Responsibility Measures for Conjunctive Queries with Negation

Meghyn Bienvenu
Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR5800, F-33400 Talence, France

Diego Figueira
Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR5800, F-33400 Talence, France

Pierre Lafourcade
Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR5800, F-33400 Talence, France

Abstract

We contribute to the recent line of work on responsibility measures that quantify the contributions of database facts to obtaining a query result. In contrast to existing work which has almost exclusively focused on monotone queries, here we explore how to define responsibility measures for unions of conjunctive queries with negated atoms ($\mathrm{UCQ}^{\neg}$s). After first investigating the question of what constitutes a reasonable notion of qualitative explanation or relevance for queries with negated atoms, we propose two approaches, one assigning scores to (positive) database facts and the other also considering negated facts. Our approaches, which are orthogonal to the previously studied score of Reshef et al. [16], can be used to lift previously studied scores for monotone queries, known as drastic Shapley and weighted sums of minimal supports (WSMS), to $\mathrm{UCQ}^{\neg}$s. We investigate the data and combined complexity of the resulting measures, notably showing that the WSMS measures are tractable in data complexity for all $\mathrm{UCQ}^{\neg}$s and further establishing tractability in combined complexity for suitable classes of conjunctive queries with negation.

2012 ACM Subject Classification Theory of computation → Data provenance; Information systems → Relational database model; Theory of computation → Incomplete, inconsistent, and uncertain databases; Theory of computation → Logic and databases

Keywords and phrases query responsibility measures, conjunctive queries with negation, non-monotone queries, Shapley value, explainability, weighted sums of minimal supports (WSMS)

Digital Object Identifier 10.4230/LIPIcs.ICDT.2026.20

Related Version *Full version*: `https://arxiv.org/abs/2601.04868` [4]

Funding ANR grants INTENDED (ANR-19-CHIA-0014) and EXPAND (ANR-25-CE23-1215).

☞ This pdf contains internal links: clicking on a notion leads to its *definition*.

1 Introduction

Responsibility measures assign scores to database facts based upon how much they contribute to the obtention of a given query answer, thereby providing a quantitative notion of explanation for query results. There has been significant recent interest in defining and computing responsibility measures [15, 17, 14, 16, 8, 13, 11, 1, 3, 12, 5], largely focusing on classes of monotone queries such as (unions of) conjunctive queries and regular path queries. In the present paper, we investigate responsibility measures for queries *with negations*. The most basic such class is that of conjunctive queries allowing for negated atoms or inequalities – here denoted by $\mathrm{CQ}^{\neg}$ – such as $q(y) = \exists x\ R(x, y) \wedge x \neq y \wedge \neg R(y, x)$. Our results and definitions will focus on the class $\mathrm{CQ}^{\neg}$ and its natural extension $\mathrm{UCQ}^{\neg}$ with unions.

© Meghyn Bienvenu, Diego Figueira, and Pierre Lafourcade;
licensed under Creative Commons License CC-BY 4.0
29th International Conference on Database Theory (ICDT 2026).
Editors: Balder ten Cate and Maurice Funk; Article No. 20; pp. 20:1–20:20
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$q = \exists x\, I(x, \mathit{fish}) \wedge \neg I(x, \mathit{meat})$ $\mathcal{D} =$

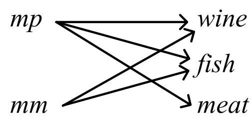

Figure 1 Example database and queries, based on recipes from [9]. All edges are I-tuples, and recipe names *Matelote Meunière* and *Matelote Pochouse* are abbreviated to *mm* and *mp* respectively.

We begin our study by first asking a fundamental question (of independent interest): What are the database entities over which the responsibility should be distributed? In other words: What are the so-called "relevant" facts for a given query being true? Usually, "relevant" facts are defined as those being part of an "explanation". But then: what is a good notion of an "explanation" for queries with negation? In the case of (Boolean) monotone queries – such as conjunctive queries without negated atoms – a simple yet natural notion of qualitative explanation is that of a *minimal support*, *i.e.*, a subset-minimal set of facts making the query true. Through this lens, a fact is relevant to a monotone query whenever it is contained in a minimal support thereof, which witnesses its active participation in making the query true[1]. However, we shall argue that for non-monotone queries, there is no unique sensible definition neither of "relevance" nor of "explanation", even when restricted to the rather basic class of $\mathsf{CQ}^{\neg}$. Instead, we shall discuss several candidate definitions and compare their properties, leading us to focus on two notions of support (each defining a corresponding notion of relevance): signed supports and positive supports.

Let us begin with a concrete example, depicted in Figure 1. The database $\mathcal{D}$ we consider has a single binary relation I for *'has as Ingredient'*, and for now we focus on the Boolean query q expressing "there exists a recipe with *fish* but not *meat* as ingredient". Note that $\mathcal{D} \models q$ via the (only) satisfying assignment $\{x \mapsto mm\}$. One might naturally consider $\{I(mm, \mathit{fish})\}$ as an explanation for $\mathcal{D} \models q$, since it is a minimal sub-database that makes q true. But then, by the same token, one might be led to conclude that $\{I(mp, \mathit{fish})\}$ is an explanation, even if it does not participate in any satisfying assignment! While one may conceive of scenarios in which it may be reasonable to regard $\{I(mp, \mathit{fish})\}$ as a valid explanation – and thus to consider $I(mp, \mathit{fish})$ as being "relevant" – we would argue that it is desirable to define notions of explanation and relevance under which a fact is deemed "non-relevant" (and hence be assigned zero responsibility) whenever it does not participate in the satisfaction of the query, as is the case of $I(mp, \mathit{fish})$.

The reason why we may consider $\{I(mm, \mathit{fish})\}$ to be an explanation but not $\{I(mp, \mathit{fish})\}$ lies, of course, in the *absence* of $I(mm, \mathit{meat})$ in the database. Hence, one way of explaining why the query is true is by using the information of both the presence and the absence of database facts. From this perspective, an explanation for the example query would be $\{+I(mm, \mathit{fish}), -I(mm, \mathit{meat})\}$, which can be read "there is a fact $I(mm, \mathit{fish})$ in the database, but no fact $I(mm, \mathit{meat})$". The 'signed facts' approach thus postulates that, since the satisfaction of the query depends both on the presence and on the absence of facts, the responsibility should be distributed over signed facts (rather than just "positive" facts). This idea can be formalized by viewing a database as a set of signed facts (using signed relations $+P$ and $-P$, for each original relation P) and defining minimal signed supports as minimal sets of signed facts that satisfy the query (also rephrased using the signed relations), where

[1] This notion of relevance for monotone queries admits multiple equivalent characterizations and notably coincides with the notion of *actual cause* from causal responsibility [15].

a fact is signed-relevant if it belongs to some such set. Conveniently, this approach yields a direct reduction to the monotone case, thereby enabling the reuse of existing results and algorithms for monotone queries.

It may not always be desirable, however, to attribute responsibility to entities with no concrete materialization, such as absent facts. Further, it may not always be practical or realistic to use this signed facts approach, due to the sheer number of negated facts that need to be considered, which leads us to our second proposal. Returning to our example, we may consider $S = \{I(mm, \mathit{fish})\}$ to be an explanation because there is a variable assignment that maps the positive query atoms to S in such a way that the negated atoms are satisfied *in the full database* (*i.e.*, the variable assignment continues to witness the satisfaction of the query once the remaining database facts are incorporated). We call subsets satisfying this condition (minimal) positive supports since only the original (positive) database facts may take part in the support. Observe that $\{I(mp, \mathit{fish})\}$ does not count as a positive support, since the only way to satisfy q in $\{I(mp, \mathit{fish})\}$ is to map x to mp, but this assignment does not satisfy $\neg I(x, \mathit{meat})$ in $\mathcal{D} = \{I(mp, \mathit{fish}), I(mp, \mathit{meat})\}$. When taking minimal positive supports as our notion of explanation, we thus obtain a unique explanation for q in $\mathcal{D}$, namely $\{I(mm, \mathit{fish})\}$. We will then say that $I(mm, \mathit{fish})$ is the sole positive-relevant fact for q in $\mathcal{D}$.

These are the two viewpoints advanced in this paper: the signed facts approach and positive support approach. For both approaches, we show how the notions of support and relevance can be used to adapt existing Shapley-value-based responsibility measures to unions of $\mathrm{CQ}^{\neg}$s. We consider in particular the "drastic-Shapley" responsibility measure, a well-studied measure [14] for Boolean non-numeric queries which in the case of a monotone query q assigns a fact $\alpha \in \mathcal{D}$ the proportion of the linear orderings $(\mathcal{D}, <)$ for which $\{\beta \in \mathcal{D} : \beta < \alpha\} \not\models q$ and $\{\beta \in \mathcal{D} : \beta \leqslant \alpha\} \models q$. We also consider the 'MS-Shapley' measure, recently introduced in [5] as an alternative Shapley-based measure for Boolean monotone queries and shown to enjoy appealing theoretical and algorithmic properties. It assigns a fact $\alpha \in D$ the sum, over all minimal supports $S \subseteq \mathcal{D}$ of q containing α, of $\frac{1}{|S|}$ and it represents the simplest instance of the family of weighted sums of minimal supports (or WSMS) measures [5].

Related Work. The most relevant prior work is undoubtedly that of Reshef, Kimelfeld, and Livshits [16], which studies a responsibility measure that adapts the "drastic-Shapley" for monotone queries to handle conjunctive queries with negation (but without inequalities). The notion of relevance underlying their measure is in terms of the "impact" that the introduction of the fact makes in a sub-database and is orthogonal to the notions of signed-relevance and positive-relevance that we propose in this paper. In particular, a fact such as $I(mp, \mathit{fish})$ (from the example in Figure 1) would be deemed relevant and receive a non-null responsibility score in their approach. We shall dedicate Section 5 to a comparison with their approach.

Contributions. The foremost contribution of our work is of a conceptual nature, namely, the introduction of novel responsibility measures for (unions of) conjunctive queries with negations, which are grounded in simple and intuitive notions of supports and relevance. Our main technical contribution is the study of the computational complexity of computing the new measures, focusing primarily on the identification of tractable cases.

In more detail, Section 3 formalizes the notions of signed support and signed-relevance, where negative facts are treated as first-class citizens. We show that for $\mathrm{CQ}^{\neg}$s these notions can be equivalently defined in terms of logical entailment, but that for $\mathrm{UCQ}^{\neg}$s, the entailment-based definition yields different notions that we argue are less intuitive. We therefore choose signed supports as the basis for our definitions of novel drastic-Shapley and MS-Shapley-like measures for $\mathrm{UCQ}^{\neg}$s. In Sections 3.2 and 3.3, we present several complexity results that identify classes of queries for which we can tractably compute the latter measures, which leverage the existence of a straightforward reduction to existing measures for queries without negation. In particular, we show that the "signed" version of the MS-Shapley measure (and more generally, WSMS measures) enjoys tractable (FP) data complexity for all $\mathrm{UCQ}^{\neg}$s.

