Determinacy of Real Conjunctive Queries. The Boolean Case.

Jarosław Kwiecień
University of Wrocław

Jerzy Marcinkowski
University of Wrocław

Piotr Ostropolski-Nalewaja
University of Wrocław

ABSTRACT

In their classical 1993 paper [1] Chaudhuri and Vardi notice that some fundamental database theory results and techniques fail to survive when we try to see query answers as bags (multisets) of tuples rather than as sets of tuples.

But disappointingly, almost 30 years after [1], the bag-semantics based database theory is still in the infancy. We do not even know whether conjunctive query containment is decidable. And this is not due to lack of interest, but because, in the multiset world, everything suddenly gets discouragingly complicated.

In this paper we try to re-examine, in the bag semantics scenario, the query determinacy problem, which has recently been intensively studied in the set semantics scenario. We show that query determinacy (under bag semantics) is decidable for boolean conjunctive queries and undecidable for unions of such queries (in contrast to the set semantics scenario, where the UCQ case remains decidable even for unary queries). We also show that – surprisingly – for path queries determinacy under bag semantics coincides with determinacy under set semantics (and thus it is decidable).

1 INTRODUCTION

1.1 The context

This paper is about the query determinacy problem. So let us maybe start with a definition:

DEFINITION 1. • *For a query q and a finite set of queries V, we say that V determines q (denoted as $V \to q$) if the implication:*

$$(\forall v \in V \quad v(D) = v(D')) \implies q(D) = q(D') \qquad (\spadesuit)$$

holds for every pair D, D' of finite[1] structures[2].
• An instance of the determinacy problem, for a query language $\mathcal{L}$, consists of a query $q \in \mathcal{L}$ and a finite set of views $V \subseteq \mathcal{L}$. We ask whether $V \to q$.

Many different variants of the determinacy problem, for various query languages, and (when applicable) various arities of queries, have been studied in the last three decades. And the point has been reached, where we have a pretty complete classification of the variants, in the sense that we know which of them are decidable (few) and which are not (most).

So, for example, as observed in [2], the problem is decidable if the queries in V are unary UCQs[3] (unions of conjunctive queries) and q is any UCQ. Let us outline how one can prove this:

As noticed in he paper [4], $V \to q$ holds if and only if:
$$(*) \quad Chase(TGD(V), green(q)) \models red(q)$$
where $green(q)$ and $red(q)$ are some structures that can easily be

constructed from q and $TGD(V)$ is some set of Tuple Generating Dependencies which can easily be constructed from V, and where $Chase(TGD(V), green(q))$ is a result of applying the TGDs from $TGD(V)$ to $green(q)$ until the fixpoint is reached. Then, it is easy to see that if the queries from V are unary then the TGDs in $TGD(V)$ are frontier one. And query entailment[4] is decidable for sets of such TGDs [5]. Then, if one is unhappy with the fact that $Chase(TGD(V), green(q))$ is potentially infinite, leading to infinite D and D', the finite controlabillity result for frontier-one TGDs (implied by [6]) can be used to replace $Chase(TGD(V), green(q))$ with a finite structure with the desired properties.

We do not really want our readers to understand this complicated reasoning (unless they already do). We only outline it in order to show that database theory has reached the point where it is no longer merely a set[5] of results about the fundamental notions and phenomena, but a real scientific theory, able to explain and interpret facts which are apparently totally unrelated: we do not believe that the authors of [5] and [6] ever expected their results to be used in a decidability proof of a variant of the determinacy problem.

Unfortunately, this beautiful palace of database theory, both the results and the tools, collapses like a house of cards when we try to be slightly more realistic and assume that the queries do not return **sets** of tuples, but they return **multisets** (or *bags*) of tuples.

And this is not a new observation. It was already spotted in [1], where the authors try to see what happens to the most important database theory fundamental, query containment, if bag semantics is assumed, concluding that the *"techniques from the set-theoretic setting do not carry over to the bag-theoretic setting"*.

The paper [1] was understood, at least by part of the community, as a call *"for a* **re-examination** *of the foundations of databases where the fundamental concepts and algorithmic problems are investigated under bag semantics, instead of set semantics"* (see [7], page 2). But only rather limited progress has been achieved. Even the decidability of conjunctive query containment problem remains open in the multiset world. And this is in spite of a considerable effort, which is reflected by a list of publications.

First, in 1995 [8] show that containment of UCQs, which is in NP when the classical (set) semantics is considered, becomes undecidable for the multiset semantics. Then (among other papers) there are [9] where it is shown that containment is undecidable if inequalities are allowed in conjunctive queries and [10] which shows decidability (and establishes complexity) for several simple subcases. And then, finally, there is a paper [11], where query containment is in an elegant way related to the information-theoretic notion of entropy, and it is shown that decidability of even a quite limited subproblem of query containment would imply a solution to a long standing open problem in information theory.

---

[1]Both "finite" and "unrestricted" versions of this problem were considered, but in this paper let us concentrate on the finite one, which is the only one to make sense in the multiset scenario.
[2]Where $v(D)$ is the result of applying v to D.
[3]"Unary" means that they have one free variable. Similar result for unary conjunctive queries was proven in [3].

[4]That is, condition (*) above.
[5]Or maybe "multiset" would be a better term in this context, as some of the results were produced more than once.

Apart from the line of research focused on the query containment problem, the number of such *re-examination* attempts, while growing, remains low. And this is – we understand – not because of lack of interest, but because (as the containment problem illustrates) everything suddenly gets very complicated when multiset semantics is assumed. One example we know about is the recent paper [7] where the authors re-examine the old result from [12], that a database schema is acyclic if and only if the local-to-global consistency property for relations over that schema holds.

1.2 Our contribution (and the future work).

In this paper we attempt *a re-examination*, under multiset semantics, of the query determinacy problem.

This means that we now read the equalities in formula ♠ as equalities of multisets. To distinguish we will use the symbol $\xrightarrow{\text{set}}$ to denote the old style set-semantics determinacy and $\xrightarrow{\text{bag}}$ for determinacy under multi-set semantics.

The first question one naturally needs to ask here is whether $\xrightarrow{\text{bag}}$ is really a different notion than $\xrightarrow{\text{set}}$. And, if they are indeed different, the second question is: does at least one implication hold? Like in the case of query containment, where, as noticed already in [1], containment under multiset semantics is a strictly stronger property than containment under set semantics?

To show that the two versions are really different let us use:

EXAMPLE 2. *Let q be the query* $\exists u, y, z\ P(u, \mathbf{x}), R(\mathbf{x}, y), S(y, z)$ *and let V consist of two conjunctive queries:*

$$\exists u, y\ P(u, \mathbf{x}), R(\mathbf{x}, y) \qquad \exists y, z\ R(\mathbf{x}, y), S(y, z)$$

Then is is easy to see that $V \xrightarrow{\text{set}} q$ *but* $V \xrightarrow{\text{bag}} \!\!\!\!\!/\ \ q$.

Regarding the second question notice that while equality of $\psi(D)$ and $\psi(D')$ (for some query ψ) under multiset semantics implies that they are also equal under set semantics, the formula ♠ has both positive and negative occurrence of equality of the answer sets. So it is not obvious at all that multiset determinacy always implies determinacy in the set semantics world. And indeed:

EXAMPLE 3. *Let q be the query* $\exists x\ R(x)$ *and let V consist of two queries:*

$$v_1 = \exists x\ P(x) \qquad v_2 = \exists x\ P(x) \lor \exists x\ R(x)$$

Then it is easy to see that $V \xrightarrow{\text{set}} \!\!\!\!\!/\ \ q$. *But under the multiset semantics for each D we have* $q(D) = v_2(D) - v_1(D)$ *(since we consider boolean queries here, the answers are natural numbers), which implies that* $V \xrightarrow{\text{bag}} q$.

Can such example be constructed for conjunctive queries rather than UCQs? We do not know. We conjecture that the answer is "no", but proving it will probably be hard. What we can show (and we find it a bit surprising, because the situations where set-semantics based notions coincide with their multiset-semantics counterparts seem to be rare) is:

THEOREM 1. *If V is a set of path queries, and q is a path query[6] then* $V \xrightarrow{\text{set}} q$ *if and only if* $V \xrightarrow{\text{bag}} q$.

Determinacy of path queries (under the set semantics) is one of the few decidable cases [13], and, as Theorem 1 implies, it remains decidable in the multiset semantics world. For the proof of Theorem 1 see Section 3. Notice also that the queries from Example 2 are not far from being path queries, but still, for some reason, the thesis of Theorem 1 does not hold for them.

But the main focus of this paper is on understanding query determinacy in the case of boolean queries. We first present:

THEOREM 2. *The problem whether, for a set V of boolean UCQs and for another boolean UCQ q, it is true that* $V \xrightarrow{\text{bag}} q$, *is undecidable.*

This is in stark contrast to the situation in the set semantics world where, as we already mentioned in Section 1.1, determinacy is decidable even for unary UCQs, not just boolean. But the proof of Theorem 2 is not hard. In order to show it, it was enough to notice that the "$p_1 \lor p_2$ trick" from [14] (or the "cold-hot" trick from [2]) can be safely used in the multiset semantics world. And then to reuse the Hilbert 10th problem encoding from [8].

Finally, our main technical contribution is:

THEOREM 3. *The problem whether, for a set V of boolean CQs and for another boolean CQ q, it is true that* $V \xrightarrow{\text{bag}} q$, *is decidable.*

The proof of Theorem 3 is presented in Sections 4-7. A corollary from the proof of Theorem 3 is that for boolean conjunctive queries $\xrightarrow{\text{bag}}$ is a strictly stronger property than $\xrightarrow{\text{set}}$.

Future work. The natural open question we leave is the decidability status of the CQ determinacy for the multiset semantics, that is of the problem whether, for a set V of CQs (with free variables) and for another CQ q, it is true that $V \xrightarrow{\text{bag}} q$. The encoding method from the proof of Theorem 2 is useless when disjunction is no longer available. And also the techniques from the proof of Theorem 3 do not seem to generalize to the scenario with free variables.

1.3 The tools. And related works.

Regarding **the tools** used in the proofs of Theorem 1 and Theorem 3, let us quote [1] again: *techniques from the set-theoretic setting do not carry over to the bag-theoretic setting*. The green-red chase (mentioned in Section 1.1), which is a fundamental tool to study determinacy in the set-semantics world, just vanishes in the multiset setting, together with the results that depend on it, like undecidability for the CQ case. And in general, the importance of concepts that stem from the first order logic diminishes in this new world. Instead, tools based on notions from linear algebra arise in a very natural way here. This is not at all a new observation: in order to read [7] one also needs to dust off the linear algebra textbook.

