University of Amsterdam

UvA-DARE (Digital Academic Repository)

Containment of acyclic conjunctive queries with negated atoms or arithmetic comparisons

Sherkhonov, E.; Marx, M.

DOI
10.1016/j.ipl.2016.12.005

Publication date
2017

Document Version
Final published version

Published in
Information Processing Letters

License
Article 25fa Dutch Copyright Act (https://www.openaccess.nl/en/in-the-netherlands/you-share-we-take-care)

Link to publication

Citation for published version (APA):
Sherkhonov, E., & Marx, M. (2017). Containment of acyclic conjunctive queries with negated atoms or arithmetic comparisons. *Information Processing Letters*, *120*, 30-39. https://doi.org/10.1016/j.ipl.2016.12.005

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:20 Jun 2024

Information Processing Letters 120 (2017) 30–39

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Containment of acyclic conjunctive queries with negated atoms or arithmetic comparisons

Evgeny Sherkhonov [a,*,1], Maarten Marx [b]

[a] University of Oxford, Parks Road, OX1 3QD, Oxford, UK
[b] University of Amsterdam, Science Park 904, 1098XH, Amsterdam, Netherlands

ARTICLE INFO

Article history:
Received 17 June 2015
Received in revised form 2 November 2016
Accepted 16 December 2016
Available online 23 December 2016
Communicated by Jef Wijsen

Keywords:
Databases
Query containment
Conjunctive query

ABSTRACT

We study the containment problem for conjunctive queries (CQ*s*) expanded with negated atoms or arithmetic comparisons. It is known that the problem is Π_2^P-complete [14,16]. The aim of this article is to find restrictions on CQ*s* that allow for tractable containment. In particular, we consider acyclic conjunctive queries. Even with the most restrictive form of acyclicity (Berge-acyclicity), containment is coNP-hard. But for a particular fragment of Berge-acyclic CQ*s* with negated atoms or arithmetic comparisons —child-only tree patterns— containment is solvable in PTime.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

We revisit the containment problem for conjunctive queries, one of the classical fundamental problems in database theory. Conjunctive queries (CQ*s*) correspond to select-from-where SQL queries, a class of most frequent queries used in practice. The containment problem is to decide, given two conjunctive queries Q_1 and Q_2, whether, over every database, the answers of Q_1 are contained in the answers of Q_2. A well-known result of Chandra and Merlin is NP-completeness of the containment problem for CQ*s* [4]. Because of relevance to practice, there have been a number of papers dedicated to finding syntactic restrictions on CQ*s* allowing polynomial-time algorithms for containment. *Acyclic* conjunctive queries have been studied as one of the restrictions [17,8].

Conjunctive queries expanded with negated atoms or arithmetic comparisons are used in practice as well. The containment problem is harder for these classes than for CQ*s* – Π_2^P-complete [14,10,15]. There has been little work on finding fragments of CQ*s* with negated atoms or comparisons that have tractable query containment. Even the restriction of acyclicity for CQ*s* has not been considered in presence of negated atoms or arithmetic comparisons. Indeed, acyclicity is a restriction on CQ*s* that allows polynomial-time containment and, furthermore, the known Π_2^P-lower bounds proofs (both in presence of negated atoms and comparisons) involve cyclic queries.

In this article we show that in some cases acyclicity does make containment easier. We show a coNP upper bound for containment of acyclic conjunctive queries with negated atoms of bounded arity. Moreover, we show that containment for acyclic conjunctive queries with arithmetic comparisons of the form $x \text{ op } c$, where x is a variable, c a constant and op a comparison operator from $\{=, \neq, <, >, \leq, \geq\}$, is also solvable in coNP. We obtain several coNP-hardness results for containment of acyclic CQ*s* with negated atoms or comparisons. These lower bounds indicate that the usual notions of acyclicity are not sufficient to obtain tractability, even with the most restrictive form of acyclicity – Berge acyclicity [7]. On a positive side we show that containment for a particular fragment

[*] Corresponding author at: Wolfson Building, Parks Road, OX1 3QD, Oxford, UK.
E-mail addresses: evgeny.sherkhonov@cs.ox.ac.uk (E. Sherkhonov), maartenmarx@uva.nl (M. Marx).