Section 4 formalizes the positive support approach. Here again we consider and reject an alternative formalization – based upon monotonicity of supports – which we show to be less intuitive. We thus adopt positive supports (and positive-relevance) as the basis for defining new drastic-Shapley and MS-Shapley-like measures for $\mathrm{UCQ}^{\neg}$s, which attribute responsibility only to the (positive) database facts. In Section 4.2 and Section 4.3, we present complexity results for both measures, including a general tractability result in data complexity for the "positive" variant of the MS-Shapley measure.

Finally, in Section 5, we discuss the connection with the closest prior work [16] and compare their definitions with our own, both conceptually and algorithmically. We prove in particular that the notion of relevance that underlies the measure of [16] is incomparable with both positive-relevance and signed-relevance.

2 Preliminaries

We fix disjoint infinite sets Const, Var of *constants* and *variables*, respectively. For any syntactic object O (*e.g.* database, query), we will use $\mathit{var}(O)$ and $\mathit{const}(O)$ to denote the sets of variables and constants contained in O.

A *(relational) schema* is a finite set of relation symbols, each symbol R associated with a (positive) *arity* $\mathrm{arity}(R)$. A *(relational) atom* over a schema Σ takes the form $R(\bar{t})$ where R is a *relation name* from Σ of some arity k, and $\bar{t} \in (\mathrm{Const} \cup \mathrm{Var})^k$. A *fact* is an atom which contains only constants. A *database* $\mathcal{D}$ over a schema Σ is a finite set of facts over Σ we call $\mathit{const}(\mathcal{D})$ its *active domain*. When we speak of the size of a database or set of atoms, we mean the number of atoms (or facts) it contains.

Conjunctive Queries with Negations. A conjunctive query with negated atoms and inequalities, or *$CQ^{\neg}$*, is a first-order query of the form $\exists \bar{x}\ \alpha_1 \wedge \cdots \wedge \alpha_n$ where each α_i can be (1) a (positive) atom, (2) a *negated atom* of the form $\neg\alpha$ where α is an atom, or (3) an *inequality atom* of the form $t \neq t'$, where t, t' are terms. We further impose the condition that each variable appearing in a negated or inequality atom must also appear in a positive atom, commonly known as the restriction to *safe negations*. We shall also consider *$UCQ^{\neg}$*s, which are defined as finite unions of $\mathrm{CQ}^{\neg}$s, as well as *$CQ^{\neq}$*s and *$UCQ^{\neq}$*s obtained by disallowing negated atoms in $\mathrm{CQ}^{\neg}$s and $\mathrm{UCQ}^{\neg}$s, and *CQ* and *UCQ* obtained by further restricting $\mathrm{CQ}^{\neq}$s and $\mathrm{UCQ}^{\neq}$s to have only positive atoms.

A query is *Boolean* if it has no free variables, in which case we use the notation $\mathcal{D} \models q$ to indicate that q holds (or is satisfied) in $\mathcal{D}$. A *satisfying assignment* for a $\mathrm{UCQ}^{\neg}$ q on a database $\mathcal{D}$ is as expected a valuation $\nu : \mathit{var}(q) \to \mathcal{D}$ making the query true on $\mathcal{D}$. When q is a CQ, we shall also call the satisfying assignment a *homomorphism* (and write $q \xrightarrow{\text{hom}} \mathcal{D}$) and we extend the definition to sets of atoms in the obvious way. A Boolean query is *monotone* if $\mathcal{D} \models q$ and $\mathcal{D} \subseteq \mathcal{D}'$ implies $\mathcal{D}' \models q$ for all databases $\mathcal{D}, \mathcal{D}'$. A *support* of q in $\mathcal{D}$ is any set $S \subseteq \mathcal{D}$ such that $S \models q$, and such a subset S is a *minimal support* if it is further subset-minimal[2] among all supports of q in $\mathcal{D}$.

[2] Throughout the paper, we will always use *minimal* to mean minimal *w.r.t.* set inclusion.

We recall some relevant structural restricted classes of CQs. The *generalized hypertree width* of a CQ is a classic measure of tree-likeness. We refer readers to [10, Definition 3.1] for a definition but simply note that bounded generalized hypertree width is a sufficient condition for tractable CQ evaluation and notably generalizes the class of "acyclic" CQs, which correspond to CQs of generalized hypertree width 1. We say that two distinct atoms α, β of a CQ q are *mergeable* if there are two homomorphisms $h_\alpha : \alpha \xrightarrow{\text{hom}} \mathcal{D}$, $h_\beta : \beta \xrightarrow{\text{hom}} \mathcal{D}$ to an arbitrary database $\mathcal{D}$ such that $h_\alpha(\alpha) = h_\beta(\beta)$ (in particular they must have the same relation name). An atom α is (individually) mergeable if it is mergeable with some other atom in q. The *self-join width* of a CQ q, defined in [5], is the cardinality of $\{t \in \mathsf{terms}(\alpha) : \alpha \text{ is a mergeable atom of } q\}$. The class of CQs with self-join width 0 generalizes the class of *self-join free* CQs (*sjf-CQ*), defined as those CQs which do not contain two distinct atoms with the same relation name. To see why the two classes do not coincide, consider the query $\exists xy\, R(a, x) \land R(b, y)$ which has no mergeable atoms (hence self-join width 0) but does have a self-join.

Responsibility Measures via Shapley Values. We shall be interested in *responsibility measures*, defined as functions ϕ which take as input a database $\mathcal{D}$, a (possibly non-Boolean) query q, an answer $\bar{a}$ to q in $\mathcal{D}$, and a fact $\alpha \in \mathcal{D}$, and which output a quantitative score measuring how much α contributes to $\bar{a} \in q(\mathcal{D})$. We may however simplify the presentation by replacing the input answer $\bar{a}$ and (possibly non-Boolean) query q by the associated Boolean query $q(\bar{a})$, defined by letting $\mathcal{D} \models q(\bar{a})$ iff $\bar{a} \in q(\mathcal{D})$. In this manner, we can eliminate the answer tuple from the arguments of the responsibility measure and work instead with ternary responsibility measures, *i.e.*, $\phi(\mathcal{D}, q, \alpha)$ with Boolean q. Henceforth, we shall thus *assume w.l.o.g. that the input query is always Boolean.*

We study responsibility measures based upon the well-known Shapley value, which was originally defined for cooperative games but can be translated into the database setting by modelling the query as a wealth function. In the context of databases, a *wealth function* for a (Boolean) query q and database $\mathcal{D}$ is a function $\xi : 2^{\mathcal{D}} \to \mathbb{R}$ outputting a number for each set S of facts in the database, which captures in some manner how S contributes to the satisfaction of q. The Shapley value can then be used for transforming such a function on subsets of $\mathcal{D}$ into a responsibility measure for individual facts in $\mathcal{D}$. Concretely, the Shapley value of a fact $\alpha \in \mathcal{D}$ and a wealth function ξ (encoding the query), denoted by $\mathrm{Sh}(\mathcal{D}, \xi, \alpha)$, is the average wealth contribution of α over all the linear orderings $\mathcal{O}$ of $\mathcal{D}$, given by

$$\mathrm{Sh}(\mathcal{D}, \xi, \alpha) \doteq \frac{\sum_{(\mathcal{D},<)\in\mathcal{O}} \xi(\{\beta \in \mathcal{D} : \beta < \alpha\} \cup \{\alpha\}) - \xi(\{\beta \in \mathcal{D} : \beta < \alpha\})}{|\mathcal{O}|}.$$

Most work to date on Shapley-based responsibility measures employs the drastic wealth function ξ_q^{dr} defined by setting $\xi_q^{\mathsf{dr}}(S) = 1$ if S is a support of q (*i.e.*, $S \models q$) or 0 otherwise.[3] More recently, [5] introduced the MS wealth function ξ_q^{ms} which outputs, for a monotone query q, the number $\xi_q^{\mathsf{ms}}(S)$ of minimal supports of q inside S.

Given a class of queries $\mathcal{C}$, we denote by $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{ms}}$ (*resp.* $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{dr}}$) the associated computational problem of computing, for a given query $q \in \mathcal{C}$, database $\mathcal{D}$ and fact $\alpha \in \mathcal{D}$, the value $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{ms}}, \alpha)$ (*resp.* $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \alpha)$), which we shall call the *MS-Shapley* (*resp.* *drastic-Shapley*) responsibility measures or scores. We will naturally extend this notation $\mathrm{SVC}_{\mathcal{C}}^{\star}$ to other wealth function families $\{\xi_q^{\star}\}_{q\in\mathcal{C}}$ introduced in this paper, simply by varying the $\star$ superscript.

[3] In the original formulation of [14], the wealth function outputs 1 *iff* $S \cup X \models q$ and $X \not\models q$, where $X \subseteq \mathcal{D}$ is a special set of facts called "exogenous". For simplicity, our work considers plain databases (without exogenous facts), see Remark 2.2, so we have adjusted the function accordingly.

The MS-Shapley measure belongs to a larger family of responsibility measures based upon aggregating the number and sizes of minimal supports, introduced in [5] as weighted sums of minimal supports, or *WSMS*. Indeed, it has been shown in [5, Proposition 4.3] that $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{ms}}, \alpha)$ can be equivalently and more simply defined as being the following sum:

$$\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{ms}}, \alpha) = \sum \{ \tfrac{1}{|S|} : S \text{ minimal support of } q \text{ in } \mathcal{D} \text{ s.t. } \alpha \in S \}. \quad (1)$$

Other WSMS measures are obtained by replacing $\frac{1}{|S|}$ with any positive *weight function* $f(|S|)$. Every such WSMS measure can be equivalently defined as the Shapley value of some suitably chosen wealth function [5, Proposition 4.4].

For the sake of readability, we shall phrase our definitions and results using the drastic and MS-Shapley responsibility measures, but we emphasize that this is merely to enhance readability, as all of our results for MS-Shapley measures extend to any WSMS measure based on a tractable weight function f, see Section 6 and [4, §E.1] for more details.

In the present work, we will build upon the following known result ensuring tractability of $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{ms}}$ and other WSMS measures in combined complexity.

▶ **Theorem 2.1** ([5, Theorem 6.6 with Lemma 5.1])**.** *For any class $\mathcal{C}$ of CQs having bounded generalized hypertree width and bounded self-join width, the problem of counting the number of minimal supports of a given size is in polynomial time. Further, $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{ms}}$ is in polynomial time in combined complexity. These results extend also to $\mathrm{CQ}^{\neq}$ queries and other WSMS measures based upon tractable weight functions.*

▶ Remark 2.2 (Exogenous facts). Prior work on database responsibility measures has considered a more complex setting in which the input database is partitioned into "exogenous" and "endogenous" facts, where the exogenous facts are treated as given and the responsibility is distributed only among endogenous facts. To keep the focus on the handling of negation, we decided to keep definitions to their most basic form by working with plain (unpartitioned) databases (i.e. treating all database facts as endogenous). It is important to note that while tractability results obtained for partitioned databases also apply to the simpler setting without exogenous facts, this is not the case for intractability results. Indeed, all hardness results concerning UCQs that have been proven for drastic-Shapley computation crucially rely upon the presence of exogenous facts, and it is an interesting open problem whether there exists a CQ q for which $\mathrm{SVC}_q^{\mathsf{dr}}$ is #P-hard in the purely endogenous setting.