While we are (as far as we know) the first to consider query determinacy under multiset semantics, there exists a line of research in database theory which concentrates on the number of answers to a query (homomorphisms), including paper [15] (again in a natural way using arguments from linear algebra). And also, due to solely mathematical motivations, such homomorphism counts (and some related numbers) were studied by researchers in combinatorics, with numerous papers published, including [16] and [17], and a book[7] [18]. Some of the results regarding homomorphism counts

---

[6]For a definition of path queries see Section 3

[7]We only had access to a free version of this book available on the web.

are useful for us (see Section 6 where we use the main result from [16]). Some, while not directly useful, are related to our paper, for example there is a construction in [15] resembling Step 1 (and partially also Step 2) from our construction in Section 6.

The title of [17] may suggest that there is a connection to determinacy and (as we learned) some less careful readers can have an impression that the main result from [17] is almost our Theorem 3. So let us take some space here to explain why this is not the case[8].

A set of connected non-isomorphic graphs $\mathcal{H} = \{H_1, \ldots H_m\}$ is consideerd in [17]. For $H \in \mathcal{H}$ and for another graph G the number $t(H, G)$ („homomorphism density") is defined as the probability that a random mapping from the set of vertices of H to the set of vertices of G will be a homomorphism.

Let now S be the set $\{\langle t(H_1, G), \ldots, t(H_m, G)\rangle : G \text{ is a graph}\}$, which clearly is a subset of $[0, 1]^m$ or, to be more precise, of $(\mathbb{Q} \cap [0, 1])^m$. The main theorem of [17] (Theorem 1 there) says that:

(*) S contains a subset B which is dense in some ball.

Then, it seems to be claimed[9] in [18] that it follows from (*) that no functional dependence between the numbers $t(H_1, G), \ldots, t(H_m, G)$ can exist, meaning that $t(H_m, G)$ cannot be a function of arguments $t(H_1, G), \ldots, t(H_{m-1}, G)$. In our language this would mean that:

(**) $t(H_1, G), \ldots, t(H_{m-1}, G)$ do not determine $t(H_m, G)$.

If this was indeed true that (*) implied (**) then one could use the graph blow-up technique from [18] (Theorem 5.32) to translate the language of „homomorphism densities" into the language of homomorphism counts and, as a result, prove our Corollary 33, which is a very special case of our Theorem 3.

But (*) **does not** imply (**). Let us define C as the projection of S on the first m-1 coordinates, that is $C = \{\langle q_1, \ldots, q_{m-1}\rangle : \exists q_m \langle q_1, ..., q_{m-1}, q_m\rangle \in S\}$.

Then (**) means that S cannot be the graph of some function $f : C \to \mathbb{Q}$. But all (*) tells us about S is that its topological closure contains a ball. And it is easy to construct such a function $f : ([0, 1] \cap \mathbb{Q})^{m-1} \to \mathbb{Q} \cap [0, 1]$ that the topological closure of the graph of f not only contains a ball but is actually the entire cube $[0, 1]^m$.

What does indeed follow from (*) is that no such **continuous** function f can exist, so in particular $t(H_m, G)$ cannot be expressed from $t(H_1, G), \ldots, t(H_{m-1}, G)$ by operations which preserve continuousness. But then it is a completely different story, as continuousness may make sense when talking about homomorphisms density, but not in the context of homomorphism count.

2 PRELIMINARIES

2.1 Database Theory Notions

Multisets. A multiset X is a mapping $Y \to \mathbb{N}$ where Y is some specified set[10]. With $X[a]$ we will denote the number of occurrences of a in X. We write that $X[a] = 0$ if $a \notin Y$. A union $X \cup X'$ of two multisets X and X' is a multiset such that $(X \cup X')[a] = X[a] + X'[a]$. We define other multiset operators analogously.

Structures. A *schema* Σ is a finite set of relational symbols. A schema Σ is *n-ary* if an arity of its relations is at most n. A *structure (or database)* D over schema Σ is a finite set[11] consisting of facts. A *fact* is simply an atom $A(\vec{t})$ where $\vec{t}$ is a tuple of *terms* from some fixed infinite set of constants. The *active domain* of D (denoted with $dom(D)$) is the set of constants that appear in facts of D.

Homomorphisms. For two structures D and D' over schema Σ, a homomorphism from D to D' is a function $h : dom(D) \to dom(D')$ such that for each atom $A(\vec{t}) \in D$ it holds that $A(h(\vec{t})) \in D'$. A set of homomorphisms from D to D' is denoted with $hom(D, D')$. Note, that $|hom(\emptyset, D)| = 1$ for the empty structure $\emptyset$.

Conjunctive Queries (CQs). A *conjunctive query* $\Phi = \exists \vec{y} \; \phi(\vec{x}, \vec{y})$ is a first order formula such that $\phi(\vec{x}, \vec{y})$ is a conjunction of atoms over variables from $\vec{x}$ and $\vec{y}$. With $vars(\Phi)$ we will denote the set of variables of $\phi(\vec{x}, \vec{y})$. The *arity* of CQ Φ is simply $|\vec{x}|$.

The *frozen body* of a CQ $\Phi = \exists \vec{y} \; \phi(\vec{x}, \vec{y})$ is a structure obtained from $\phi(\vec{x}, \vec{y})$ by bijective replacement of variables with fresh constants. For a CQ $\Phi = \exists \vec{y} \; \phi(\vec{x}, \vec{y})$ and a structure D, with $hom(\Phi, D)$ we denote the set of all homomorphisms from the frozen body of Φ to D. A *result* $\Phi(D)$ of a CQ Φ over a structure D is a multiset such that $\Phi(D)[\vec{a}] = |\{h \in hom(\Phi, D) : \vec{a} = h(\vec{x})\}|$.

Path Queries. For a binary schema Σ a *path query* Λ is a CQ of the form $\exists x_1, \ldots, x_{n-1} \; R_1(x, x_1), R_2(x_1, x_2), \ldots, R_n(x_{n-1}, y)$.

Let Σ^* denote the set of all words over relational symbols from Σ. Given the nature of path queries we will identify them with words from Σ^*, so instead of writing $\Lambda = \exists x_1, x_2 A(x, x_1), B(x_1, x_2), C(x_2, y)$ we may conveniently write[12] $\Lambda = ABC$.

Boolean Queries. A CQ q with no free variables is called *boolean*. Boolean CQs will be always identified with their frozen bodies.

Accordingly to previous definitions a result $q(D)$ of a boolean CQ q over some structure D is a multiset containing $|hom(q, D)|$ copies of the empty tuple. For brevity we write $q(D)$ instead of $q(D)[\langle\rangle]$, so $q(D) = |hom(q, D)|$.

A *union of boolean conjunctive queries (boolean UCQ)* Ψ is a disjunction of a finite number of boolean CQs. A *result* $\Psi(D)$ of a boolean UCQ Ψ over a D is the natural number $\sum_{\Phi \in \Psi} \Phi(D)$.

A boolean CQ q is **contained under set semantics** in a boolean CQ q' (denoted as $q \subseteq_{\text{set}} q'$) if for every structure D it holds that $q(D) > 0 \Rightarrow q'(D) > 0$. It is well-known that $q \subseteq_{\text{set}} q'$ if and only if $hom(q', q)$ is non-empty.

2.2 Graph Theoretic Tools

Operations on Structures. Following [16] we will use some operations on structures. For structures A and B over schema Σ:
• $A + B$ is a disjoint union[13] of A and B;
• $A \times B$ is a structure such that $dom(A \times B) = dom(A) \times dom(B)$ and for any $R \in \Sigma$ the following holds: $R(\langle a_1, b_1\rangle, \ldots, \langle a_k, b_k\rangle)$ is an atom of $A \times B$ if and only if $R(a_1, \ldots, a_k) \in A$ and $R(b_1, \ldots, b_k) \in B$;

---

[8]It may be a good idea to skip the rest of this Section now and come back here after you read Section 4 at the earliest.
[9]Remarks after Corollary 5.45 in [18]; unfortunately the language is quite sloppy there, and it is not entitely clear for us how this part of text should correctly be interpreted.
[10]We (of course) think that $0 \in \mathbb{N}$.

[11]Which means that we assume that answers to the queries are multisets, but the structures are sets. However, all our results and techniques would survive if we defined structures which are multisets of facts.
[12]Note however, that an empty word ε is identified with the query $\Lambda(x, y) = "x = y"$, although it is not a valid path query.
[13]That is if $dom(A) \cap dom(B) \neq \emptyset$ we bijectively rename variables of B with fresh ones and then make $A + B = A \cup B$

• we use symbols $\sum$ and $\prod$ as generalized $+$ and $\times$ in the usual way;
• for $t \in \mathbb{N}_+$, $tA = \sum_{i=1}^{t} A$ and $A^t = \prod_{i=1}^{t} A$. Furthermore, $0A$ is an empty structure and A^0 is a singleton $\{\alpha\}$ such that for any $R \in \Sigma$ $R(\alpha, \alpha, \dots, \alpha) \in A^0$ (α has loops of all types).