[1] This work was done while the author was at University of Amsterdam.

http://dx.doi.org/10.1016/j.ipl.2016.12.005
0020-0190/© 2016 Elsevier B.V. All rights reserved.

Table 1
Complexity of the containment problem: known results and the results of this article. Here ¬ denotes presence of negated atoms and ACQ*s* denotes α-acyclic CQ*s*.

Class	Complexity
CQ*s* w. ¬	Π_2^P-c [14,16]
CQ*s* w. comparisons	Π_2^P-c [10,15]
ACQ*s* w. ¬, ACQ*s* w. comparisons	coNP-c (Theorem 2, Corollary 1)
Child-only tree patterns w. ¬	PTime (Corollary 2)
Child-only tree patterns w. comparisons	PTime (Corollary 3)

of Berge-acyclic conjunctive queries with negated atoms, namely child-only tree patterns, is decidable in PTime. We extend this PTime result to the case with arithmetic comparisons. These results are based on the characterization of containment in terms of existence of a homomorphism. The latter can be checked by reducing to the known efficient algorithms for positive acyclic queries [8].

The contributions of this article are summarized in Table 1. In particular,

- We identify a fragment of CQ*s* with negated atoms for which containment is coNP-complete: α-acyclic conjunctive queries with negated atoms of bounded arity. We derive the same bound for α-acyclic CQ*s* with arithmetic comparisons.
- Consider the following three conditions on a conjunctive query $\mathcal{Q}$ with negated atoms (resp. with arithmetic comparisons).
  (i) $\mathcal{Q}$ contains an atom with a constant as an argument,
  (ii) $\mathcal{Q}$ is connected,
  (iii) $\mathcal{Q}$ is Berge-acyclic.
  For every class of CQ*s* with negated atoms (arithmetic comparisons) satisfying at most two of the conditions (i)–(iii), containment is coNP-hard.
- Although we could not show that CQ*s* with negated atoms or comparisons satisfying all of (i)–(iii) have a PTime containment problem, we could do that for an even further restricted case: CQ*s* corresponding to XML tree patterns with multiple labels on the nodes [11]. If these tree patterns only contain either child or the descendant edges, their expansions with negated labels or arithmetic comparisons have a PTime containment problem.

Related work. For (positive) conjunctive queries, containment and evaluation problems are equivalent. The PTime result for evaluation of α-acyclic CQ*s* from [17] implies PTime result for containment. Gottlob et al. [8] proved that in fact evaluation (and thus containment) is complete for LOGCFL, the class of problems that are logspace reducible to a context-free language. This class of problems allows for efficient parallelizable algorithms. Since then there have been a number of papers on generalizing the acyclicity condition while keeping the evaluation and containment problems tractable. Chekuri and Rajaraman [5] introduced the notion of query width and proved that containment for CQ*s* with bounded query width is in PTime. The class of α-acyclic queries is exactly the class of queries with a query width of 1. Later, Gottlob et al. [9] introduced the notion of hypertree-width. They showed that CQ*s* of bounded hypertree-width can also be evaluated efficiently, and, moreover, this class strictly generalizes the class of queries with bounded query width.

Containment for CQ*s* expanded with negated atoms and arithmetic comparisons has been considered in [14,16] and [10,15] respectively. In both cases, containment is Π_2^P-complete. In either of the expansions, the lower bound proofs involved cyclic CQ*s*. There has been little work studying restrictions of CQ*s* (in particular, acyclicity) with negated atoms or arithmetic comparisons that lower the complexity of containment. In [15], van der Meyden considered *monadic* CQ*s* with arithmetic comparisons, which trivially are a fragment of acyclic CQ*s* with comparisons, and argued that containment for this class is solvable in PTime.

Tree pattern containment over trees has received considerable attention as well. Child-only tree patterns are acyclic queries and thus containment is in PTime. In fact, any two-combinations of the child, descendant and the wildcard (empty node label) axes allow PTime containment [2,11]. When all the three axes are allowed, the problem becomes coNP-complete [11]. In case label negation is added to tree patterns, containment becomes PSpace-complete [6].