3 Responsibility Measures via Signed Supports

In this section, we formalize the "signed fact approach" sketched in the introduction and analyze the data and combined complexity of the resulting responsibility measures.

3.1 Relevance and Responsibility of Signed Facts

Given a schema Σ, let $\Sigma^{\pm}$ be the schema having $+R$ and $-R$ for every R in Σ with the same arity as R. For every atom $R(\bar{t})$ in the schema Σ, let $+R(\bar{t})$ and $-R(\bar{t})$ denote the corresponding atoms over the schema $\Sigma^{\pm}$, which we shall call *positive* and *negative atoms* (not to confuse with negated atoms).[4] We generally use the term *signed* to refer to such atoms or facts, and we will extend this notation to sets of facts S by letting $+S \triangleq \{+\alpha : \alpha \in S\}$ and $-S \triangleq \{-\alpha : \alpha \in S\}$. Given a database $\mathcal{D}$ over a schema Σ, we let $\mathcal{D}^{\pm}$ be the database over $\Sigma^{\pm}$ defined as follows:

$$\mathcal{D}^{\pm} \triangleq +\mathcal{D} \cup \{-R(\bar{c}) : R \in \Sigma, \mathrm{arity}(R) = |\bar{c}|, \mathit{const}(\bar{c}) \subseteq \mathit{const}(\mathcal{D}), \text{ and } R(\bar{c}) \notin \mathcal{D}\}.$$

[4] In particular *positive* and *negative facts* are facts over $+$ and $-$ relations, respectively.

$q = \exists xyz\, E(x,y) \land E(y,z) \land \neg E(z,x) \land x \neq z$

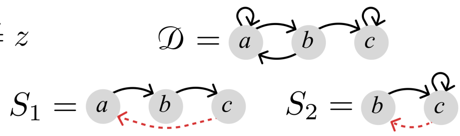

Figure 2 Query and database from Example 3.1, together with the minimal signed supports S_1 and S_2. Solid black arrows represent positive facts and dotted red arrows negative facts.

For any set S of signed facts, let S^+ and S^- be the sets of (unsigned) facts which appear positively and negatively in S, respectively, where we shall write $\mathcal{D}^-$ instead of $(\mathcal{D}^\pm)^-$ for brevity (note that $(\mathcal{D}^\pm)^+$ is simply $\mathcal{D}$). Observe that all constants occurring in $\mathcal{D}^\pm$ belong to the active domain of $\mathcal{D}$, that is, we view negation with a "closed-world" point of view, and in line with the restriction on safe negations.

Given a Boolean UCQ$^\neg$ query q over Σ, let $q^\pm$ be the UCQ$^{\neq}$ over $\Sigma^\pm$ resulting from replacing in q each negated atom $\neg R(\bar{t})$ with the (positive) atom $-R(\bar{t})$ over $\Sigma^\pm$, and each positive atom $R(\bar{t})$ with $+R(\bar{t})$.[5] Observe that

$$\mathcal{D} \models q \quad \Leftrightarrow \quad \mathcal{D}^\pm \models q^\pm$$

This naturally suggests the following notion of support for signed facts: a set $S \subseteq \mathcal{D}^\pm$ of signed facts is a *signed support* of q in $\mathcal{D}$ if $S \models q^\pm$, which is equivalent to requiring the set S to be a support of the monotone query $q^\pm$ in $\mathcal{D}^\pm$. We argue that a minimal signed support is a sensible definition for a qualitative explanation for why q holds in $\mathcal{D}$, since it contains the minimal information on which facts should be present and absent to make the query true in the context of $\mathcal{D}$. This in turn leads us to define the corresponding notion of relevance: a signed fact is *signed-relevant* for q in $\mathcal{D}$ if it belongs to a minimal signed support of q in $\mathcal{D}$. We illustrate these notions on an example:

▶ **Example 3.1.** Consider the CQ$^\neg$ q of Figure 2 over directed graphs (with edge relation E), and the database $\mathcal{D}$ given by the depicted graph. Under the signed facts approach, there are two possible explanations for q given by the minimal signed supports $S_1 = \{+E(a,b), +E(b,c), -E(c,a)\}$ and $S_2 = \{+E(b,c), +E(c,c), -E(c,b)\}$. It follows that the signed-relevant facts are $+E(a,b), +E(b,c), +E(c,c), -E(c,a), -E(c,b)$, and the other signed facts in $\mathcal{D}^\pm$, like $+E(b,a)$ and $-E(b,b)$, would be deemed irrelevant under this approach. ⌟

The following lemma clarifies how signed supports relate to finite entailment.

▶ **Lemma 3.2.** *For every UCQ$^\neg$ q, database $\mathcal{D}$, and set $S \subseteq \mathcal{D}^\pm$, if S is a signed support of q in $\mathcal{D}$, then $\mathcal{D}' \models q$ for every database $\mathcal{D}'$ with $S^+ \subseteq \mathcal{D}'$ and $\mathcal{D}' \cap S^- = \varnothing$ (equivalently, $S^+ \cup \{\neg R(\bar{t}) : R(\bar{t}) \in S^-\} \models^{\mathsf{fin}} q$). However, the converse does not hold even for CQ$^\neg$s.*

To see why the converse statement in Lemma 3.2 fails and why an alternative definition of minimal support via finite entailment can yield unexpected results, it is instructive to consider the following example.

▶ **Example 3.3.** Consider the CQ$^\neg$ $q = \exists x, y\ A(x) \land R(x,y) \land \neg A(y)$ and database $\mathcal{D} = \{A(b), R(b,c), A(c), R(c,d)\}$. The only minimal signed support is $\{+A(c), +R(c,d), -A(d)\}$. However, if we were to adopt the *entailment semantics* that defines minimal supports

[5] Recall that q may also have inequalities $x \neq y$, in which case $q^\pm$ will also contain them.

using finite entailment (cf. Lemma 3.2), then there would be a further minimal support $S = \{+A(b), +R(b,c), +R(c,d), -A(d)\}$. Indeed, for any database $\mathcal{D}'$ with $S^+ \subseteq \mathcal{D}'$ and $\mathcal{D} \cap S^- = \varnothing$, one of two things must happen: either (i) $A(c) \in \mathcal{D}'$, in which case $\mathcal{D}' \models q$ by $(x,y) \mapsto (c,d)$, or (ii) $A(c) \notin \mathcal{D}'$, in which case we still have $\mathcal{D}' \models q$, this time by $(x,y) \mapsto (b,c)$. This is at odds with the fact that since $A(c) \in \mathcal{D}$ there is no satisfying valuation for q that involves the constant b and hence $A(b)$ is not useful for obtaining q in the considered database $\mathcal{D}$. ⌟

In addition to yielding an arguably unnatural notion of relevance, the entailment semantics has another disadvantage, namely, it can yield minimal supports of unbounded size:

▶ **Example 3.4.** Reconsider the CQ$^\neg$ $q = \exists x, y\ A(x) \wedge R(x,y) \wedge \neg A(y)$, and for each $n \geqslant 1$, define the database $\mathcal{D}_n = \{A(c_i) : 0 \leqslant i < n\} \cup \{R(c_i, c_{i+1}) : 0 \leqslant i < n\}$ (note that $\mathcal{D}_n$ does not contain $A(c_n)$). Then for every $n \geqslant 1$, the entailment semantics would give rise to the minimal support $S_n = \{+A(c_0), -A(c_n)\} \cup \{+R(c_i, c_{i+1}) : 0 \leqslant i < n\}$. ⌟

We find it both unintuitive and computationally complex to use a notion of explanation for UCQ$^\neg$s that produces explanations whose number of atoms do not depend on the size of the query. For this reason, we will not consider this alternative notion further and instead retain our original definition of signed support defined in terms of minimal subsets of $\mathcal{D}^\pm$ that satisfy (rather than entail) q.

With suitable notions of signed-relevance and signed supports at hand, we can now introduce responsibility measures for signed facts, by simply applying existing responsibility measures for monotone queries to the signed version of the query ($q^\pm$) and input database ($\mathcal{D}^\pm$). In particular, we will be interested in the following Shapley-value-based measures:

- *MS-Shapley measure for signed facts*: $\mathrm{Sh}(\mathcal{D}^\pm, \xi^{\mathsf{ms}}_{q^\pm}, \alpha)$
- *Drastic-Shapley measure for signed facts*: $\mathrm{Sh}(\mathcal{D}^\pm, \xi^{\mathsf{dr}}_{q^\pm}, \alpha)$

Observe that since $q^\pm$ is monotone, these measures always return non-negative numbers.

Moreover, we can show that these measures allow us to decide signed-relevance:

▶ **Lemma 3.5.** *For every database $\mathcal{D}$, signed fact $\alpha \in \mathcal{D}^\pm$, and UCQ$^\neg$ q: α is signed-relevant for q in $\mathcal{D}$ iff $\mathrm{Sh}(\mathcal{D}^\pm, \xi^{\mathsf{ms}}_{q^\pm}, \alpha) > 0$ iff $\mathrm{Sh}(\mathcal{D}^\pm, \xi^{\mathsf{dr}}_{q^\pm}, \alpha) > 0$.*

▶ **Example 3.6.** Reconsider the CQ$^\neg$ q and database $\mathcal{D}$ from Figure 2. Under the MS-Shapley measure for signed facts, the positive fact $+E(b,c)$ would get value $2/3 = 1/3 + 1/3$ according to Equation (1) since it participates in two minimal signed supports, both of size 3. Each of the positive facts $+E(a,b)$ and $+E(c,c)$ and the negative facts $-E(c,a)$ and $-E(c,b)$ receives a value $1/3$, corresponding to a participation in a single minimal signed support of size 3. The remaining signed facts (which are not signed-relevant) all receive a null score. ⌟

The remainder of this section will be devoted to studying the complexity of computing responsibility scores for signed facts. Formally, for a class $\mathcal{C}$ of Boolean UCQ$^\neg$ queries, we denote by $\pm\mathrm{SVC}^{\mathsf{ms}}_{\mathcal{C}}$ the problem of computing $\mathrm{Sh}(\mathcal{D}^\pm, \xi^{\mathsf{ms}}_{q^\pm}, \alpha)$ given $\mathcal{D}$, $\alpha \in \mathcal{D}^\pm$ and $q \in \mathcal{C}$; and similarly for $\pm\mathrm{SVC}^{\mathsf{dr}}_{\mathcal{C}}$ with $\xi^{\mathsf{dr}}_{q^\pm}$. We shall investigate these problems both in *data* and *combined complexity*, depending on whether the query is considered to be fixed or not.

We first observe that we can restrict to having only negative facts of relations which appear under a negation in the query. Let $\mathcal{N}_q$ be the set of relations that appear in negated atoms in q, and let $\mathcal{D}^\pm_q \subseteq \mathcal{D}^\pm$ be $\mathcal{D}^\pm \setminus \{-R(\bar{c}) : R \notin \mathcal{N}_q\}$.