Graph Theoretic Lemma. From [16] we recall:

LEMMA 4. *Let A, B, C be structures and $t \in \mathbb{N}$, then:*

(1) If A is connected, then $|hom(A, B+C)| = |hom(A, B)| + |hom(A, C)|$
(2) If A is connected, then $|hom(A, tB)| = t \cdot |hom(A, B)|$
(3) $|hom(A, B \times C)| = |hom(A, B)| \cdot |hom(A, C)|$
(4) $|hom(A, B^t)| = |hom(A, B)|^t$
(5) $|hom(A + B, C)| = |hom(A, C)| \cdot |hom(B, C)|$

2.3 Basic Mathematical Tools and Notations

We are going use standard notation from linear algebra, which should be clear in most cases. Below we describe all the conventions that might be non-obvious:
• For a set $A \subseteq \mathbb{R}^k$, span(A) means the linear span of A (i.e. the smallest linear space containing A). For a set $B \subseteq \mathbb{R}$, we define $\text{span}^B(A) = \{b_1\vec{a_1} + \dots + b_n\vec{a_n} \mid n \in \mathbb{N}; \vec{a_1}, \dots, \vec{a_n} \in A; b_1, \dots, b_n \in B\}$.
• For two vectors $\vec{u}, \vec{u'} \in \mathbb{R}^k$, $\langle \vec{u}, \vec{u'} \rangle$ denotes the dot product of $\vec{u}, \vec{u'}$. Vector $\vec{u}$ is *orthogonal* to $\vec{u'}$ if and only if $\langle \vec{u}, \vec{u'} \rangle = 0$.
• For a vector $\vec{u} \in \mathbb{R}^k, i \in \{1, \dots, k\}$, $u(i)$ denotes the value of the i-th coordinate of $\vec{u}$.
• For a matrix $M \in \mathbb{R}^{k \times k}, i, j \in \{1, \dots, k\}$, $M(i, j)$ denotes the value of the element in the i-th row and j-th column of M.
• For a matrix $M \in \mathbb{R}^{k \times k}$ and a set $A \subseteq \mathbb{R}^k$, $M(A) = \{M\vec{x} \mid \vec{x} \in A\}$.

We will use the following well-known mathematical facts:

FACT 5. *Let $\vec{u}_1, \dots, \vec{u}_n, \vec{u} \in \mathbb{Q}^k$ such that $\vec{u} \notin \text{span}\{\vec{u}_1, \dots, \vec{u}_n\}$. Then there is a vector $\vec{z} \in \mathbb{Q}^k$ such that $\vec{z}$ is orthogonal to $\vec{u}_1, \dots, \vec{u}_n$ but is not orthogonal to $\vec{u}$.*

FACT 6. *If matrix $M \in \mathbb{R}^{k \times k}$ is nonsingular, then the mapping $\vec{x} \mapsto M\vec{x}$ is a homeomorphism (a continuous bijection whose inverse function is continuous too).*

FACT 7. *$\mathbb{Q}^k$ is a dense subset of $\mathbb{R}^k$, i. e., for any $\vec{x} \in \mathbb{R}^k$ and $r > 0$ there is $\vec{y} \in \mathbb{Q}^k$ such that $\|\vec{x} - \vec{y}\| < r$.*

COROLLARY 8. *Suppose $M \in \mathbb{R}^{k \times k}$ is nonsingular. Then there is $\vec{p} \in M(\mathbb{R}_{\geq 0}^k) \cap \mathbb{Q}^k$ such that*

$$\exists r > 0 \; \forall \vec{x} \in \mathbb{R}^k \;\; \|\vec{x} - \vec{p}\| < r \Rightarrow \vec{x} \in M(\mathbb{R}_{\geq 0}^k) \qquad (\star)$$

PROOF. From Fact 6 we know that the set $M(\mathbb{R}_{\geq 0}^k)$ has non-empty interior (i. e. the set of points satisfying $\star$), since it is a homeomorphic image of a set with non-empty interior. By Fact 7 we get that this interior must contain a point with rational coordinates. □

Important Notational Convention. 0^0 equals 1 in this paper.

3 THE PATH QUERIES CASE

In this section we prove:

THEOREM 1. *If V is a set of path queries, and q is a path query, then $V \xrightarrow{set} q$ if and only if $V \xrightarrow{bag} q$.*

One can find this theorem a bit surprising. Path queries are a reasonably wide class of queries. And we have already learned that one should not expect a set-semantics based notion to agree with its multi-set based counterpart on a wide class of objects[14]. But, as it turns out, both versions of determinacy for path queries enjoy the same elegant combinatorial characterisation:

DEFINITION 9. *For a set V of path queries and for another path query q we define an undirected graph $G_{q,V}$ as follows:*

• *$dom(G_{q,V}) = \{w \in \Sigma^* \mid w$ is a prefix of $q\}$. In particular, the empty word ε and q itself[15] are elements of $dom(G_{q,V})$.*
• *There is an edge between w and w' if and only if $w' = wv$ for some $v \in V$.*

The following fact is well known [2, 13]:

FACT 10. *$V \xrightarrow{set} q$ iff there is a path in $G_{q,V}$ from ε to q.*

In order to prove Theorem 1 we will show that the same is true for determinacy in the multiset setting:

LEMMA 11. *$V \xrightarrow{bag} q$ iff there is a path, in $G_{q,V}$, from ε to q.*

The rest of this section is devoted to the proof of Lemma 11.

It turns out that the (simple) proof of the ($\Rightarrow$) direction for the set semantics survives also in the multiset context. We include it here for the sake of completeness, but due to the space limitations defer it to Appendix B.

Let us now deal with the ($\Leftarrow$) direction. Assume that q and V are fixed and such that there is a path in $G_{q,V}$, of some length m, from ε to q. This means that there exist:
• a sequence $w_0, w_1, \dots w_m$ of prefixes of q, with $\varepsilon = w_0$ and $w_m = q$;
• a sequence of numbers ϵ_j, for $j \in \{1, \dots m\}$, each of them either equal 1 or -1;
• a sequence $v_{p_1}, \dots v_{p_m}$ of elements of V such that, for each $j \in \{1, \dots m\}$, one of the conditions is true:

$\quad \bullet \; \epsilon_j = 1$ and $w_j = w_{j-1}v_{p_j}$; $\qquad \bullet \; \epsilon_j = -1$ and $w_jv_{p_j} = w_{j-1}$.

We are going to show that in such case there will also be $V \xrightarrow{bag} q$. So we assume that there are two stuctures D and D' such that $v(D) = v(D')$ for each $v \in V$. Without loss of generality we can also assume that domains of D and D' are equal[16], so let $dom(D) = dom(D') = \{a_1, \dots, a_n\}$.

3.1 The q-walks and how to reduce them.

Let $\bar{\Sigma} = \Sigma \cup \Sigma^{-1}$ be a new alphabet[17], where $\Sigma^{-1} = \{R^{-1} \mid R \in \Sigma\}$.

DEFINITION 12. *Let $w \in \bar{\Sigma}^*$ and $w = A_1^{\iota_1} \dots A_k^{\iota_k}$ for some $A_1, \dots, A_k \in \Sigma$ and for some $\iota_1, \dots, \iota_k \in \{1, -1\}$. Then w is called a q-walk if:*

(1) for each $i \in \{1, \dots, k\}$ it holds that $0 \leq \sum_{j=1}^{i} \iota_j \leq |q|$;
(2) $\sum_{j=1}^{k} \iota_j = |q|$;

---

[14]On the other hand, for path queries, query containment under set semantics also (trivially) coincides with query containment under bag semantics. We have no idea whether there is any relation between this observation and Theorem 1.
[15]Recall that we identify path queries with words over alphabet Σ.
[16]By domain we do not mean the active domain here: we accept that there are elements, in $dom(D)$ or $dom(D')$ which do not appear in any facts.
[17]Or schema, in the world of path queries words and queries are the same thing.

(3) *for each* $i \in \{1, \ldots, k\}$ *it holds that* $A_i = \begin{cases} Q_{s_i+1} & \text{if } \iota_i = 1 \\ Q_{s_i} & \text{if } \iota_i = -1 \end{cases}$

where $s_i = \sum_{j=1}^{i-1} \iota_j$ *and* Q_j *is the j-th symbol in* q.

Our path in $G_{q,V}$, leading from ε to q, induces, in a natural way, a q-walk[18] $(v_{p_1})^{\epsilon_1}(v_{p_2})^{\epsilon_2} \ldots (v_{p_m})^{\epsilon_m}$. For clarity, let us illustrate this with:

EXAMPLE 13. *Imagine that* $q = ABCD$ *and* $V = \{ABC, BC, BCD\}$. *Then there is a path* $\varepsilon \to ABC \to A \to ABCD$ *in* $G_{q,V}$. *This path induces a q-walk* $(ABC)(BC)^{-1}(BCD)$, *which is equal to* $ABCC^{-1}B^{-1}BCD$.

Now we are going to explain how each q-walk can be turned into q by a sequence of simple reductions:

DEFINITION 14. *For any* $w, w' \in \bar{\Sigma}^*$ *and for any* $A \in \Sigma$ *we define:*

$$wAA^{-1}w' \xrightarrow{+/-} ww' \quad \text{and} \quad wA^{-1}Aw' \xrightarrow{-/+} ww'.$$

Relations $\xrightarrow{*}{+/-}$ *and* $\xrightarrow{*}{-/+}$ *are defined as the reflexive transitive closure of* $\xrightarrow{+/-}$ *and of* $\xrightarrow{-/+}$, *respectively.*

LEMMA 15. *If* $w \in \bar{\Sigma}^*$ *is a q-walk, then* $w \xrightarrow{*}{+/-} q$ *and* $w \xrightarrow{*}{-/+} q$.

For the proof of Lemma 15 see Appendix C.

3.2 Seeing D (and D') as relations in $\mathbb{Q}^n \times \mathbb{Q}^n$.

DEFINITION 16. *Let* $R \in \Sigma$. *Then* M_R^D *is the incidence matrix of the relation* R *in structure* D, *that is* $M_R^D \in \mathbb{Q}^{n \times n}$ *and* $M_R^D(i,j) = 1$ *if and only if* $R(a_i, a_j) \in D$ *and* $M_R^D(i,j) = 0$ *if and only if* $R(a_i, a_j) \notin D$.

DEFINITION 17. *Let* $w \in \Sigma^*$. *Then we define a matrix* $M_w^D \in \mathbb{Q}^{n \times n}$ *in the inductive way:*
- M_ε^D *is the* $n \times n$ *identity matrix.*
- *For* $R \in \Sigma, w \in \Sigma^*, M_{Rw}^D = M_R^D M_w^D$.

It is well-known that:

FACT 18. *If* $w \in \Sigma^*$ *then* $w(D)[a_i, a_j] = M_w^D(i,j)$.

Matrices $M_R^{D'}$ and $M_w^{D'}$ are defined analogously, and, obviously, Fact 18 remains true for them.

Of course in general we cannot assume that $M_w^D = M_w^{D'}$ for $w \in \Sigma^*$. But, since for each $v \in V$ we have $v(D) = v(D')$, we know that for each $v \in V$ we have $M_v^D = M_v^{D'}$, so we can write just M_v instead of M_v^D or $M_v^{D'}$. Recall that we need to show that $M_q^D = M_q^{D'}$. So, if we manage to somehow present M_q^D (and hence also $M_q^{D'}$) as a function of arguments $\{M_v \mid v \in V\}$, then we are done.

