Tannen. Provenance semirings. In *Proceedings of the Twenty-Sixth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 11-13, 2007, Beijing, China*, pages 31–40, 2007.

[17] Z. G. Ives, T. J. Green, G. Karvounarakis, N. E. Taylor, V. Tannen, P. P. Talukdar, M. Jacob, and F. C. N. Pereira. The ORCHESTRA collaborative data sharing system. *SIGMOD Record*, 37(3):26–32, 2008.

[18] G. Karvounarakis, Z. G. Ives, and V. Tannen. Querying data provenance. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2010, Indianapolis, Indiana, USA, June 6-10, 2010*, pages 951–962, 2010.

[19] P. G. Kolaitis and M. Y. Vardi. Conjunctive-query containment and constraint satisfaction. *J. Comput. Syst. Sci.*, 61(2):302–332, 2000.

[20] A. Meliou, W. Gatterbauer, and D. Suciu. Reverse data management. *PVLDB*, 4(12):1490–1493, 2011.

[21] A. Meliou and D. Suciu. Tiresias: the database oracle for how-to queries. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24, 2012*, pages 337–348, 2012.

[22] L. Pachter and B. Sturmfels. *Algebraic Statistics for Computational Biology*. Cambridge University Press, 2005.

[23] V. Tannen. Provenance propagation in complex queries. In *In Search of Elegance in the Theory and Practice of Computation - Essays Dedicated to Peter Buneman*, pages 483–493, 2013.

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