▶ **Lemma 3.7.** *Every minimal signed support of $\mathcal{D}, q$ is in $\mathcal{D}^\pm_q$.*

We say that a class $\mathcal{C}$ of UCQ$^\neg$ has *bounded negative arity* if there is a bound N such that arity$(R) \leqslant N$ for all $q \in \mathcal{C}$ and $R \in \mathcal{N}_q$. The following lemma formalizes a simple and useful (many-one) reduction to queries without negation:

▶ **Lemma 3.8.** *For any class $\mathcal{C} \subseteq$ UCQ$^\neg$ of bounded negative arity, there are polynomial-time reductions from $\pm\text{SVC}^{\textsf{ms}}_{\mathcal{C}}$ to $\text{SVC}^{\textsf{ms}}_{\mathcal{C}^\pm}$ and from $\pm\text{SVC}^{\textsf{dr}}_{\mathcal{C}}$ to $\text{SVC}^{\textsf{dr}}_{\mathcal{C}^\pm}$, where $\mathcal{C}^\pm = \{q^\pm : q \in \mathcal{C}\}$.*

Proof. Given a UCQ$^\neg$ $q \in \mathcal{C}$, a database $\mathcal{D}$, and signed fact $\alpha \in \mathcal{D}^\pm$, we need to compute $\text{Sh}(\mathcal{D}^\pm, \xi^{\textsf{ms}}_{q^\pm}, \alpha)$. First observe that since $\mathcal{C}$ is assumed to have bounded negative arity, we can build $\mathcal{D}^\pm_q$ in polynomial time. Moreover, by Lemma 3.7, we can restrict our attention to $\mathcal{D}^\pm_q$ instead of $\mathcal{D}^\pm$, *i.e.*, $\text{Sh}(\mathcal{D}^\pm, \xi^{\textsf{ms}}_{q^\pm}, \alpha) = \text{Sh}(\mathcal{D}^\pm_q, \xi^{\textsf{ms}}_{q^\pm}, \alpha)$ and $\text{Sh}(\mathcal{D}^\pm, \xi^{\textsf{dr}}_{q^\pm}, \alpha) = \text{Sh}(\mathcal{D}^\pm_q, \xi^{\textsf{dr}}_{q^\pm}, \alpha)$, to obtain the desired many-one reduction to the instance $(\mathcal{D}^\pm_q, q^\pm, \alpha)$ of $\text{SVC}^{\textsf{ms}}_{\mathcal{C}^\pm}$ / $\text{SVC}^{\textsf{dr}}_{\mathcal{C}^\pm}$. ◀

3.2 Computing MS-Shapley Values for Signed Facts

We shall now study the complexity of computing MS-Shapley values for signed facts.

Data Complexity. Regarding data complexity, it was shown in [5] that for every UCQ q, the problem $\text{SVC}^{\textsf{ms}}_q$ is tractable. We first observe that this result immediately lifts to UCQ$^\neg$, by applying the reduction in Lemma 3.8.

▶ **Proposition 3.9.** *$\pm\text{SVC}^{ms}_{UCQ^\neg}$ is in polynomial time in data complexity.*

As an immediate corollary of the preceding proposition and Lemma 3.5, we can further conclude that deciding signed-relevance is tractable. In fact, with a bit more work, we can show that this problem enjoys the lowest possible data complexity (namely, AC_0).

▶ **Proposition 3.10.** *It can be decided in AC_0 data complexity whether a signed fact $\alpha \in \mathcal{D}^\pm$ is signed-relevant w.r.t. a given UCQ$^\neg$ and database $\mathcal{D}$.*

Proof. Consider a UCQ$^\neg$ q. We will write "$p \in q^\pm$" to denote that p is a CQ$^\neg$ disjunct of $q^\pm$. To test whether a given signed fact α is signed-relevant w.r.t. q and the input database, we want to express the following three-part condition

1. There is a signed support (for some disjunct $p \in q^\pm$)...
2. ...that contains α, such that...
3. ...it does not strictly contain a signed support (for some disjunct $p' \in q^\pm$).

by means of a first-order sentence that can be evaluated on $\mathcal{D}$ (as opposed to $\mathcal{D}^\pm$), and which can be constructed from q alone.

Let At_p be the set of all relational atoms in a CQ$^\neg$ $p \in q^\pm$. We will write $\beta = \beta'$ for a signed fact β and a signed atom β' (containing only variables as terms) to denote that the relation name and sign are the same, and moreover $\bigwedge_i x_i = c_i$ holds, where i is the arity of their shared relation and x_i (resp. c_i) is the ith variable (resp. constant) of β' (resp. β). For $\mu : var(q) \to var(q)$ and $p \in q^\pm$, let p_μ denote the Boolean query $\hat{p} \wedge \bigwedge_{x,y \in Im(\mu)} x \neq y$, where $\hat{p}$ is obtained by replacing each variable x with $\mu(x)$ in the disjunct $\check{p}$ of q corresponding to p (note that both $\check{p}$ and $\hat{p}$ may contain negated atoms). For a set At of signed atoms, let $\mu(At)$ denote the result of replacing each variable x in At with $\mu(x)$.

The above condition can then be captured with the following first-order sentence:

$$\underbrace{\bigvee_{p \in q^\pm} \bigvee_{\mu : var(p) \to var(q)} \Big(p_\mu \wedge}_{\text{Item 1}} \underbrace{\bigvee_{\beta \in At_p} \alpha = \mu(\beta)}_{\text{Item 2}} \wedge \neg \underbrace{\bigvee_{p' \in q^\pm} \bigvee_{\mu' : var(p') \to var(q) \text{ s.t. } \mu'(At_{p'}) \subsetneq \mu(At_p)} p'_{\mu'} \Big)}_{\text{Item 3}}$$

Observe that this formula holds in $\mathcal{D}$ *iff* α is signed-relevant *w.r.t.* q and $\mathcal{D}$. Since evaluating first-order formulas is in AC_0 for data complexity, the statement follows. ◀

Combined Complexity. It is known from prior work that the combined complexity of computing MS-Shapley scores for UCQs without negated atoms is #P-hard, even under severe syntactic restrictions (see [6, Proposition 13]). This is why we focus on $\text{CQ}^\neg$s.

Enumerating all minimal signed supports for q in $\mathcal{D}$ as done in Proposition 3.9 is not an option for obtaining tractable combined complexity since the number of supports can be of the order of $O(|\mathcal{D}|^{|q|})$. The recent work [5], rephrased in Theorem 2.1, provides conditions for tractability of counting minimal supports for CQs. Since counting minimal signed supports for a $\text{CQ}^\neg$ q corresponds to counting minimal supports of $q^\pm$ on $\mathcal{D}^\pm$, we can obtain a similar condition for classes $\mathcal{C}$ of $\text{CQ}^\neg$s ensuring tractability of $\pm\text{SVC}^{\mathsf{ms}}_{\mathcal{C}}$ by reusing the same algorithm developed in [5, Theorem 6.6]. The only potential issue is that the database $\mathcal{D}^\pm$ which we need to consider can be substantially larger than $\mathcal{D}$, and for this reason, we need to bound the arity of the negated atoms of q to ensure $\mathcal{D}^\pm$ remains polynomial in size.

We say that a class $\mathcal{C}$ of $\text{CQ}^\neg$ has bounded generalized hypertree width (or bounded self-join width) if the corresponding class of $\text{CQ}^{\neq}$s $\{q^\pm : q \in \mathcal{C}\}$ enjoys this property. We are now ready to state our tractability result:

▶ **Proposition 3.11.** *For every class $\mathcal{C}$ of $\text{CQ}^\neg$ queries having bounded negative arity, bounded generalized hypertree width and bounded self-join width, $\pm\text{SVC}^{\mathsf{ms}}_{\mathcal{C}}$ is in polynomial time in combined complexity, as a corollary of Theorem 2.1 and Lemma 3.8.*

3.3 Computing Drastic-Shapley Values for Signed Facts

We now turn to the problem of computing drastic-Shapley values for signed facts. From the existing literature on drastic-Shapley-based measures for monotone queries, the largest known class of UCQs for which SVC is tractable in data complexity is the class of *safe UCQs* [8, Corollary 3.2]. We refer readers to [7, Definition 4.10] for a formal definition but point out that the safe UCQs have been shown to characterize the UCQs which can be tractably evaluated on probabilistic databases (assuming $\mathsf{FP} \neq \#\mathsf{P}$). The following result shows that this tractability result can be lifted to $\text{UCQ}^\neg$s. In the statement, we shall say that a $\text{UCQ}^\neg$ is *safe* when $q^\pm$ is a safe UCQ, where the "$\neq$" relation is treated like any binary relation.

▶ **Proposition 3.12.** *For every safe $\text{UCQ}^\neg$ query q, $\pm\text{SVC}^{\mathsf{dr}}_q \in \mathsf{FP}$.*

Proof. By Lemma 3.8 we can reduce to $\text{SVC}^{\mathsf{dr}}_{q^\pm}$. In order to compute $\text{Sh}(\mathcal{D}, \xi^{\mathsf{dr}}_{q^\pm}, \alpha)$, it suffices to first materialize $\mathcal{D}_{\neq} = \mathcal{D} \cup \{\neq(c, d) : c, d \in \mathit{const}(\mathcal{D}) \cup \mathit{const}(q), c \neq d\}$, which we can do in polynomial time, and then to compute $\text{Sh}(\mathcal{D}_{\neq}, \xi^{\mathsf{dr}}_{q^\pm}, \alpha)$ using the materialized $\neq$-relation to evaluate inequality atoms in $q^\pm$. As q is assumed to be a safe $\text{UCQ}^\neg$, it follows that $q^\pm$ is a safe UCQ, so the latter task can be performed in FP due to [8, Corollary 3.2]. ◀

While some $\#\mathsf{P}$-hardness results and FP-$\#\mathsf{P}$ dichotomies have been proven for various subclasses of UCQs [14, 3], all of the existing hardness proofs crucially rely on the so-called "exogenous" facts, which we do not consider in this work (*cf.* Remark 2.2). Moreover, even if new hardness results were to be proven for the purely endogenous setting, it is not at all obvious how one would reduce $\text{SVC}^{\mathsf{dr}}_{q^\pm}$ to $\pm\text{SVC}^{\mathsf{dr}}_q$. This is because while the former problem considers arbitrary databases over the signed schema $\Sigma^\pm$, the problem $\pm\text{SVC}^{\mathsf{dr}}_{q^\pm}$ is defined for input databases of the restricted form $\mathcal{D}^\pm$.

4 Responsibility Measures via Positive Supports

The signed facts approach put forth in the previous section may not always be suitable, either because one may prefer to attribute responsibility to more tangible elements which are explicitly present in the database instead of "watering down" the responsibility score among "absent" facts, or simply because the number of negative facts is just too large, especially if one considers schemas with relations of high arity (in particular, note that Proposition 3.11 assumes bounded negative arity). This motivates us to explore how to define notions of relevance and responsibility measures for positive facts.