Let us also remark that if M_v were invertible, for all $v \in V$, then it would be easy to see that $M_q^D = M_{v_{p_1}}^{\epsilon_1} \ldots M_{v_{p_m}}^{\epsilon_m}$ and likewise $M_q^{D'}$. However, in the general case, there is of course no reason to think that the matrices M_v are invertible, and thus we need our argument to be a little bit more sophisticated.

Now the matrices will be understood as linear functions. And these functions will be understood as relations. And, while we know that not all matrices are invertible, and in consequence not all the functions under consideration are, relations can always be inverted!

By I we will denote the identity relation: $I = \{\langle x, x \rangle \mid x \in \mathbb{Q}^n\}$.

DEFINITION 19. (1) *For a matrix* $M \in \mathbb{Q}^{n \times n}$ *let the function* $h_M : \mathbb{Q}^n \to \mathbb{Q}^n$ *be defined as* $h_M(v) = Mv$.

(2) *For a function* f *let* $\bar{f}$ *denote the relation equal*[19] *to* f.

(3) *For* $R \in \Sigma$ *let* $H_R = \overline{h_{M_R^D}}$ *and* $H_{R^{-1}} = H_R^{-1}$.

(4) *For* $w \in \bar{\Sigma}^*$ *we define* H_w *inductively:*
- $H_\varepsilon = I$
- $H_{\alpha w} = H_w H_\alpha$ *for* $\alpha \in \bar{\Sigma}, w \in \bar{\Sigma}^*$

The relations H_w depend on D (in the sense that they would not be equal if we computed them in D' instead of D), so the reader may think that there should be H_w^D instead of H_w. But omitting the superscript leads to no confusion: D is the only structure for which the relations H_w are ever considered.

OBSERVATION 20. *For* $w \in \Sigma^*$, $H_w = \overline{h_{M_w^D}}$ *and* $H_{w^{-1}} = (H_w)^{-1}$

For the proof of the Observation use (easy) induction and the fact that for $w, w' \in \Sigma^*$ it holds that $H_{ww'} = H_{w'} H_w = \overline{h_{M_{w'}^D}} \, \overline{h_{M_w^D}} = \overline{h_{M_w^D} \circ h_{M_{w'}^D}} = \overline{h_{M_{ww'}^D}}$. □

It is well-known that the correspondence $M \mapsto h_M$ is 1-1. Also the correspondence $f \mapsto \bar{f}$ is 1-1. So in order to represent M_q^D as a function of arguments $\{M_v \mid v \in V\}$ it is enough to represent H_q as a function of $\{H_v \mid v \in V\}$. Which we do in the next subsection.

3.3 Using Lemma 15.

Let us start this subsection with a really very simple lemma:

LEMMA 21. *Let* $f : \mathbb{Q}^n \to \mathbb{Q}^n$. *Then* $\bar{f} (\bar{f})^{-1} \supseteq I$ *and* $(\bar{f})^{-1} \bar{f} \subseteq I$.

PROOF. $\bar{f} (\bar{f})^{-1} = \{\langle x, y \rangle \mid \exists z \ f(x) = z \wedge f(y) = z\} = \{\langle x, y \rangle \mid f(x) = f(y)\} \supseteq I$

$(\bar{f})^{-1} \bar{f} = \{\langle x, y \rangle \mid \exists z \ f(z) = x \wedge f(z) = y\} = \{\langle x, x \rangle \mid \exists z \ f(z) = x\} \subseteq I$ □

Now we will see what the relations H_w are good for:

LEMMA 22. *For* $u, u' \in \bar{\Sigma}^*$:
(1) *if* $u \xrightarrow{+/-} u'$ *then* $H_u \subseteq H_{u'}$;
(2) *if* $u \xrightarrow{-/+} u'$ *then* $H_u \supseteq H_{u'}$.

For the proof of Lemma 22 see Appendix C. Notice that, for a q-walk w, Lemmas 22 and 15 give us two approximations of H_w:

LEMMA 23. *If* w *is a q-walk, then* $H_q \subseteq H_w \subseteq H_q$. □

Our next corollary is certainly not going to come as a surprise:

COROLLARY 24. *If* w *is a q-walk, then* $H_q = H_w$.

Now, recall that $(v_{p_1})^{\epsilon_1}(v_{p_2})^{\epsilon_2} \ldots (v_{p_m})^{\epsilon_m}$ is a q-walk. So, by the last corollary $H_q = H_{v_{p_m}}^{\epsilon_m} \ldots H_{v_{p_2}}^{\epsilon_2} H_{v_{p_1}}^{\epsilon_1}$.

Which shows that H_q is indeed a function of $\{H_v \mid v \in V\}$ and ends the proof of Lemma 11($\Leftarrow$) and of Theorem 1.

4 THE BOOLEAN CASE. OUR MAIN RESULTS.

In contrast to the set-semantics world, where determinacy is easily decidable for unary UCQs, and trivially decidable for boolean UCQs, in the multiset setting already the boolean case is undecidable:

THEOREM 2. *The problem whether, for a set* V *of boolean UCQs and for another boolean UCQ* q, *it holds that* $V \xrightarrow{bag} q$ *is undecidable.*

---

[18]If $w \in \Sigma^*$, by w^{-1} we mean w reversed with every letter α replaced with α^{-1}.

[19]This means that $\bar{f} = \{\langle x, y \rangle \mid f(x) = y\}$. We make such distinction since composition and inversion work for functions slightly differently than for relations.

This negative result is not really hard to prove (see Appendix A). The main technical result of this paper, however, is:

THEOREM 3. *The problem whether, for a given set V_0 of boolean conjunctive queries and for a given boolean conjunctive query q, it holds that $V_0 \xrightarrow{bag} q$, is decidable.*

The rest of this section, and Sections 5-7 are devoted to the proof of Theorem 3. A set V_0 of boolean conjunctive queries and a boolean conjunctive query q are fixed from now on.

DEFINITION 25. *Let V be the set $\{v \in V_0 \mid q \subseteq_{set} v\}$. Let us also denote $V' = V \cup \{q\}$.*

Queries from V are the ones that cannot return 0 in any interesting (from the point of view of this proof) structure D. Queries from $V \setminus V$ are free to return 0, and they actually will.

OBSERVATION 26. *If D is any structure, $v \in V$ and $v(D) = 0$ then also $q(D) = 0$.*

DEFINITION 27. *Let $W = \{w_1, ..., w_k\}$ be the set[20] of all connected components of the query[21] $\sum_{v \in V'} v$. In other words, W is a minimal set of structures such that for every connected component u of $\sum_{v \in V'} v$ there is $w \in W$ isomorphic to u. From now on, the letter k will always denote the cardinality of W.*

Queries from W are going to serve us as *basis* queries, in the linear algebra sense:

OBSERVATION 28. *Let $v \in V'$. Then $v = \sum_{i=1}^{k} a_i w_i$ for some $a_1, ..., a_k \in \mathbb{N}$.*

Note, that such representation is unique. Thus:

DEFINITION 29. *For a query $v \in V'$ we define the vector representation of v as $\vec{v} = \begin{bmatrix} a_1 \\ \vdots \\ a_k \end{bmatrix}$, where $a_1, ..., a_k$ are as in Observation 28.*

So all the queries of interest are now seen as vectors in some k-dimensional vector space.

OBSERVATION 30. *If D is any structure and $v \in V'$ then $v(D) = \prod_{i=1}^{k} w_i(D)^{\vec{v}(i)}$.*

Proof: Notice that $v(D) = |hom(v, D)| = |hom(\sum_{i=1}^{k} \vec{v}(i) w_i, D)| = \prod_{i=1}^{k} w_i(D)^{\vec{v}(i)}$. The last equality follows from Lemma 4. $\square$

Now we are ready for our **Main Lemma**:

LEMMA 31. $V_0 \xrightarrow{bag} q$ *if and only if $\vec{q} \in span\{\vec{v} \mid v \in V\}$.*

Clearly, Theorem 3 easily follows from Lemma 31 as finding V is of course decidable (in Σ_2^P – we first need to guess a set of homomorphisms and then check that we guessed all of them), while finding W and testing whether $\vec{q} \in span\{\vec{v} \mid v \in V\}$ are polynomial.

In Sections 5–7 we present the (more complicated) ($\Rightarrow$) part of the proof of Lemma 31. The (much easier) ($\Leftarrow$) part is deferred to Appendix D. We however illustrate the idea of the ($\Leftarrow$) part with:

EXAMPLE 32. *Let w_1, w_2, w_3 be some non-empty, pairwise non-isomorphic structures and let:*
$$q = w_1 + w_2 + 2w_3$$
$$v_1 = 2w_1 + w_2 + 3w_3 \qquad v_2 = 5w_1 + 2w_2 + 7w_3$$
Then for a structure D: $q(D) = w_1(D)w_2(D)w_3(D)^2$,
$v_1(D) = w_1(D)^2 w_2(D) w_3(D)^3$ *and* $v_2(D) = w_1(D)^5 w_2(D)^2 w_3(D)^7$.

If $v_2(D) \neq 0$, then $q(D) = v_1(D)^3/v_2(D)$ so it is uniquely determined by $v_1(D), v_2(D)$. This equality corresponds to the equality of vector representations $\vec{q} = 3\vec{v}_1 - \vec{v}_2$. If $v_2(D) = 0$, then for some $i \in \{1, 2, 3\}$, $w_i(D) = 0$, so $q(D) = 0$ and it is determined again.

It easily follows from Lemma 31 that in the very specific case of connected queries no non-trivial determinacy is possible:

COROLLARY 33. *If all the queries in V_0 are connected, and q is connected, then $V_0 \xrightarrow{bag} q$ if and only if $q \in V_0$.*

5 PROOF OF LEMMA 31 ($\Rightarrow$). PART 1.

In this section we assume that $\vec{q} \notin span\{\vec{v} \mid v \in V\}$. And we are going to show that $V_0 \xrightarrow{bag} q$. To this end we need to find a pair of structures D and D' which is a counterexample for determinacy, which means that:

(A) $q(D) \neq q(D')$
(B) $\forall v \in V \quad v(D) = v(D')$ (B0) $\forall v \in V_0 \setminus V \quad v(D) = v(D')$

Notice that there is nothing in Definition 1 that would tell us where to look for such a counterexample: D and D' are just **any** structures in this definition. Our main discovery is that if such D and D', forming a counterexample, can be found at all, then a counterexample can also be found in some k-dimensional[22] vector space that we are now going to introduce. And this is convenient, because living in a vector space one can use linear algebra tools.