Containment for tree patterns expanded with attribute value comparisons has also been studied in the past. Attribute value comparisons are specific to XML documents (trees), where each node can have a number of associated attribute values. In [1] it has been shown that containment for this fragment is Π_2^P-complete. Notably, the lower bound used a reduction from containment of CQ*s* with arithmetic comparisons, and used the construct $@_a X = @_b Y$ that allows to compare attributes of two distinct nodes. In [13] it has been shown that if only constructs of the form $@_a \mathrm{op}\ c$ ($\mathrm{op} \in \{=, \neq, <, >, \leq, \geq\}$), i.e., comparison with a *constant* only, are allowed, then containment remains in coNP.

Overview. Section 2 recalls the needed concepts and notation. Section 3 is about our coNP completeness results. Section 4 contains the PTime results for the expanded tree patterns. We end with conclusions, open problems and future work.

2. Preliminaries

A relational schema **S** is a set of relational names with associated arities. We assume countably infinite disjoint sets of variables and constants **Var** and **Const**. A *term* is an element from **Var** ∪ **Const**. We also assume a dense linear order $<$ on **Const**. For tuples of terms $\bar{x}$ and $\bar{y}$, by $\bar{x} \subseteq \bar{y}$ we denote the fact that every element of $\bar{x}$ is an element of $\bar{y}$. An *instance* I over **S** is a set of *facts* of the form $R(a_1, \ldots, a_n)$, where $R \in \mathbf{S}$ is a relational name of arity n and each $a_i \in \mathbf{Const}$. By $dom(I)$ we denote the domain of I, i.e., the constants appearing in I. A *positive atom* (or just an *atom*) and a *negated atom* are expressions of the form $R(x_1, \ldots, x_n)$ and $\neg R(x_1, \ldots, x_n)$ respectively, where $R \in \mathbf{S}$ is a relational name of arity n and each x_i is a term.

For $k \geq 1$, an expression $\mathcal{Q}(\bar{x}) = \exists\bar{y}\varphi(\bar{x}, \bar{y})$, where $\bar{x}$ is a k-tuple of variables, is a k-ary *conjunctive query* (CQ) if $\varphi(\bar{x}, \bar{y})$ is a conjunction of positive atoms with variables from $\bar{x}$ and $\bar{y}$ only. We say that $\mathcal{Q}(\bar{x}) = \exists\bar{y}\varphi(\bar{x}, \bar{y})$ is a k-ary *conjunctive query with negated atoms* (CQ$^{\neg}$) if $\varphi(\bar{x}, \bar{y})$ is a conjunction of atoms and negated atoms. Likewise, $\mathcal{Q}(\bar{x}) = \exists\bar{y}\varphi(\bar{x}, \bar{y})$ is a k-ary *conjunctive query with arithmetic comparisons* (CQ^{comp}) if $\varphi(\bar{x}, \bar{y})$ is a conjunction of atoms and expressions of the form $x \text{ op } c$, where $\text{op} \in \{=, \neq, <, >, \leq, \geq\}$, $x \in \mathbf{Var}$ is a variable from $\bar{x}$, $\bar{y}$, and $c \in \mathbf{Const}$. Note that we do *not* allow comparisons of the form $x \text{ op } y$, where x and y are both variables. As usual, we stipulate that each variable in $\bar{x}$ occurs in some conjunct of φ. A 0-ary query is called *Boolean*. A CQ$^{\neg}$ query $\mathcal{Q}$ is *consistent* if an atom and its negation do not appear in $\mathcal{Q}$ at the same time. We say that a CQ^{comp} query $\mathcal{Q}$ is *consistent* if the comparisons of $\mathcal{Q}$ are consistent.

For a positive or negated atom P and a conjunctive query with negated atoms or arithmetic comparisons $\mathcal{Q}$, $P \in \mathcal{Q}$ denotes the fact that P is a conjunct of $\mathcal{Q}$. We denote by $Var(\mathcal{Q})$, $Const(\mathcal{Q})$ and $Term(\mathcal{Q})$ the sets of variables, constants and terms occurring in $\mathcal{Q}$. We say that $\mathcal{Q}$ is *connected* if for every pair t and t' of terms in $\mathcal{Q}$, there is a sequence of atoms $P_1, \ldots, P_n$ in $\mathcal{Q}$ such that $t \in Term(P_1)$, $t' \in Term(P_n)$ and $Term(P_i) \cap Term(P_{i+1}) \neq \emptyset$, for every i, $1 \leq i < n$.