4.1 Relevance and Responsibility of Positive Facts

One needs to be careful when defining supports based solely on positive facts. As already discussed in the introduction in relation to the example in Figure 1, we should avoid considering $\{I(mp, fish)\}$ as a valid explanation, because the satisfaction of q crucially relies on the absence of $I(mp, meat)$, which is not justified in the context of the database $\mathcal{D}$, which contains the latter fact. With this in mind, in order to attribute responsibility only to (positive) database facts, we shall consider that a set $S \subseteq \mathcal{D}$ is a *positive support* for q in $\mathcal{D}$ if $+S \cup -\mathcal{D}^- \models q^\pm$. Observe how the database context is still represented by $\mathcal{D}^-$: we cannot assume the negation of a fact just because it is not in S, it must be also not in $\mathcal{D}$. We say that a database fact is *positive-relevant* if it appears in some minimal positive support.

These positive supports are "monotone" in the following sense. Let us call a set S of facts a *$\mathcal{D}$-monotone support* for a query q if $S \subseteq \mathcal{D}$ and further $S' \models q$ for every $S \subseteq S' \subseteq \mathcal{D}$.

▶ **Lemma 4.1.** *If S is a positive support of q in $\mathcal{D}$, then S is a $\mathcal{D}$-monotone support.*

However, $\mathcal{D}$-monotone supports do not in general coincide with positive supports, nor do minimal $\mathcal{D}$-monotone supports yield a good notion of relevance, as the next examples show.

▶ **Example 4.2.** Consider the query $q = (\exists x\ A(x) \wedge \neg B(x)) \vee (\exists x\ B(x) \wedge C(x))$ and $\mathcal{D} = \{A(c), B(c), C(c)\}$ from Example 3.3. We have that $\{A(c), C(c)\}$ is a (minimal) $\mathcal{D}$-monotone support since $\{A(c), C(c)\} \models q$ and $\{A(c), B(c), C(c)\} \models q$, but it is not a positive support. As discussed before, $\{A(c), C(c)\}$ should not be deemed a valid explanation for q in the context of $\mathcal{D}$. Thus, we argue that the seemingly simpler notion of considering supports which are monotone does not yield a reasonable notion of explanation. ⌟

Observe that the previous example uses disjunction in a non-trivial way, as $\{A(c)\}$ and $\{A(c), B(c)\}$ satisfy the query due to different disjuncts of q being made true. A natural question is then whether $\mathcal{D}$-monotone supports and positive supports coincide for CQ$^\neg$ queries. The answer, again, is negative.

▶ **Example 4.3.** Consider the CQ$^\neg$ $q = \exists xyzu\ R(x,y,y) \wedge R(y,z,u) \wedge \neg R(u,x,x)$, the database $\mathcal{D} = \{R(a,b,b), R(b,c,d), R(d,a,a)\}$, and the set of facts $S = \{R(a,b,b), R(b,c,d)\}$. First observe that S is a $\mathcal{D}$-monotone support since we have $S \models q$ via the satisfying assignment $\nu = (x,y,z,u) \mapsto (a,b,c,d)$, and $\mathcal{D} \models q$ via $\nu' = (x,y,z,u) \mapsto (d,a,b,b)$. Moreover, S is a subset-minimal $\mathcal{D}$-monotone support, since removing any set of facts from S results in a non-support.

On the other hand, S is not a positive support since the sole satisfying assignment for the positive part of q is ν, sending the negated atom $R(u,x,x)$ to an existing atom $R(d,a,a) \in \mathcal{D}$. Further, the only minimal positive support is $S' = \{R(b,c,d), R(d,a,a)\}$. For this reason, we would obtain that $R(a,b,b)$ is "monotone-relevant" in the sense that it belongs to some minimal $\mathcal{D}$-monotone support while it is not positive-relevant. Analogously as in previous examples, we consider that any notion making $R(a,b,b)$ relevant is not a sensible choice for responsibility attribution, since $R(a,b,b)$ does not effectively participate in any satisfying assignment making the query q true in $\mathcal{D}$. ⌟

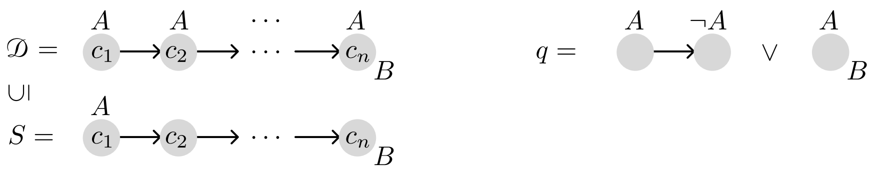

Figure 3 Construction of minimal $\mathcal{D}$-monotone supports of unbounded size.

An additional drawback of the notion of $\mathcal{D}$-monotone supports is that it can lead to minimal supports of unbounded size.

▶ **Example 4.4.** Consider, as depicted in Figure 3, the database $\mathcal{D}_n = \{R(c_i, c_{i+1}) : i < n\} \cup \{A(c_i) : i \leqslant n\} \cup \{B(c_n)\}$ and the UCQ$^\neg$ $q = (\exists xy\ A(x) \land R(x, y) \land \neg A(y)) \ \vee\ (\exists x\ A(x) \land B(x))$, adapted from [2]. We can observe that $S = \{R(c_i, c_{i+1}) : i < n\} \cup \{A(c_1)\} \cup \{B(c_n)\}$ is a minimal $\mathcal{D}$-monotone support. ⌟

Summing up what we have learned from the preceding examples:

▶ **Lemma 4.5.** *Every positive support of q in $\mathcal{D}$ is $\mathcal{D}$-monotone for q, but not all $\mathcal{D}$-monotone supports are positive supports, even when restricted to q being a CQ$^\neg$. Further, $\mathcal{D}$-monotone supports may be of unbounded size w.r.t. the size of the query.*

In light of the preceding examples and results, we shall not study the notion of $\mathcal{D}$-monotone supports further and will focus instead on positive supports for the remainder of the section. Observe that the definition of positive supports has an obvious relation to that of signed support, and it can be seen that every minimal positive support must be part of the positive part of a signed support. However, the notion of relevance over (positive) facts is in general more restrictive than that of signed-relevance, as we show next.

▶ **Lemma 4.6.** *Every positive-relevant fact is signed-relevant as a positive fact for all UCQ$^\neg$s. However, not every signed-relevant positive fact is positive-relevant, even for CQ$^\neg$s.*

Proof sketch. For the second statement, we consider the database $\mathcal{D} = \{R(a, b), R(a, c), B(b)\}$ and the CQ$^\neg$ $q = \exists xyz\ R(x, y) \land R(x, z) \land \neg A(y) \land \neg B(z)$. It is easy to see that $S = \{+R(a, b), +R(a, c), -A(b), -B(c)\}$ is a minimal signed support and thus that $+R(a, b)$ is signed-relevant. However, the only minimal positive support is $S' = \{R(a, c)\}$, meaning that $R(a, b)$ is not positive-relevant. Indeed, observe that $\{R(a, b)\}$ is not a positive support since it would require having $-B(b)$ in $-\mathcal{D}^-$. ◀

However, when a CQ$^\neg$ has no mergeable atoms (which holds, in particular, if it is self-join free), then the two notions coincide.

▶ **Lemma 4.7.** *Let q be a CQ$^\neg$ with no mergeable atoms, $\mathcal{D}$ a database and $S \subseteq \mathcal{D}$. Then, S is a minimal positive support if, and only if, $\hat{S}$ is a minimal signed support for some $\hat{S}$ such that $\hat{S}^+ = S$. Further, if S is a minimal positive support, then there exists exactly one minimal signed support $\hat{S}$ such that $\hat{S}^+ = S$.*

▶ **Corollary 4.8.** *For any CQ$^\neg$ with no mergeable atoms, a fact α is positive-relevant iff $+\alpha$ is signed-relevant.*

Utilizing the notion of positive supports, we may define Shapley-value-based responsibility measures analogously to what has been done for the class of all supports for monotone queries in [5]. Concretely, for a database $\mathcal{D}$ and Boolean query q, we define the *positive-drastic-Shapley* value of a fact $\alpha \in \mathcal{D}$ as being $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{pdr}}, \alpha)$, where for every $S \subseteq \mathcal{D}$, we define $\xi_q^{\mathsf{pdr}}(S) = 1$ if S contains a positive support for q and $\xi_q^{\mathsf{pdr}}(S) = 0$ otherwise. For a class $\mathcal{C}$ of queries, we denote by $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{pdr}}$ the task of given $q \in \mathcal{C}$ a database $\mathcal{D}$ and a fact $\alpha \in \mathcal{D}$ computing $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{pdr}}, \alpha)$.[6] Analogously, the *MPS-Shapley* value of a fact $\alpha \in \mathcal{D}$ is $\mathrm{Sh}(D, \xi_q^{\mathsf{mps}}, \alpha)$ where $\xi_q^{\mathsf{mps}}(B)$ is the number of minimal positive supports of q inside S, and we denote by $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{mps}}$ the task of computing it for a class $\mathcal{C}$ of queries.[7]

4.2 Computing MPS-Shapley Values for Positive Facts

We start our study of the complexity of computing responsibility scores for positive facts by considering the task $\mathrm{SVC}^{\mathsf{mps}}$. First we remark that ξ_q^{mps} can be equivalently defined via a sum of inverse support sizes:

▶ **Lemma 4.9.** *$\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{mps}}, \alpha)$ is equal to the sum, over all minimal positive supports S of q in $\mathcal{D}$ containing α, of $\frac{1}{|S|}$.*

Data Complexity. Regarding data complexity, as was the case for the signed facts approach, we can establish tractable data complexity for all $\mathrm{UCQ}^{\neg}$s.

▶ **Proposition 4.10.** *$\mathrm{SVC}_{UCQ^{\neg}}^{\mathsf{mps}}$ is in polynomial time in data complexity.*

Proof. For every fixed $\mathrm{UCQ}^{\neg}$ query q and database $\mathcal{D}$, observe that: (1) the size of a minimal positive support is bounded by the maximum size N of a $\mathrm{CQ}^{\neg}$ in q, and (2) testing if a set S is a positive support is in polynomial time (indeed, it amounts to evaluating $q^{\pm}$ over the database $+S \cup -\mathcal{D}^{-}$). Hence, given $\alpha \in \mathcal{D}$, we can enumerate, in polynomial time in the size of $\mathcal{D}$, all polynomially-many subsets S of $\mathcal{D}$ of size $\leqslant N$ that contain α and check, for each such subset, whether it is a minimal positive support, adding $\frac{1}{|S|}$ to the current value whenever the check succeeds. The correctness of the result is assured by Lemma 4.9. ◀

The preceding result directly yields tractability of deciding positive-relevance. As in the case of signed-relevance, the upper bound can be improved to AC_0.