DEFINITION 34. *For any set S of k structures (call them basis structures) let $\mathcal{S}$ be the set of all structures which can be represented as sums of elements of S, that is $\mathcal{S} = span^{\mathbb{N}}(S)$.*

Now, the totally informal idea is as follows. We know that $\vec{q} \notin span\{\vec{v} \mid v \in V\}$. So there is vector $\vec{z}$ which is orthogonal to $\vec{v}$ for each $v \in V$ but not to $\vec{q}$. Let us *somehow* define D and D' in such a way that $\vec{z}$ is *"the difference"* between D and D'. Then none of the $v \in V$ will spot the difference between D and D' but q will.

DEFINITION 35. *Set S of basis structures is decent if for each $s \in S$ and for each $v \in V_0 \setminus V$ we have $v(s) = 0$.*
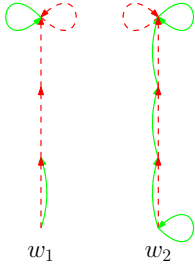
It is easy to see that:

OBSERVATION 36. *If S is decent, then for each $D \in \mathcal{S}$ and for each $v \in V_0 \setminus V$ we have $v(D) = 0$. In consequence, if S is decent, then any pair D, D' of structures from $\mathcal{S}$ satisfies condition (B0) above.*

DEFINITION 37. *For a set of structures $S = \{s_1, ..., s_{|S|}\}$ we define its evaluation matrix $M_S \in \mathbb{R}^{k \times |S|}$ by the formula $M_S(i, j) = w_i(s_j)$.*

In other words, the (i, j)-entry of M_S is defined as the number of homomorphisms from w_i to s_j.

DEFINITION 38. *S is good when S is decent and M_S is nonsingular.*

---

[20]When we say "set" we mean that each such connected component only occurs once in W. And we think that isomorphic structures are equal.
[21]$\sum$ is an operation on structures here, as defined in Section 2.2.

[22]Recall, that k denotes, as always, the cardinality of W, the set of basis queries.

Figure 1: (Example 39) Here, M_W is singular (w_1 and w_2 are structures over a schema consisting of two binary relations, with w_2 having three additional green edges, compared to w_1).

Recall that the set W, consisting of k queries, is also a set of k structures. What would happen if we just took W as our set S of basis structures? W is of course always decent: if there were $w \in W, v \in V_0 \setminus V$ such that $v(w) > 0$, then, because $w(q) > 0$, we would get $v(q) > 0$. But M_W is not always nonsingular:

EXAMPLE 39. *Let W consist of w_1 and w_2 as in Fig. 1, then:*

$$M_W = \begin{array}{c} \\ w_1 \\ w_2 \end{array} \begin{array}{c} w_1 \quad w_2 \\ \left[\begin{array}{cc} 2 & 4 \\ 1 & 2 \end{array} \right] \end{array}$$

where $M_W(i, j) = |hom(w_i, w_j)|$.

Now, **proof of Lemma 31 will be completed once we prove the following two lemmas:**

LEMMA 40. *There exists a good set S of basis structures.*

LEMMA 41. *If $\vec{q} \notin span\{\vec{v} \mid v \in V\}$ and S is a good set of basis structures then there exist $D, D' \in S$ satisfying conditions (A) and (B) above.*

For the proof of Lemma 40 see Section 6, and for the proof of Lemma 41 see Section 7. Notice that Lemma 41 would not be true without the assumption that M_S is nonsingular:

EXAMPLE 42. *Let $q = w_1$ and $V_0 = \{w_2\}$ (w_1, w_2 are still as in Fig. 1), so that (according to Definition 27) $W = \{w_1, w_2\}$. Also, since $w_1 \subseteq_{set} w_2$, we get that $V = V_0$.*

Since $\vec{q} \notin span\{\vec{v} \mid v \in V\}$, it follows from our Main Lemma that $V \xrightarrow{bag} q$. So couldn't we just take $S = W$ (notice that, since $V = V_0$, such S is trivially decent) and look for the counterexample structures D and D' in $\mathcal{S} = span^{\mathbb{N}}(S)$?

This would be in vain. For any structure $D \in \mathcal{S}$ the equality $|hom(w_1 = q, D)| = 2 \cdot |hom(w_2, D)|$ holds. So for any pair of structures $D, D' \in \mathcal{S}$ there will be $V_0(D) = V_0(D') \Rightarrow q(D) = q(D')$.

Let us however reiterate: the above example does not contradict our Main Lemma. It only shows that $S = \{w_1, w_2\}$ is not **good** enough to serve as a basis for a counterexample pair D, D'.

6 PROOF OF LEMMA 40

Our proof relies on the following lemma stated in [1] and proved, as [19] report, in the paper [20], which is not easy to access.

LEMMA 43. *Two structures G, G' are isomorphic if and only if $|hom(G, H)| = |hom(G', H)|$ for every structure[23] H.*

However, the proof of Lemma 43 is analogous to the proof of:

---
[23]We of course assume here that all the structures in question are over some fixed relational schema.

LEMMA 44. *Two structures G, G' are isomorphic if and only if $|hom(H, G)| = |hom(H, G')|$ for every structure H.*

in the paper [16], so we think we can skip it here.

A **good S will now be constructed** in four steps:

Step 1. $S^{(1)} = \{s_1^{(1)}, ..., s_m^{(1)}\}$ can be any finite set such that

$$\forall w \neq w' \in W \; \exists i \leq m \quad |hom(w, s_i^{(1)})| \neq |hom(w', s_i^{(1)})|$$

Such $S^{(1)}$ can be found thanks to Lemma 43. Indeed, because elements of W are pairwise non-isomorphic, for any $w \neq w'$ there is a structure H such that $w(H) \neq w'(H)$ - it is enough to take one such structure for every pair $w \neq w' \in W$.

In the Steps 2-4 we construct a good S from $S^{(1)}$ using addition and multiplication of structures. And, by Lemma 4, addition and multiplication of structures correspond to addition and multiplication (elementwise) of columns of the evaluation matrix. So this part is more about linear algebra than about homomorphism counting.

Step 2. Let $T \in \mathbb{N}$ be greater than any element of the matrix $M_{S^{(1)}}$. Then the set $S^{(2)}$ consists of a single structure $s^{(2)}$ where:

$$s^{(2)} = \sum_{i=1}^{m} T^i s_i^{(1)}$$

OBSERVATION 45. *Suppose $w, w' \in W$ and $w \neq w'$. Then $|hom(w, s^{(2)})| \neq |hom(w', s^{(2)})|$.*

For the proof of Observation 45 see Appendix D.

Step 3. Let now $S^{(3)} = \{s_1^{(3)}, ..., s_k^{(3)}\}$ be a set of k structures, where $s_i^{(3)} = \left(s^{(2)}\right)^{i-1}$. We are going to prove that the matrix $M_{S^{(3)}}$ is nonsingular. Recall that $M_{S^{(3)}}(i, j) = |hom(w_i, s_j^{(3)})|$. Notice that:

$$|hom(w_i, s_j^{(3)})| = |hom(w_i, \left(s^{(2)}\right)^{j-1})| = |hom(w_i, s^{(2)})|^{j-1}$$

Then use the following lemma, which is proven in Appendix D:

LEMMA 46. *Let $a_1, ..., a_k$ be pairwise-distinct real numbers. Then the matrix $A \in \mathbb{R}^{k \times k}$ defined as $A(i, j) = a_i^{j-1}$ is nonsingular.*

Step 4. Now, $S^{(3)}$ is almost good. Almost, because we are still not sure if it is *decent*. So let $S^{(4)} = \{s_1^{(4)}, ..., s_k^{(4)}\}$ where $s_i^{(4)} = s_i^{(3)} \times q$. Observe that $M_{S^{(4)}}$ is just $M_{S^{(3)}}$ where i-th row has been multiplied by $w_i(q)$. However we know that w_i is a subquery of some query $v \in V \cup \{q\}$ - such a v satisfies $v(q) > 0$. Therefore, $w_i(q) > 0$ and it is well-known that multiplying a matrix row by a non-zero factor doesn't affect its (non)singularity.

Let's observe that $S^{(4)}$ is *decent*, that is, $\forall v \in V_0 \setminus V, s \in S^{(4)} \; v(s) = 0$. Indeed, any $s \in S$ is of form $s = s' \times q$, so for any $v \in V_0 \setminus V$ we have $v(s) = v(s')v(q)$ and, by definition of V, $v(q) = 0$.

To sum up, we have found a good set of basis structures $S^{(4)}$.

From now on, we put $S = S^{(4)}$ and $s_i = s_i^{(4)}$ for $i \in \{1, ..., k\}$. We will also write M instead of M_S.

7 PROOF OF LEMMA 41

First some formulae which we will need:

DEFINITION 47 (VECTOR REPRESENTATION OF A STRUCTURE $s \in \mathcal{S}$).

For $s = \sum_{i=1}^{k} a_i s_i$ we define $\vec{s} = \begin{bmatrix} a_1 \\ \vdots \\ a_k \end{bmatrix}$

DEFINITION 48.

(1) *Let $\vec{u}, \vec{v} \in \mathbb{R}^k$. Then* $\vec{u} \circ \vec{v} = \begin{bmatrix} \vec{u}(1)\vec{v}(1) \\ \vdots \\ \vec{u}(k)\vec{v}(k) \end{bmatrix}$

(2) *Let $\vec{u}, \vec{v} \in \mathbb{R}_{\geq 0}^k$. Then $\vec{u} \; \sigma^{\!\!\!\prime} \; \vec{v} = \prod_{i=1}^{k} \vec{u}(i)^{\vec{v}(i)}$.*

(3) *Let $t \in \mathbb{R}_+, \vec{u} \in \mathbb{R}^k$. Then* $t^{\vec{u}} = \begin{bmatrix} t^{\vec{u}(1)} \\ \vdots \\ t^{\vec{u}(k)} \end{bmatrix}$

OBSERVATION 49. (1) $(\vec{u} \circ \vec{v}) \; \sigma^{\!\!\!\prime} \; \vec{w} = (\vec{u} \; \sigma^{\!\!\!\prime} \; \vec{w})(\vec{v} \; \sigma^{\!\!\!\prime} \; \vec{w})$
(2) $t^{\vec{u}} \; \sigma^{\!\!\!\prime} \; \vec{v} = t^{\langle \vec{u}, \vec{v} \rangle}$

LEMMA 50. *Let $v \in V, s \in \mathcal{S}$. Then $v(s) = (M\vec{s}) \; \sigma^{\!\!\!\prime} \; \vec{v}$.*

For the proof of Lemma 50 see Appendix D.