The *answer set* of a k-ary CQ$^{\neg}$ query $\mathcal{Q}(\bar{x})$ on an instance I is a k-ary relation $Ans(\mathcal{Q}, I) \subseteq \mathbf{Const}^k$ which consists of all tuples $\theta(\bar{x})$ such that $\theta : Var(\mathcal{Q}) \to dom(I)$ is a substitution with the properties that for every positive atom $R(\bar{u}) \in \mathcal{Q}$ it holds that $R(\theta(\bar{u})) \in I$, and for every negated atom $\neg P(\bar{v}) \in \mathcal{Q}$ it holds that $P(\theta(\bar{v})) \notin I$ (here we assume that θ is identity on $\mathbf{Const}$). The semantics for conjunctive queries with comparisons is defined similarly. Now instead of preserving negation, a substitution θ must preserve the comparisons. That is, if $x \text{ op } c$ is a comparison in $\mathcal{Q}$, then $\theta(x) \text{ op } c$ must hold. For a Boolean query $\mathcal{Q}$, by $I \models \mathcal{Q}$ we denote the fact that $Ans(\mathcal{Q}, I) = \{\langle\rangle\}$. If $I \models \mathcal{Q}$, we refer to θ that witnesses this fact as a *satisfying assignment* for $\mathcal{Q}$ in I.

Let $\mathcal{Q}_1$ and $\mathcal{Q}_2$ be CQ*s* of the same arity (with negated atoms or comparisons). We say that $\mathcal{Q}_1$ is *contained* in $\mathcal{Q}_2$, denoted as $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$, if $Ans(\mathcal{Q}_1, I) \subseteq Ans(\mathcal{Q}_2, I)$ holds for every instance I.

The *containment problem* for a class of conjunctive queries $\mathcal{C}$ consists of deciding, given $\mathcal{Q}_1$ and $\mathcal{Q}_2$ from $\mathcal{C}$, whether $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$.

We follow [8] in the definition of acyclic CQ*s*. Acyclicity is defined using the notion of a hypergraph. A *hypergraph* is a pair $H = (V, E)$, where V is a set of vertices and $E \subseteq 2^V$ a set of hyperedges. Given a hypergraph $H = (V, E)$, the *GYO-reduct*, denoted as $GYO(H)$, is the hypergraph obtained from H by repeatedly applying the following rules in exhaustive manner:

- Remove hyperedges that are empty or contained in other hyperedges,
- Remove vertices that appear in at most one hyperedge.

A hypergraph $H = (V, E)$ is *α-acyclic* if GYO(H) is the empty hypergraph. The *incidence graph* of H is the undirected bipartite graph where $V \cup E$ is the set of vertices and (x, R) is an edge if and only if $x \in R$. We say that H is *Berge-acyclic* if its incidence graph is acyclic. Note that a Berge-acyclic hypergraph is α-acyclic, but not vice versa.

The hypergraph $H(\mathcal{Q}) = (V, E)$ of a CQ$^{\neg}$ query $\mathcal{Q}$ is defined as follows. The set of vertices $V = Var(\mathcal{Q})$, and for each atom $R(\bar{x})$ or a negated atom $\neg P(\bar{x})$ in $\mathcal{Q}$, the set E contains a hyperedge consisting of all the variables occurring in $\bar{x}$. Then $\mathcal{Q}$ is *α-acyclic* (ACQ$^{\neg}$) (resp. Berge-acyclic), if $H(\mathcal{Q})$ is α-acyclic (resp. Berge-acyclic). A CQ^{comp} query is *α-acyclic* (ACQ^{comp}) (resp. Berge-acyclic) if its “relational part”, i.e., the CQ obtained by removing the comparisons, is α-acyclic (resp. Berge-acyclic). Here we assume that if the comparisons of a query entail that $x = c$, then every occurrence of x in the query is replaced by c.