▶ **Proposition 4.11.** *It can be decided in AC_0 data complexity whether a fact $\alpha \in \mathcal{D}$ is positive-relevant w.r.t. a given $UCQ^{\neg}$ and database $\mathcal{D}$.*

Combined Complexity. We cannot proceed by reduction to the monotone case as was done in Proposition 3.11 for computing $\mathrm{SVC}^{\mathsf{mps}}$. This is because while the minimal signed supports of q in $\mathcal{D}$ coincide with the minimal supports of $q^{\pm}$ in $\mathcal{D}^{\pm}$, in general there is no bijection between the positive supports of q in $\mathcal{D}$ and the latter supports. Moreover, we need to know the number of positive supports per size, and even in cases where a positive support S is induced from a unique minimal (signed) support S', the sizes of S and S' will differ as soon as S' contains negative facts of the "$-P$" relations. However, when the query has no mergeable atoms, the minimal signed supports are in bijection with the minimal positive supports (Lemma 4.7). We exploit this property to establish the following tractability result:

[6] The "pdr" superscript stands for positive support drastic value.
[7] The "mps" superscript stands for minimal positive supports.

▶ **Proposition 4.12.** *For any class $\mathcal{C}$ of CQ$^{\neg}$s having bounded generalized hypertree width, no mergeable atoms, and bounded negative arity,* $\mathrm{SVC}^{\mathsf{mps}}_{\mathcal{C}}$ *is in polynomial time.*

Proof sketch. Consider a class $\mathcal{C}$ of CQ$^{\neg}$s satisfying the stated conditions, and take some input instance $q, \mathcal{D}, \alpha$. The minimal positive supports are in bijection with the minimal signed supports of the previous section by Lemma 4.7. In particular, we can transform every minimal signed support into a minimal positive support by simply removing the negative facts. Leveraging on this bijection, it suffices to build $\mathcal{D}^{\pm}_{q}$ by Lemma 3.7. We then have that the minimal positive supports of q in $\mathcal{D}$ are in bijection with the minimal supports for $q^{\pm}$ in $\mathcal{D}^{\pm}_{q}$. However, we need to count them for every size and the bijection does not preserve sizes (since it adds/removes negative facts). This issue is fairly minor, but it nevertheless requires delving into the proof of [5, Theorem 6.6], see appendix for details. ◀

We further show that we can drop the restriction to queries of bounded negative arity by instead imposing that negations are "guarded". A negation $\neg R(\bar{x})$ in a CQ$^{\neg}$ q is *guarded* if there exists a positive atom $S(\bar{y})$ of q such that $\bar{y}$ contains all variables of $\bar{x}$. A CQ$^{\neg}$ is guarded if all negations are guarded, and we denote by *CQ*$^{\mathsf{g}\neg}$ the class of all guarded CQ$^{\neg}$s.

▶ **Proposition 4.13.** *For any class $\mathcal{C}$ of CQ$^{\mathsf{g}\neg}$s having bounded generalized hypertree width and no mergeable atoms,* $\mathrm{SVC}^{\mathsf{mps}}_{\mathcal{C}} \in \mathsf{FP}$.

4.3 Computing Drastic-Shapley Values for Positive Facts

The only existing tractability result for computing a drastic-Shapley value for CQ$^{\neg}$s concerns the class of self-join free queries [16], and so we shall also restrict our attention to the class of sjf-CQ$^{\neg}$s. To transfer this tractability result to our setting, we rely on a slight adaptation of the notion of "non-hierarchical paths" introduced in [16, §4.1]. The definition makes use of the *Gaifman graph* $\mathcal{G}(q)$ of a query q, whose set of vertices is *var*(q) and which contains an edge between u and v if some atom of q contains both u and v. We say a CQ$^{\neg}$ q has a *non-hierarchical neg-path*[8] if there are two atoms α_x, α_y whose relations are R_x and R_y and two variables x, y such that: (1) the relations R_x and R_y do not occur in negated atoms, (2) variable x appears in α_x but not α_y, while variable y appears in α_y but not α_x, and (3) the graph obtained from $\mathcal{G}(q)$ by removing every vertex corresponding to a variable in α_x or α_y (excepting x and y) contains a path between x and y.

▶ **Proposition 4.14.** *If a sjf-CQ$^{\neg}$ q has no non-hierarchical neg-path, then* $\mathrm{SVC}^{\mathsf{pdr}}_{q} \in \mathsf{FP}$.

The hardness side of [16, Theorem 4.3] cannot be used to show the hardness of the remaining sjf-CQ$^{\neg}$ with non-hierarchical neg-paths because it heavily relies on exogenous facts which we do not consider here. However, by using relations of negated atoms to "encode" exogenous relations we can sometimes transfer intractability results to our setting:

▶ **Proposition 4.15.** *Consider the CQ$^{\neg}$ $q_{\mathsf{R}\neg\mathsf{ST}} \mathrel{\hat{=}} \exists x, y. R(x) \wedge \neg S(x, y) \wedge T(y)$. Then* $\mathrm{SVC}^{\mathsf{pdr}}_{q_{\mathsf{R}\neg\mathsf{ST}}}$ *is #P-hard.*

5 Related Work on Shapley Values for Queries with Negation

Another notion of relevance that has been put forth is based on the "impact" that the addition of a given fact may have on the satisfaction of the query on a sub-database. More concretely, for a given Boolean query q and a database D, a fact α has a *positive impact*

[8] This definition coincides with that of "non-hierarchical path" from [16] except that in Item (1), we forbid relations in negated atoms, rather than excluding atoms using exogenous relations.

(*resp.* *negative impact*) on $D' \subseteq D$ if we have $D' \not\models q$ and $D' \cup \{\alpha\} \models q$ (*resp.* $D' \models q$ and $D' \cup \{\alpha\} \not\models q$). A fact is then *impact-relevant* – called "relevant" in [16] – if it has some *impact* (positive, negative, or both) on some sub-database.[9]

The article [16] introduces and investigates precisely this notion of relevance, in particular studying the complexity of whether a fact is impact-relevant for a given query. Through this lens, the cited work assigns a positive responsibility score to a fact which has a positive impact, a negative score to those with a negative impact, and null score to facts which have no impact. But it should be noted that facts may have simultaneously positive and negative impacts, in which case some relative weight of the "positivity" and "negativity" must be computed, which in particular may output a null score, even though the fact has an impact and thus is impact-relevant under this view.

Akin to what is done in the case of monotone queries, [16] defines a drastic-Shapley value of a fact α in a database $\mathcal{D}$ for a (non-monotone) query q as being proportional to the number of sub-databases on which α has a positive impact minus the number of sub-databases on which it has a negative impact. More precisely, they consider the value $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \alpha)$ for the drastic wealth function ξ_q^{dr} (defined as $\xi_q^{\mathsf{dr}}(S) = 1$ if $S \models q$ and $\xi_q^{\mathsf{dr}}(S) = 0$ otherwise).[10] Observe that while a non-null score implies impact-relevance, the converse does not always hold (this was already observed by the authors [16, Example 5.3], see also Proposition 5.2). However, if a fact has positive impact but no negative impact, the score will be strictly positive. In the example of Figure 1, the fact $\alpha = I(mp, wine)$ has positive score $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \alpha) > 0$ since: (a) it makes a positive impact on the empty sub-database, and (b) it cannot ever make a negative impact on a sub-database – in other words, $\xi_q^{\mathsf{dr}}(S) = 0$ implies $\xi_q^{\mathsf{dr}}(S \setminus \{\alpha\}) = 0$. The following results show that impact-relevance is orthogonal to the relevance notions we have introduced in our work.

▶ **Lemma 5.1.** *There exist impact-relevant facts that are neither positive-relevant nor signed-relevant for $\mathrm{CQ}^{\neg}$ queries, and facts that are positive-relevant and signed-relevant but not impact-relevant for $\mathrm{UCQ}^{\neg}$ queries.*

▶ **Proposition 5.2.** *There exist a database $\mathcal{D}$, a $\mathrm{UCQ}^{\neg}$ q, a positive-relevant and impact-relevant fact $\alpha \in \mathcal{D}$ and a non-positive-relevant fact $\beta \in \mathcal{D}$ such that $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \alpha) = 0$ and $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \beta) \neq 0$. Further, $+\alpha$ is signed-relevant and $+\beta$ is not.*[11]

Proof sketch. Building on the example of Figure 1, with the query $q_2 = \exists x\ (I(x, meat) \wedge I(x, wine)) \vee (I(x, fish) \wedge \neg I(x, wine))$, we observe that $\alpha = I(mp, wine)$ is positive- and signed-relevant but $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \alpha) = 0$ (since the positive and negative impacts cancel out), and $\beta = I(mm, wine)$ is neither positive- nor signed-relevant yet has $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \beta) < 0$. ◀

In [16, Theorems 3.1 and 4.3] it was shown that computing $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{dr}}, \alpha)$ for self-join free $\mathrm{CQ}^{\neg}$ queries q (without inequalities) is tractable in data complexity when the query is hierarchical, or more generally, does not contain any non-hierarchical path. As shown in Propositions 3.12 and 4.14, analogous tractability results hold for the drastic-Shapley measures introduced in our work. In contrast, it is shown in [16, Proposition 5.5] that testing whether a fact is impact-relevant is generally an NP-complete problem, while we have tractability for testing for signed- and positive-relevance (*cf.* Propositions 3.10 and 4.11).

---

[9] Observe that, for monotone queries, it is equivalent whether a fact is impact-relevant, has positive impact, is part of a minimal support, or is an actual cause in the framework of causal responsibility [15].
[10] We stress that here we are adapting [16]'s definition to our simpler setting without exogenous facts.
[11] We currently do not have an example of a positive-relevant fact which has null-score for a $\mathrm{CQ}^{\neg}$ (instead of a $\mathrm{UCQ}^{\neg}$), and hence we do not know if Proposition 5.2 holds when replacing $\mathrm{UCQ}^{\neg}$ with $\mathrm{CQ}^{\neg}$.

6 Concluding Remarks

Our study on responsibility measures led us to investigate and compare different notions of qualitative explanations (supports) and relevance for queries with negation. We believe that these results are of independent interest and note in particular that both signed-relevance and positive-relevance are easily computable. These notions moreover provide a solid formal underpinning for the novel responsibility measures we introduce, and our complexity study shows how existing tractability results can be smoothly extended to our proposed measures. In particular, the reduction from Lemma 3.8 means that the measures for signed facts can be straightforwardly implemented using algorithms for monotone queries.

While we have focused on developing measures inspired by the MS-Shapley and drastic-Shapley measures, all of the results for signed and positive variants of MS-Shapley – namely Theorem 2.1 and Propositions 3.11, 4.10, 4.12, and 4.13 – can be trivially extended to other WSMS measures based on tractable weight functions [4, §E.1].

One can naturally extend the current framework to handle "exogenous facts" (cf. Remark 2.2), while preserving the established upper bounds and enabling the possibility of transferring existing intractability results. What is less clear is how to extend our approaches to other classes of queries with negation, and we leave the investigation of meaningful notions of relevance, explanation, and responsibility for other sorts of non-monotone queries – such as universal queries – as an interesting but challenging direction for future work. In particular, it would be relevant to explore what are the desirable properties of responsibility measures for non-monotone queries, in line with a recent study for their monotone counterparts [5], which could provide a further axiomatic justification for our proposed measures and their generalizations.