7.1 The set $\mathcal{P}$ and the cone C

So far the objects of our interest in this proof lived in two k-dimensional vector spaces. One was the space of queries, with W as the basis. Another one was the space $\mathcal{S}$ of structures, with basis S, where we are looking for the candidates for D and D'.

But we also need the third such k-dimensional space. Imagine you take some structure $s \in \mathcal{S}$. And you ask what will be the results of applying the k queries from W to s. What you get is a k-dimensional vector of natural numbers, which lives in the space of all possible (and impossible) answer vectors.

DEFINITION 51. $\mathcal{P} = \{M\vec{s} \mid s \in \mathcal{S}\} = \{M\vec{u} \mid \vec{u} \in \mathbb{N}^k\}$

$\mathcal{P}$ is the subset of our new space consisting of the actual answer vectors, generated by real structures from $\mathcal{S}$. A related notion is:

DEFINITION 52. $C = span^{\mathbb{R}_{\geq 0}}\{M\vec{s} \mid s \in S\} = span^{\mathbb{R}_{\geq 0}}\{Me_i \mid i \in \{1, \dots, k\}\}$. *In other words, C is a convex cone generated by basis standard vectors multiplied by matrix M.*

The following easy observation shows that $\mathcal{P}$ is a subset of C. A proper subset, since only vectors of natural numbers can be in $\mathcal{P}$. And there is even no reason to think that $\mathcal{P} = C \cap \mathbb{N}^k$.

OBSERVATION 53 (EASY). $C = M(\mathbb{R}_{\geq 0}^k) = span^{\mathbb{R}_{\geq 0}}\{M\vec{s} \mid s \in \mathcal{S}\}$

EXAMPLE 54. *Let w_1, w_2 be as in Fig. 1. Let s_1 be a single vertex, with red and green loops and let $s_2 = w_2$ Then:*

$$M_S = \begin{matrix} & \begin{matrix} s_1 & s_2 \end{matrix} \\ \begin{matrix} w_1 \\ w_2 \end{matrix} & \begin{bmatrix} 1 & 4 \\ 1 & 2 \end{bmatrix} \end{matrix}$$

Then C and $\mathcal{P}$ are as in Fig. 2. Notice that M_S is now non-singular. This observation is not unrelated to the fact that the grey area in Fig. 2 has non-empty interior.
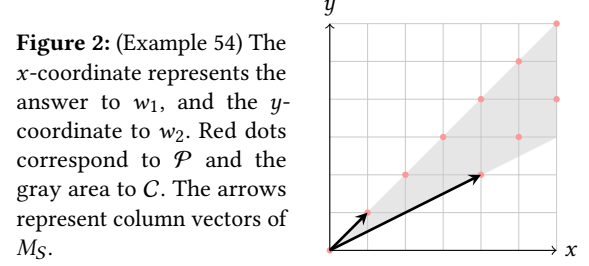

Figure 2: (Example 54) The x-coordinate represents the answer to w_1, and the y-coordinate to w_2. Red dots correspond to $\mathcal{P}$ and the gray area to C. The arrows represent column vectors of M_S.

7.2 It is here where things finally happen

We spent several pages pushing a rabbit into the hat. Now we are finally going to pull it out. First, we notice that while not all the vectors in $C \cap \mathbb{Q}^k$ are in $\mathcal{P}$, all of them are somehow related to $\mathcal{P}$:

LEMMA 55. *Let $\vec{u} \in C \cap \mathbb{Q}^k$. Then there exists $c \in \mathbb{N}_+$ with $c\vec{u} \in \mathcal{P}$.*

PROOF. M is nonsingular[24] so there exists $M^{-1} \in \mathbb{Q}^{k \times k}$. Let $\vec{\alpha} = M^{-1}\vec{u}$. Since $\vec{u} \in C \cap \mathbb{Q}^k$ we have that $\vec{\alpha} \in C \cap \mathbb{Q}^k$. Since $\vec{u} \in \{M\vec{v} \mid \vec{v} \in \mathbb{R}_{\geq 0}^k\}$ we know that $\vec{\alpha} \in \mathbb{R}_{\geq 0}^k$. So there is $c \in \mathbb{N}_+$ such that[25] $c\vec{\alpha} \in \mathbb{N}_+$. Now, $c\vec{u} = c(M\vec{\alpha}) = M(c\vec{\alpha}) \in \mathcal{P}$. □

LEMMA 56. *There are $\vec{p}, \vec{p}' \in C \cap \mathbb{Q}^k$ such that:*
(1) $\forall v \in V \; \vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v} = \vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v}$ (2) $\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{q} \neq \vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{q}$

Before we prove **Lemma 56** let us show that it **implies Lemma 41**: Indeed, if we find p, p' as in Lemma 56, then, by Lemma 55 we can find $c, c' \in \mathbb{N}_+$ such that $c\vec{p}, c'\vec{p}' \in \mathcal{P}$. Of course then $cc'\vec{p}, cc'\vec{p}' \in \mathcal{P}$ too. Let $\vec{cc'} = \begin{bmatrix} cc' \\ \vdots \\ cc' \end{bmatrix}$. Then, for $v \in V \cup \{q\}$ we have

$(cc'\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v}) - (cc'\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v}) =$

$= (\vec{cc'} \; \sigma^{\!\!\!\prime} \; \vec{v})(\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v}) - (\vec{cc'} \; \sigma^{\!\!\!\prime} \; \vec{v})(\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v})$ (by Observation 49)

$= (\vec{cc'} \; \sigma^{\!\!\!\prime} \; \vec{v})((\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v}) - (\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v}))$

Because $cc' > 0$, also $(\vec{cc'} \; \sigma^{\!\!\!\prime} \; \vec{v}) > 0$ and we get:
$(cc'\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v}) - (cc'\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v}) = 0$ iff $(\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v}) - (\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v}) = 0$.
Then we take $s, s' \in \mathcal{S}$ such that $M\vec{s} = cc'\vec{p}, M\vec{s'} = cc'\vec{p}'$. By Lemma 50 we have:
(1) for $v \in V, v(s) = cc'\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{v} = cc'\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{v} = v(s')$
(2) $q(s) = cc'\vec{p} \; \sigma^{\!\!\!\prime} \; \vec{q} \neq cc'\vec{p}' \; \sigma^{\!\!\!\prime} \; \vec{q} = q(s')$
So $D = s, D' = s'$ are the structures as postulated by Lemma 41.

The last thing needed for the proof of Lemma 41 is:

PROOF OF LEMMA 56. Take $\vec{z_0} \in \mathbb{Q}^k$ such that (1) $\forall v \in V \; \langle \vec{z_0}, \vec{v} \rangle = 0$ and (2) $\langle \vec{z_0}, \vec{q} \rangle \neq 0$. Such $\vec{z_0}$ exists thanks to Fact 5 and to the assumption that $\vec{q} \notin span\{\vec{v} \mid v \in V\}$. Then take $d \in \mathbb{N}_+$ such that $d\vec{z_0} \in \mathbb{Z}^k$. Let $\vec{z} = d\vec{z_0}$. Clearly, $\vec{z}$ satisfies conditions (1) and (2) too.

Let $\vec{p}$ and r be as in Corollary 8. This means that $\vec{p} \in C \cap \mathbb{Q}^k$ and the ball with center $\vec{p}$ and radius r is contained in C, with $r > 0$. So we already have $\vec{p}$ and we will find $\vec{p}'$ in this ball.

LEMMA 57. *There exists $t \in \mathbb{R}_+ \setminus \{1\}$ such that $t^{\vec{z}} \circ \vec{p} \in C \cap \mathbb{Q}^k$.*

---

[24]Finally we are using this nonsingularity. But it is the proof of Lemma 56 where it is really fundamentally needed.
[25]Take a common multiple of all denominators of the coordinates of $\vec{\alpha}$.

PROOF. Observe that the function $t \mapsto t^{\vec{z}} \circ \vec{p}$ is continuous and it maps 1 to $\vec{p}$. Thus there is $\delta > 0$ such that:

$$\forall t \in (1 - \delta, 1 + \delta) \;\; \|t^{\vec{z}} \circ \vec{p} - \vec{p}\| < r$$

It is now enough[26] to take any $t \neq 1$ in $(1 - \delta, 1 + \delta) \cap \mathbb{Q}$. □

Let $\vec{p}' = t^{\vec{z}} \circ p$ where t is as in Lemma 57. By Observation 49:
• For $v \in V$, $\vec{p}' \circlearrowleft \vec{v} = (t^{\vec{z}} \circ p) \circlearrowleft \vec{v} = (t^{\vec{z}} \circlearrowleft \vec{v})(\vec{p} \circlearrowleft \vec{v}) = t^{\langle \vec{z}, \vec{v} \rangle}(\vec{p} \circlearrowleft \vec{v}) = \vec{p} \circlearrowleft \vec{v}$
• $\vec{p}' \circlearrowleft \vec{q} = (t^{\vec{z}} \circ p) \circlearrowleft \vec{q} = (t^{\vec{z}} \circlearrowleft \vec{q})(\vec{p} \circlearrowleft \vec{q}) = t^{\langle \vec{z}, \vec{q} \rangle}(\vec{p} \circlearrowleft \vec{q}) \neq \vec{p} \circlearrowleft \vec{q}$
This ends the proof of Lemma 56, of Lemma 41 and of Theorem 3.

REFERENCES

[1] S. Chaudhuri and M. Y. Vardi, "Optimization of real conjunctive queries," in *Proceedings of the Twelfth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, PODS '93, (New York, NY, USA), p. 59–70, Association for Computing Machinery, 1993.

[2] J. Marcinkowski, "What Makes a Variant of Query Determinacy (Un)Decidable? (Invited Talk)," in *23rd International Conference on Database Theory (ICDT 2020)* (C. Lutz and J. C. Jung, eds.), vol. 155 of *Leibniz International Proceedings in Informatics (LIPIcs)*, (Dagstuhl, Germany), pp. 2:1–2:20, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020.

[3] A. Nash, L. Segoufin, and V. Vianu, "Determinacy and rewriting of conjunctive queries using views: A progress report," in *Database Theory – ICDT 2007* (T. Schwentick and D. Suciu, eds.), (Berlin, Heidelberg), pp. 59–73, Springer Berlin Heidelberg, 2006.

[4] T. Gogacz and J. Marcinkowski, "The hunt for a red spider: Conjunctive query determinacy is undecidable," in *Proceedings of the 2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, LICS '15, (USA), p. 281–292, IEEE Computer Society, 2015.

[5] J.-F. Baget, M. Leclère, M.-L. Mugnier, and E. Salvat, "On rules with existential variables: Walking the decidability line," *Artificial Intelligence*, vol. 175, no. 9, pp. 1620–1654, 2011.

[6] V. Bárány, B. Cate, and L. Segoufin, "Guarded negation," vol. 62, pp. 356–367, 07 2011.

[7] A. Atserias and P. Kolaitis, "Structure and complexity of bag consistency," 12 2020.

[8] Y. E. Ioannidis and R. Ramakrishnan, "Containment of conjunctive queries: Beyond relations as sets," *ACM Trans. on Database Systems (TODS)*, 1995.

[9] T. S. Jayram, P. G. Kolaitis, and E. Vee, "The containment problem for <bi>real</bi> conjunctive queries with inequalities," in *Proceedings of the Twenty-Fifth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '06, (New York, NY, USA), p. 80–89, Association for Computing Machinery, 2006.

[10] F. N. Afrati, M. Damigos, and M. Gergatsoulis, "Query containment under bag and bag-set semantics," *Information Processing Letters*, vol. 110, no. 10, pp. 360–369, 2010.

[11] M. Abo Khamis, P. G. Kolaitis, H. Q. Ngo, and D. Suciu, "Bag query containment and information theory," in *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS'20, (New York, NY, USA), p. 95–112, Association for Computing Machinery, 2020.

[12] C. Beeri, R. Fagin, D. Maier, and M. Yannakakis, "On the desirability of acyclic database schemes," *J. ACM*, vol. 30, p. 479–513, July 1983.

[13] F. Afrati, "Determinacy and query rewriting for conjunctive queries and views," *Theor. Comput. Sci.*, vol. 412, pp. 1005–1021, 03 2011.

[14] L. Segoufin and V. Vianu, "Views and queries: Determinacy and rewriting," in *Proceedings of the Twenty-Fourth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '05, (New York, NY, USA), p. 49–60, Association for Computing Machinery, 2005.

[15] H. Chen and S. Mengel, "The logic of counting query answers," in *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pp. 1–12, IEEE Computer Society, 2017.

[16] L. Lovász, "Operations with structures," *Acta Mathematica Academiae Scientiarum Hungarica*, vol. 18, no. 3-4, pp. 321–328, 1967.

[17] P. Erdős, L. Lovász, and J. Spencer, "Strong independence of graphcopy functions," 1979.

[18] L. Lovász, *Large Networks and Graph Limits*, vol. 60 of *Colloquium Publications*. American Mathematical Society, 2012.

[19] A. Atserias, P. Kolaitis, and W.-L. Wu, "On the expressive power of homomorphism counts," 01 2021.

[20] S. Fisk, "Distinguishing graphs by the number of homomorphisms," *Discuss. Math. Graph Theory*, vol. 15, pp. 73–75, 1995.

[26]It is here where one needs $\vec{z}$ to be in $\mathbb{Z}^k$, otherwise $t^{\vec{z}}$ would not be rational.

8 APPENDIX A — THE UCQ CASE

This section is entirely devoted to the proof of Theorem 2.

As our source of undecidability we take **Hilbert's Tenth Problem**. It is well known that the following problem is undecidable:

PROBLEM 58. *Given a polynomial equation with finite number of unknowns and integer coefficients, determine whether it has a solution such that every unknown is a natural number.*

An instances of Hilbert's Tenth Problem can be seen as a set of monomials (with integer coefficients). For a given monomial m we will denote with $c(m)$ its coefficient and with $m(x)$ we denote degree of m with respect to x in m (if x in not present in m then $m(x) = 0$).

In order to prove Theorem 2 we will construct a reduction from (the complement) of Hilbert's Tenth Problem. As an instance of that problem we are given a set $I = \{m_1, m_2, \ldots, m_k\}$ of monomials. Let $x_1, x_2, \ldots, x_n$ be the unknowns present in I. We are going to produce a schema Σ, a boolean UCQ q and a set V of boolean UCQ such that I has **no** solution if and only if $V \xrightarrow{\text{bag}} q$.

We start with Σ, which will consist[27] of nullary and unary predicates: $H, C, X_1(x), \ldots, X_n(x)$. For a structure D and for $R \in \Sigma$ let us denote, with D_R, the number of atoms of relation R in structure D. Notice that, since H and C are nullary, $D_H, D_C \in \{0, 1\}$ for each D.

Now. the general idea, that one could have in mind, is that the upcoming set of boolean CQs V will make sure that any pair of *distinct* structures D, D' over Σ satisfying $V(D) = V(D')$ is equal on X_i and differs on H and C.

Before we can define V let us construct two UCQs Ψ_N and Ψ_P. First, for every monomial m we define the following boolean CQ:

$$\Phi_m = \exists^* \bigwedge_{X_i \in \Sigma} \bigwedge_{j=1}^{m(x_i)} X_i(y_{i,j})$$

where the quantifier $\exists^*$ binds all the variables $y_{i,j}$ that occur in the formula.

For a structure D over Σ and for $m \in I$ let m_D be the value of m after substituting, for each unknown x_i in m, the number D_{X_i}. For a solution f of instance I we write m_f to denote the value of m after substituting each unknown x_i with its value in solution f.

LEMMA 59. *For each D and each $m \in I$:*

$$m_D = c(m) \cdot \Phi_m(D)$$

PROOF. It follows from Lemma 4 (5). □

Let P be subset of I containing monomials with positive coefficients and let N contain monomials with negative coefficients, then define:

$$\Psi_P = \bigvee_{m \in P} \bigvee_{i=1}^{c(m)} \Phi_m \wedge H, \qquad \Psi_N = \bigvee_{m \in N} \bigvee_{i=1}^{c(m)} \Phi_m \wedge C.$$

LEMMA 60. *For each D it holds that:*

$$D_H \cdot \sum_{m \in P} m_D = \Psi_P(D).$$

[27]One needs to mention here that our nullary predicates come from [14] and [2] and our unary predicates come from [8].

PROOF.

$$\Psi_P(D) = \sum_{m \in P} \sum_{i=1}^{c(m)} (\Phi_P \wedge H)(D) \qquad \text{(Lemma 4)}$$

$$= \sum_{m \in P} \sum_{i=1}^{c(m)} D_H \cdot \Phi_P(D)$$

$$= D_H \cdot \sum_{m \in P} \sum_{i=1}^{c(m)} \Phi_P(D)$$

$$= D_H \cdot \sum_{m \in P} c(m) \cdot \Phi_P(D)$$

$$= D_H \cdot \sum_{m \in P} m_D \qquad \text{(Lemma 59)}$$

$\square$

LEMMA 61. *For each D it holds that:*

$$D_C \cdot \sum_{m \in N} m_D = -\Psi_N(D)$$

PROOF. Analogous to the proof of Lemma 60. $\square$

Finally we are able to define a query q and a set of queries V. Our boolean UCQ q will simply be equal to H. The set V will contain the following boolean UCQs :

- $V_1 = H \vee C$,
- $V_{X_i} = \exists y \, X_i[y]$ for each X_i in schema Σ,
- $V_I = \Psi_P \vee \Psi_N$.

The above definition of V implies the following property:

LEMMA 62. *For every pair of distinct structures D, D' such that $V(D) = V(D')$ the following holds:*

$$D_{X_i} = D'_{X_i}, \qquad D_H = D'_C, \qquad D_C = D'_H, \qquad D_H \neq D_C$$

PROOF. Property $D_{X_i} = D'_{X_i}$ is obvious given views V_{X_i}. From $V_1(D) = V_1(D')$ we get following possibilities:

(1) $D_H = D'_C, \quad D_C = D'_H, \quad D_H \neq D_C$ $(V_1(D) = 1)$
(2) $D_H = D'_H, \quad D_C = D'_C, \quad D_H \neq D_C$ $(V_1(D) = 1)$
(3) $D_H = D'_H = D_C = D'_C = 0$ $(V_1(D) = 0)$
(4) $D_H = D'_H = D_C = D'_C = 1$ $(V_1(D) = 2)$

From $D_{X_i} = D'_{X_i}$ and the fact that $D \neq D'$ we conclude that only (1) can hold. $\square$

Thus whenever we will have two different structures D, D' satisfying $V(D) = V(D')$ we will assume without loss of generality that: $D_H = D'_C = 1$ and $D_C = D'_H = 0$. Notice that this implies that $q(D) \neq q(D')$ for such D and D'.

To finish the proof of Theorem 2 it is now enough to show:

LEMMA 63. *There exists a pair of different structures D, D' over schema Σ that satisfies $V(D) = V(D')$ if and only if I has a solution over natural numbers.*

PROOF. ($\Leftarrow$). Let f be a solution over $\mathbb{N}$ of I and let a_i be a value of x_i in f. Then let D and D' be such that:

- $D_H = 1, D'_H = 0, D_C = 0, D'_C = 1$,
- $D_{X_i} = D'_{X_i} = a_i$,

From Lemmas 60 and 61 we show that:

$$V_I(D) - V_I(D') = \sum_{m \in P} m_f + \sum_{m \in N} m_f = \sum_{m \in I} m_f = 0$$

($\Rightarrow$). Now we will show that $\sum_{m \in I} m_D = 0$. From Lemmas 60 to 62 we get:

$$V_I(D) = V_I(D')$$

$$\Psi_P(D) + \Psi_N(D) = \Psi_P(D') + \Psi_N(D')$$

$$\Psi_P(D) = \Psi_N(D')$$

$$\sum_{m \in P} m_D = -\sum_{m \in N} m_D$$

$$\sum_{m \in P} m_D + \sum_{m \in N} m_D = 0$$

$$\sum_{m \in I} m_D = 0$$

$\square$

9 APPENDIX B. PROOF OF LEMMA 11($\Rightarrow$).

Suppose there is no path, in $G_{q,V}$, from ε to q. We will show that in such case V does not determine q. Let structure D be defined in the following way:

- $dom(D) = \{[w, j] \mid w \text{ is a prefix of } q, j \in \{0, 1\}\}$
- For $[w, i], [u, j] \in dom(D), R \in \Sigma$ we have $R([w, i], [u, j]) \in D$ if and only if $u = wR$ and $i = j$.