References

1 Omer Abramovich, Daniel Deutch, Nave Frost, Ahmet Kara, and Dan Olteanu. Banzhaf values for facts in query answering. *Proceedings of the ACM on Management of Data (PACMMOD)*, 2(3 (SIGMOD)):123:1–123:26, 2024. doi:10.1145/3654926.

2 Meghyn Bienvenu and Camille Bourgaux. Inconsistency handling in prioritized databases with universal constraints: Complexity analysis and links with active integrity constraints. In *Principles of Knowledge Representation and Reasoning (KR)*, pages 97–106, 2023. doi:10.24963/KR.2023/10.

3 Meghyn Bienvenu, Diego Figueira, and Pierre Lafourcade. When is Shapley value computation a matter of counting? *Proceedings of the ACM on Management of Data (PACMMOD)*, 2(2 (PODS)):105, 2024. doi:10.1145/3651606.

4 Meghyn Bienvenu, Diego Figueira, and Pierre Lafourcade. Responsibility measures for conjunctive queries with negation. *CoRR*, abs/2601.04868, 2025. Full version of current paper. arXiv:2601.04868.

5 Meghyn Bienvenu, Diego Figueira, and Pierre Lafourcade. Shapley revisited: Tractable responsibility measures for query answers. *Proceedings of the ACM on Management of Data (PACMMOD)*, 3(2 (PODS)):112, 2025. doi:10.1145/3725249.

6 Meghyn Bienvenu, Diego Figueira, and Pierre Lafourcade. Tractable responsibility measures for ontology-mediated query answering. In *Principles of Knowledge Representation and Reasoning (KR)*, pages 154–163, 2025. doi:10.24963/KR.2025/16.

7 Nilesh N. Dalvi and Dan Suciu. The dichotomy of probabilistic inference for unions of conjunctive queries. *Journal of the ACM*, 59(6):30:1–30:87, 2012. doi:10.1145/2395116.2395119.

8 Daniel Deutch, Nave Frost, Benny Kimelfeld, and Mikaël Monet. Computing the Shapley value of facts in query answering. In *ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pages 1570–1583. ACM Press, 2022. `doi:10.1145/3514221.3517912`.

9 Auguste Escoffier. *Le guide culinaire : aide-mémoire de cuisine pratique.* Bibliothèque Professionnelle, Paris, 1903.

10 Georg Gottlob, Gianluigi Greco, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: Questions and answers. In *ACM Symposium on Principles of Database Systems (PODS)*, pages 57–74. ACM Press, 2016. `doi:10.1145/2902251.2902309`.

11 Ahmet Kara, Dan Olteanu, and Dan Suciu. From Shapley value to model counting and back. *Proceedings of the ACM on Management of Data (PACMMOD)*, 2(2 (PODS)):79, 2024. `doi:10.1145/3651142`.

12 Pratik Karmakar, Mikaël Monet, Pierre Senellart, and Stephane Bressan. Expected Shapley-like scores of boolean functions: Complexity and applications to probabilistic databases. *Proc. ACM Manag. Data*, 2(2):92:1–92:26, 2024. `doi:10.1145/3651593`.

13 Majd Khalil and Benny Kimelfeld. The complexity of the Shapley value for regular path queries. In *International Conference on Database Theory (ICDT)*, volume 255 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 11:1–11:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. `doi:10.4230/LIPIcs.ICDT.2023.11`.

14 Ester Livshits, Leopoldo E. Bertossi, Benny Kimelfeld, and Moshe Sebag. The Shapley value of tuples in query answering. *Logical Methods in Computer Science (LMCS)*, 17(3), 2021. `doi:10.46298/LMCS-17(3:22)2021`.

15 Alexandra Meliou, Wolfgang Gatterbauer, Katherine F. Moore, and Dan Suciu. The complexity of causality and responsibility for query answers and non-answers. *International Conference on Very Large Data Bases (VLDB)*, 4(1):34–45, 2010. `doi:10.14778/1880172.1880176`.

16 Alon Reshef, Benny Kimelfeld, and Ester Livshits. The impact of negation on the complexity of the Shapley value in conjunctive queries. In *ACM Symposium on Principles of Database Systems (PODS)*, pages 285–297. ACM Press, 2020. `doi:10.1145/3375395.3387664`.

17 Babak Salimi, Leopoldo E. Bertossi, Dan Suciu, and Guy Van den Broeck. Quantifying causal effects on query answering in databases. In *8th USENIX Workshop on the Theory and Practice of Provenance (TaPP)*. USENIX Association, 2016. URL: `https://www.usenix.org/conference/tapp16/workshop-program/presentation/salimi`.

18 Lloyd S. Shapley. *A value for n-person games*, pages 307–318. Princeton University Press, Princeton, 1953. In *Contributions to the Theory of Games*, Volume II. `doi:10.1515/9781400881970-018`.

A Selected Omitted Proofs

The remaining proofs not included here can be found in the long version [4].

▶ **Lemma 3.2.** *For every UCQ*$^\neg$ *q, database $\mathcal{D}$, and set $S \subseteq \mathcal{D}^\pm$, if S is a signed support of q in $\mathcal{D}$, then $\mathcal{D}' \models q$ for every database $\mathcal{D}'$ with $S^+ \subseteq \mathcal{D}'$ and $\mathcal{D}' \cap S^- = \varnothing$ (equivalently, $S^+ \cup \{\neg R(\bar{t}) : R(\bar{t}) \in S^-\} \models^{\mathsf{fin}} q$). However, the converse does not hold even for CQ*$^\neg$*s.*

Proof. If S is a signed support, we have $S \models q^\pm$ via some satisfying assignment $\nu : var(q^\pm) \to const(S)$. Observe that the same mapping ν is a satisfying assignment witnessing $\mathcal{D}' \models q$ as soon as $\mathcal{D}'$ contains all positive facts of S and no negative fact of S.

The fact that the converse implication does not hold means that there exist a CQ$^\neg$ q, database $\mathcal{D}$, and set $S \subseteq \mathcal{D}^\pm$ such that: (1) S is not a signed support and yet (2) we have that $\mathcal{D}' \models q$ for every database $\mathcal{D}'$ with $S^+ \subseteq \mathcal{D}'$ and $\mathcal{D}' \cap S^- = \varnothing$. Such $S, \mathcal{D}, q$ are exhibited in Example 3.3. ◀

▶ **Lemma 3.5.** *For every database $\mathcal{D}$, signed fact $\alpha \in \mathcal{D}^{\pm}$, and UCQ$^{\neg}$ q: α is signed-relevant for q in $\mathcal{D}$ iff $\mathrm{Sh}(\mathcal{D}^{\pm}, \xi^{\mathsf{ms}}_{q^{\pm}}, \alpha) > 0$ iff $\mathrm{Sh}(\mathcal{D}^{\pm}, \xi^{\mathsf{dr}}_{q^{\pm}}, \alpha) > 0$.*

Proof. Observe first that since $q^{\pm}$ is monotone, both scores are defined as sums of non-negative numbers. If S is a minimal signed support of q *w.r.t.* $\mathcal{D}$ containing α, then $\xi^{\mathsf{dr}}_{q^{\pm}}(S) = 1$, $\xi^{\mathsf{ms}}_{q^{\pm}}(S) = 1$, $\xi^{\mathsf{dr}}_{q^{\pm}}(S \setminus \{\alpha\}) = 0$ and $\xi^{\mathsf{ms}}_{q^{\pm}}(S \setminus \{\alpha\}) = 0$, yielding a strictly positive number both for $\mathrm{Sh}(\mathcal{D}^{\pm}, \xi^{\mathsf{ms}}_{q^{\pm}}, \alpha)$ and $\mathrm{Sh}(\mathcal{D}^{\pm}, \xi^{\mathsf{dr}}_{q^{\pm}}, \alpha)$. If, on the other hand, α is in no minimal signed support, then for every set $S \subseteq \mathcal{D}^{\pm}$ we have $\xi^{\mathsf{ms}}_{q^{\pm}}(S) = \xi^{\mathsf{ms}}_{q^{\pm}}(S \setminus \{\alpha\})$ and $\xi^{\mathsf{dr}}_{q^{\pm}}(S) = \xi^{\mathsf{dr}}_{q^{\pm}}(S \setminus \{\alpha\})$. This in turn implies that $\mathrm{Sh}(\mathcal{D}^{\pm}, \xi^{\mathsf{ms}}_{q^{\pm}}, \alpha) = \mathrm{Sh}(\mathcal{D}^{\pm}, \xi^{\mathsf{dr}}_{q^{\pm}}, \alpha) = 0$. ◀

▶ **Lemma 3.7.** *Every minimal signed support of $\mathcal{D}, q$ is in $\mathcal{D}^{\pm}_{q}$.*

Proof. By means of contradiction, suppose a signed support S of q contains some $\alpha = -R(\bar{c})$ for some $R \notin \mathcal{N}_q$. Let ν be a satisfying assignment of $q^{\pm}$ in S. Note that ν is still a satisfying assignment of $q^{\pm}$ in $S \setminus \{\alpha\}$ since $q^{\pm}$ contains no atom over the "$-R$" relation of $\Sigma^{\pm}$. Hence, S cannot be a minimal signed support. ◀

▶ **Proposition 3.9.** *$\pm\mathrm{SVC}^{ms}_{UCQ^{\neg}}$ is in polynomial time in data complexity.*

Proof. Given q, $\mathcal{D}$, α, first observe that since we work in data complexity, the arity of negated atoms in q is fixed. We can then apply Lemma 3.8 to reduce to $\mathrm{SVC}^{\mathsf{ms}}_{\mathrm{UCQ}^{\neq}}$. By [5, Theorem 5.2], and using the facts that ξ^{ms} is tractable and that UCQ$^{\neq}$s are bounded, monotone and tractable, the data complexity of $\mathrm{SVC}^{\mathsf{ms}}_{\mathrm{UCQ}^{\neq}}$ is in polynomial time. ◀

▶ **Proposition 3.11.** *For every class $\mathcal{C}$ of CQ$^{\neg}$ queries having bounded negative arity, bounded generalized hypertree width and bounded self-join width, $\pm\mathrm{SVC}^{ms}_{\mathcal{C}}$ is in polynomial time in combined complexity, as a corollary of Theorem 2.1 and Lemma 3.8.*

Proof. By Lemma 3.8 we can reduce to $\mathrm{SVC}^{\mathsf{ms}}_{\mathcal{C}^{\pm}}$, where $\mathcal{C}^{\pm} = \{q^{\pm} : q \in \mathcal{C}\} \subseteq \mathrm{CQ}^{\neq}$. Since the self-join width and generalized hypertree width of $\mathcal{C}$ and $\mathcal{C}^{\pm}$ coincide by definition, we can conclude by Theorem 2.1. ◀

▶ **Lemma 4.5.** *Every positive support of q in $\mathcal{D}$ is $\mathcal{D}$-monotone for q, but not all $\mathcal{D}$-monotone supports are positive supports, even when restricted to q being a CQ$^{\neg}$. Further, $\mathcal{D}$-monotone supports may be of unbounded size w.r.t. the size of the query.*

Proof. The first statement is Lemma 4.1, the second statement is given by Examples 4.2 and 4.3, and the third statement by Example 4.4.