So D is just $q + q$, that is the union of two disjoint frozen bodies of q. It follows easily from the definition that $\langle [\varepsilon, 0], [q, 0] \rangle \in q(D)$ with multiplicity 1.

For $w, u \in G_{q,V}$ we define $w \sim u$ if either both w and u are reachable, in graph $G_{q,V}$, from ε or if none of them is. Clearly, $w \sim w'$ is an equivalence relation with two equivalence classes, and with $\varepsilon \not\sim q$.

We will now define the second structure, our D', with the same domain as D, but different atoms:
For $u = wR$:

- $R([w, i], [u, i]) \in D'$ if and only if $u = wR$ and $w \sim u$;
- if $i \neq j$ then $R([w, i], [u, j]) \in D'$ if and only if $u = wR$ and $w \not\sim u$.

Notice that this means that if there is any path in D' from some $[w, i]$ to $[u, j]$ then:

$$i = j \text{ if and only if } w \sim u.$$

Which in particular means that $\langle [\varepsilon, 0], [q, 0] \rangle \notin q(D')$ and hence $q(D) \neq q(D')$.

But on the other hand, for $v \in V$, it is very easy to see that if $uvu' = q$ for some $u, u' \in \Sigma^*$ and if $i \in \{0, 1\}$ then $\langle [u, i], [uv, i] \rangle \in v(D)$, with multiplicity 1, and that there are no other tuples in $V(D)$. And it is also not hard to verify that such $V(D)$ exacty equals $V(D')$. This is since, if u and v are as above, then $u \sim uv$.

10 APPENDIX C. PROOFS OF SOME LEMMAS NEEDED FOR THEOREM 1.

10.1 Proof of Lemma 15

We only show the first claim, as the other one is symmetric. The proof will be by induction with respect to $|w|$:

(1) $|w| = |q|$. Then $w = q$ and we are done.

(2) $|w| > |q|$. Then there is i such that $\iota_i = -1$. By Definition 12 (1) we know that $\iota_1 = 1$, so there exists $j < i$ such that $\iota_j = 1$ and $\iota_{j+1} = -1$. Then, by Definition 12 (3) we conclude that $A_j = A_{j+1} = A$ for some $A \in \Sigma$. This means that $w = uAA^{-1}u'$ for some $u, u' \in \bar{\Sigma}^*$. It is easy to see that the word uu' constitutes a q-walk. And it is shorter than w. So, by the hypothesis, we have $uu' \xrightarrow{*}_{+/-} q$. And of course there is also $w \xrightarrow{}_{+/-} uu'$, so we get $w \xrightarrow{*}_{+/-} q$. $\qquad ^*\square$

10.2 Proof of Lemma 22

(1) If $u \xrightarrow{}_{+/-} u'$ then there are $w, w' \in \bar{\Sigma}^*$ and $R \in \Sigma$ such that $u = wRR^{-1}w'$ and $u' = ww'$. Then, using Lemma 21:

$$H_u = H_{wRR^{-1}w'} = H_{w'}H_R^{-1}H_RH_w =$$
$$= H_{w'}\overline{h_R}^{-1}\overline{h_R}H_w \subseteq H_{w'}IH_w = H_{ww'} = H_{u'}$$

(2) If $u \xrightarrow{}_{-/+} u'$ then there are $w, w' \in \bar{\Sigma}^*$ and $R \in \Sigma$ such that $u = wR^{-1}Rw'$ and $u' = ww'$. Then, again using Lemma 21:

$$H_u = H_{wR^{-1}Rw'} = H_{w'}H_RH_R^{-1}H_w =$$
$$= H_{w'}\overline{h_R}\overline{h_R}^{-1}WH_w \supseteq H_{w'}IH_w = H_{ww'} = H_{u'} \qquad \square$$

11 APPENDIX D. PROOFS OF SOME LEMMAS NEEDED FOR THEOREM 3.

11.1 Proof of Lemma 31 ($\Leftarrow$).

Let D, D' be some structures such that $v(D) = v(D')$ for each $v \in V_0$. We need to show that $q(D) = q(D')$. There are two cases:

• **Case 1:** $\exists v \in V \; v(D) = 0$.

Then of course also $v(D') = 0$. By Observation 26 this implies that $q(D) = 0$. Likewise we get that $q(D') = 0$, so $q(D) = q(D')$.

• **Case 2:** $\forall v \in V \; v(D) \neq 0$.

Take $\alpha_1, ..., \alpha_k \in \mathbb{R}$ such that $\vec{q} = \sum_{i=1}^{|V|} \alpha \vec{v}_i$.

$$q(D) = \prod_{i=1}^{k} w_i(D)^{\vec{q}(i)} \qquad \text{(by Observation 30)}$$
$$= \prod_{i=1}^{k} w_i(D)^{\sum_{j=1}^{|V|} \alpha_j \vec{v}_j(i)}$$
$$= \prod_{i=1}^{k} \prod_{j=1}^{|V|} \left(w_i(D)^{\vec{v}_j(i)} \right)^{\alpha_j}$$
$$= \prod_{j=1}^{|V|} \left(\prod_{i=1}^{k} w_i(D)^{\vec{v}_j(i)} \right)^{\alpha_j}$$
$$= \prod_{j=1}^{|V|} v_j(D)^{\alpha_j} \qquad \text{(by Observation 30 again)}$$

Note that since for every $j \in \{1, ..., k\}$ we have $v_j(D) > 0$, the expression $\prod_{j=1}^{|V|} v_j(D)^{\alpha_j}$ is correct, even if for some j the number α_j is not natural.

Likewise, we show that $q(D') = \prod_{j=1}^{|V|} v_j(D')^{\alpha_j}$. However, we know that for $j \in \{1, ..., k\}$ it holds that $v_j(D) = v_j(D')$, so this implies that $q(D') = \prod_{j=1}^{|V|} v_j(D)^{\alpha_j} = q(D)$. $\qquad \square$

11.2 Proof of Observation 45

By Lemma 4 we have:

$$|hom(w, s^{(2)})| = |hom(w, \sum_{i=1}^{m} T^i s_i^{(1)})| = \sum_{i=1}^{m} T^i |hom(w, s_i^{(1)})|$$

Likewise $|hom(w', s^{(2)})| = \sum_{i=1}^{m} T^i |hom(w', s_i^{(1)})|$.

Notice that this means that the sequence:

$$|hom(w, s_m^{(1)})|; |hom(w, s_{m-1}^{(1)})|; \ldots |hom(w, s_1^{(1)})|; 0$$

is a representation[28] of $|hom(w, s^{(2)})|$ in radix T. And:

$$|hom(w', s_m^{(1)})|; |hom(w', s_{m-1}^{(1)})|; \ldots |hom(w', s_1^{(1)})|; 0$$

is a representation of $|hom(w', s^{(2)})|$ in radix T. The two representations are different since $|hom(w, s_i^{(1)})| \neq |hom(w', s_i^{(1)})|$ for some i (by Step 1). So the two numbers must be different too. $\qquad \square$

11.3 Proof of Lemma 46

Take any $\alpha_1, ..., \alpha_k \in \mathbb{R}$ such that for all $i \in \{1, \ldots, k\}$, $\sum_{j=1}^{k} \alpha_j A(i, j) = 0$. We will show that then $\alpha_1 = \cdots = \alpha_k = 0$. Let us define a polynomial $P(X) = \alpha_1 + \alpha_2 X + ... + \alpha_k X^{k-1}$. Then $\sum_{j=1}^{k} \alpha_j A(i, j) = \sum_{j=1}^{k} \alpha_j a_i^{j-1} = P(a_i)$. Because $a_1, ..., a_k$ are pairwise distinct, we know that P has at least k zeros. But the degree of P is at most $k-1$, hence $P = 0$ and all of its coefficients are 0. $\qquad \square$

11.4 Proof of Observation 49

(1)

$$(\vec{u} \circ \vec{v}) \;\sigma^{\!\!\!*}\; \vec{w} = \prod_{i=1}^{k} (\vec{u}(i)\vec{v}(i))^{\vec{w}(i)}$$
$$= \prod_{i=1}^{k} (\vec{u}(i))^{\vec{w}(i)} \prod_{i=1}^{k} (\vec{v}(i))^{\vec{w}(i)}$$
$$= (\vec{u} \;\sigma^{\!\!\!*}\; \vec{w})(\vec{v} \;\sigma^{\!\!\!*}\; \vec{w})$$

(2)

$$t^{\vec{u}} \;\sigma^{\!\!\!*}\; \vec{v} = \prod_{i=1}^{k} \left(t^{\vec{u}(i)} \right)^{\vec{v}(i)} = \prod_{i=1}^{k} t^{\vec{u}(i)\vec{v}(i)}$$
$$= t^{\sum_{i=1}^{k} \vec{u}(i)\vec{v}(i)} = t^{\langle \vec{u}, \vec{v} \rangle}$$

---

[28]Recall that each of $|hom(w, s_i^{(1)})|$ is smaller than T.

11.5 Proof of Lemma 50

$$(M\vec{s})(i) = \sum_{j=1}^{k} M(i,j)\vec{s}(j)$$

$$= \sum_{j=1}^{k} w_i(s_j)\vec{s}(j) \qquad \text{(by definition of } M\text{)}$$

$$= w_i\left(\sum_{j=1}^{k} \vec{s}(j)s_j\right) \qquad \text{(by Lemma 4)}$$

$$= w_i(s) \qquad \text{(by definition of } \vec{s}\text{)}$$

$$(M\vec{s}) \, \female \, \vec{v} = \prod_{i=1}^{k} (M\vec{s})(i)^{\vec{v}(i)}$$

$$= \prod_{i=1}^{k} w_i(s)^{\vec{v}(i)}$$

$$= \left(\sum_{i=1}^{k} \vec{v}(i)w_i\right)(s) \qquad \text{(by Lemma 4)}$$

$$\square$$