For the third statement, we provide additional details here. Reconsider the database $\mathcal{D}_n = \{R(c_i, c_{i+1}) : i < n\} \cup \{A(c_i) : i \leqslant n\} \cup \{B(c_n)\}$ and the UCQ$^{\neg}$ $q = (\exists xy\ A(x) \wedge R(x,y) \wedge \neg A(x)) \vee (\exists x\ A(x) \wedge B(x))$, which is depicted in Figure 3. We observe that $S = \{R(c_i, c_{i+1}) : i < n\} \cup \{A(c_1)\} \cup \{B(c_n)\}$ is a minimal $\mathcal{D}$-monotone support. Concretely:

(a) $S \models q$, via the assignment $\{x \mapsto c_1, y \mapsto c_2\}$ which makes the first disjunct hold,

(b) $S' \models q$ for all $S \subseteq S' \subseteq \mathcal{D}$, since there is always a "$A \xrightarrow{R} \neg A$" pattern unless all A-facts are present, in which case $\exists x\ A(x) \wedge B(x)$ holds (and thus so does q), and

(c) S is subset-minimal with respect to (a) and (b), since: (i) removing any set of facts containing $A(c_1)$ leads to a set of facts which does not satisfy q, (ii) removing any set of facts containing $B(c_n)$ leads to a set S' of facts which is not monotone since $S' \cup \{A(c_j) : 1 \leqslant j \leqslant n\} \not\models q$, (iii) removing any set of facts containing $R(c_i, c_{i+1})$ leads to a set S' of facts which is not monotone since $S' \cup \{A(c_j) : 1 \leqslant j \leqslant i\} \not\models q$. ◀

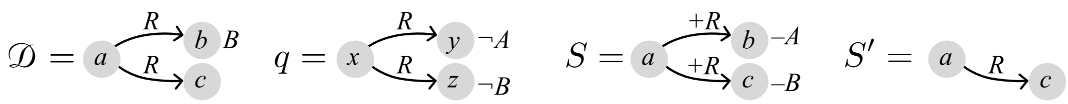

Figure 4 Counterexample for proof of Lemma 4.6.

▶ **Lemma 4.6.** *Every positive-relevant fact is signed-relevant as a positive fact for all UCQ*$^\neg$*s. However, not every signed-relevant positive fact is positive-relevant, even for CQ*$^\neg$*s.*

Proof.

First statement. Consider a database $\mathcal{D}$ and UCQ$^\neg$ q, and let $\alpha \in \mathcal{D}$ be positive-relevant. It follows that there exists a minimal positive support S such that $\alpha \in S$. In particular, $+S \cup -\mathcal{D}^- \models q^\pm$, and hence $+S \cup -\mathcal{D}^-$ is a signed support of q. If $+S \cup -\mathcal{D}^-$ is a minimal signed support, then we are done, as $+S \cup -\mathcal{D}^-$ witnesses that $+\alpha$ is signed-relevant. Otherwise, take any $S' \subseteq S$ and $D' \subseteq \mathcal{D}^-$ such that $+S' \cup -D'$ is a minimal signed support. We must have $S' = S$, otherwise S' would also be a positive support, contradicting the minimality of S. Hence, $+S \cup -D'$ is a minimal signed support that contains $+\alpha$, so $+\alpha$ is signed-relevant.

Second statement. Consider the database $\mathcal{D} = \{R(a,b), R(a,c), B(b)\}$ and the CQ$^\neg$ $q = \exists xyz\ R(x,y) \wedge R(x,z) \wedge \neg A(y) \wedge \neg B(z)$, see Figure 4 for a depiction. It is easy to see that $S = \{+R(a,b), +R(a,c), -A(b), -B(c)\}$ is a minimal signed support and thus that $+R(a,b)$ is signed-relevant. However, the only minimal positive support is $S' = \{R(a,c)\}$, meaning that $R(a,b)$ is not positive-relevant. Indeed, observe that $\{R(a,b)\}$ is not a positive support since it would require having $-B(b)$ in $-\mathcal{D}^-$. ◀

▶ **Lemma 4.7.** *Let q be a CQ$^\neg$ with no mergeable atoms, $\mathcal{D}$ a database and $S \subseteq \mathcal{D}$. Then, S is a minimal positive support if, and only if, $\hat{S}$ is a minimal signed support for some $\hat{S}$ such that $\hat{S}^+ = S$. Further, if S is a minimal positive support, then there exists exactly one minimal signed support $\hat{S}$ such that $\hat{S}^+ = S$.*

Proof. First observe that:

1. If $\hat{S} \models q^\pm$, then the size of $\hat{S}^+$ must be, at least, the number k of positive atoms of q, since otherwise we would have two mergeable atoms in q.
2. If $\hat{S}$ is a minimal signed support, then there exists exactly one homomorphism $q^\pm \xrightarrow{\text{hom}} \hat{S}$. Indeed, there is only one homomorphism from the positive atoms to the positive facts since otherwise there would be two mergeable atoms; and the variable assignment of the unique homomorphism for the positive atoms also fixes where the variables in the negative atoms are mapped since all negations in q are safe.

First statement. The left-to-right direction is shown in the proof of the first statement of Lemma 4.6. For the right-to-left direction, suppose now that $\hat{S} \subseteq \mathcal{D}^\pm$ is a minimal signed support – in particular $\hat{S} \models q^\pm$. Note that by minimality the size of $\hat{S}^+$ must be exactly k by Item 1. By monotonicity of $q^\pm$ (it is a CQ$^{\neq}$), we have $+\hat{S}^+ \cup -\mathcal{D}^- \models q^\pm$ and thus $\hat{S}^+$ is a positive support. If $\hat{S}^+$ was not minimal, we would have $+S' \cup -\mathcal{D}^- \models q^\pm$ for some $S' \subsetneq \hat{S}^+$, contradicting that the number of positive atoms of $+S' \cup -\mathcal{D}^-$ is at least k.

Second statement. For any given minimal signed support $\hat{S}$, there is only one homomorphism from the positive atoms of $q^\pm$ to $+\hat{S}^+$, as explained in Item 2, and this homomorphism fixes the negative facts of $\hat{S}$. Hence, there cannot be two minimal signed supports sharing the same positive atoms. ◀

▶ **Lemma 4.9.** $\mathrm{Sh}(\mathcal{D}, \xi_q^{\mathsf{mps}}, \alpha)$ *is equal to the sum, over all minimal positive supports S of q in $\mathcal{D}$ containing α, of $\frac{1}{|S|}$.*

Proof. This proof is analogous to that of [5, Proposition 4.3]. Consider some minimal positive support S, and let q_S be the query whose only minimal positive support is S. By the so-called "Shapley axioms", which $\mathrm{Sh}(\mathcal{D}, \xi_{q_S}^{\mathsf{mps}}, \alpha)$ verifies by definition [18], we have by axiom (Null) that $\mathrm{Sh}(\mathcal{D}, \xi_{q_S}^{\mathsf{mps}}, \alpha) = 0$ for any fact $\alpha \in \mathcal{D} \setminus S$. By axiom (Sym), the remaining values must all have the same value, and their sum is fixed to be $\xi_{q_S}^{\mathsf{mps}}(\mathcal{D})$, that is, the number of minimal positive supports in $\mathcal{D}$, which is 1. This means that $\mathrm{Sh}(\mathcal{D}, \xi_{q_S}^{\mathsf{mps}}, \alpha) = \frac{1}{|S|}$ if $\alpha \in S$ or 0 otherwise. Finally, by axiom (Lin) we can sum the contributions of all positive minimal supports to obtain the desired formula. ◀

▶ **Proposition 4.13.** *For any class $\mathcal{C}$ of $CQ^{g\neg}$s having bounded generalized hypertree width and no mergeable atoms, $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{mps}} \in \mathsf{FP}$.*

Proof. Consider such a class $\mathcal{C}$, and an input instance $q, \mathcal{D}, \alpha$. Let us divide q into its non-negated and negated atoms $q = q_+ \wedge q_-$, where we put inequality atoms in q_-. Remember that, by guardedness, for every negated atom $\neg\gamma$ of q_- there is a positive atom $\check{\gamma}$ of q_+ such that $var(\gamma) \subseteq var(\check{\gamma})$. Further, by non-mergeability, for every fact β of $\mathcal{D}$, there is at most one atom $\hat{\beta}$ of q_+ such that $\hat{\beta} \xrightarrow{\text{hom}} \beta$. Let $\mathcal{D}' \subseteq \mathcal{D}$ be the set of all facts $\beta \in \mathcal{D}$ such that $\hat{\beta} \xrightarrow{\text{hom}} \beta$ for some $\hat{\beta} \in q_+$, and observe that any minimal positive support must be inside $\mathcal{D}'$. Note that $\mathcal{D}'$ can be computed in polynomial time since it amounts to testing whether a fact is in the result of an atomic CQ. Let $\mathcal{D}'' \subseteq \mathcal{D}'$ be the result of further removing any fact β from $\mathcal{D}'$ if it is not part of the evaluation of $q_\beta = \hat{\beta} \wedge \bigwedge_{\check{\gamma}=\hat{\beta}} \neg\gamma$ on $\mathcal{D}$. Notice that $\mathcal{D}''$ can be computed in polynomial time since each q_β is polynomial-time tractable.We can further show that a set $S \subseteq \mathcal{D}$ is a minimal positive support of q in $\mathcal{D}$ if, and only if, it is a minimal support of q_+ on $\mathcal{D}''$ [4, Claim C.1]. Hence, there is a polynomial-time reduction from $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{mps}}$ to $\mathrm{SVC}_{\mathcal{C}_+}^{\mathsf{ms}}$, for $\mathcal{C}_+ = \{q_+ : q \in \mathcal{C}\}$. Since the queries of $\mathcal{C}_+$ have no mergeable atoms and the same generalized hypertree width bound as $\mathcal{C}$, we can apply Theorem 2.1 to conclude that $\mathrm{SVC}_{\mathcal{C}_+}^{\mathsf{ms}}$, and thus $\mathrm{SVC}_{\mathcal{C}}^{\mathsf{mps}}$, are in polynomial time. ◀

▶ **Lemma 5.1.** *There exist impact-relevant facts that are neither positive-relevant nor signed-relevant for $CQ^\neg$ queries, and facts that are positive-relevant and signed-relevant but not impact-relevant for $UCQ^\neg$ queries.*

Proof.

First statement. In the example of Figure 1, the fact $I(mp, wine)$ is impact-relevant but not signed-relevant nor positive-relevant.

Second statement. Consider the $\mathrm{UCQ}^\neg$ $q = (\exists x\ A(x) \wedge B(x)) \vee (\exists x\ \neg A(x) \wedge B(x))$ and the database $\mathcal{D} = \{A(c), B(c)\}$. It is easy to see that the fact $A(c)$ is not impact-relevant. However, $+A(c)$ is signed-relevant since it belongs to the minimal signed support $\{+A(c), +B(c)\}$, and $A(c)$ is positive-relevant since it belongs to the minimal positive support $\{A(c), B(c)\}$. ◀