Next we give the definition of tree patterns. Essentially, they are tree patterns from [11] where nodes can have multiple positive and negative labels or attribute comparisons. As we will see, tree patterns containing only the child relation can be considered as a fragment of Berge-acyclic CQ*s*. Let Σ be a set of node labels. A *tree pattern with label negation* P is a node-labeled tree $(N, E, E_{//}, r, l^+, l^-)$, where N is the set of nodes, $E \cup E_{//} \subseteq N^2$ is the edge relation consisting of disjoint child and descendant relations respectively, $r \in N$ is the root, and $l^+, l^- : N \to 2^\Sigma$ are positive and negative node labeling functions. Let additionally A be a set of attribute names. By Σ_A we denote the set $\{@_a\text{op } c \mid a \in A, \text{op} \in \{=, \neq, <, >, \leq, \geq\}, c \in \mathbf{Const}\}$. A *tree pattern with attribute comparisons* is a node-labeled tree $(N, E, E_{//}, r, l)$, such that N, E, $E_{//}$, r are as above, and $l : N \to 2^{\Sigma \cup \Sigma_A}$ is a node labeling function. For a tree pattern P, by $Nodes(P)$ we denote the set of nodes of P.

We define semantics of tree patterns as follows. Let $G = (dom(G), E', r', \rho)$ be a graph, where $dom(G)$ is the set of nodes, $E' \subseteq dom(G)^2$ the edge relation, $\rho : dom(G) \to 2^\Sigma$ is a node labeling function, and $r' \in dom(G)$ is a fixed designated node. We say that a tree pattern with label negation P is *true* in G, or G *satisfies* P, denoted as $G \models P$, if there is a function $e : N \to dom(G)$, called *embedding* of P in G, such that all of the following hold:

(1) if $(x, y) \in E$, then $(e(x), e(y)) \in E'$,
(2) if $(x, y) \in E_{//}$, then $(e(x), e(y)) \in E'^+$, where E'^+ is the transitive closure of E',
(3) for every $x \in N$, $l^+(x) \subseteq \rho(e(x))$,
(4) for every $x \in N$, $l^-(x) \cap \rho(e(x)) = \emptyset$.

We write $G \models_{root} P$ if there is an embedding e of P in G that additionally satisfies the following condition:

(0) $e(r) = r'$.

Semantics of tree patterns with attribute comparisons is defined over graphs which are additionally equipped with a partial function $att : dom(G) \times A \to \mathbf{Const}$. The definition of $G \models P$ and $G \models_{root} P$, where P is a tree pattern with attribute comparisons, is defined similarly to the above definition, where (3) and (4) are replaced with the following conditions:

(3') For every $x \in N$, $l(x) \cap \Sigma \subseteq \rho(e(x))$,
(4') For every $x \in N$, if $@_a \mathrm{op}\ c \in l(x)$, then $att(e(x), a)$ is defined and $att(e(x), a)\ \mathrm{op}\ c$.

We say that a tree pattern with label negation $P = (N, E, E_{//}, r, l^+, l^-)$ is *consistent* if $l^+(x) \cap l^-(x) = \emptyset$ holds for every $x \in N$. Similarly, a tree pattern with attribute comparisons is consistent if the comparisons of every attribute in every node are consistent. Note that a tree pattern P is consistent if and only if there is a graph G which satisfies P. Furthermore, consistency of a tree pattern can be checked in PTime. For tree patterns with label negation or comparisons P and Q, we say that P is *contained* in Q (resp. *root-to-root contained*), denoted as $P \subseteq Q$ ($P \subseteq_{root} Q$), if for every G it holds that $G \models P$ ($G \models_{root} P$) implies $G \models Q$ ($G \models_{root} Q$).

We say that a tree pattern is *child-only* if the set $E_{//}$ is empty, and *descendant-only* if E is empty. In these cases we omit the relations $E_{//}$ and E respectively. By a *canonical tree* for a consistent child-only tree pattern $P = (N, E, r, l^+, l^-)$ we mean the tree $T_P = (N, E, r, l^+)$. Obviously P is satisfied at the root of its canonical tree, i.e., $T_P \models_{root} P$.

The containment and root-to-root containment problems for child-only tree patterns with label negation (with attribute comparisons) can be reduced to containment for Boolean CQ$^{\neg}$s (CQ^{comp}s) (cf. [3]) with the following two translations. For a child-only tree pattern with label negation $P = (N, E, r, l^+, l^-)$, we define TR(P) as

$$\bigwedge_{v \in N, p \in l^+(v)} p(v) \wedge \bigwedge_{u \in N, q \in l^-(u)} \neg q(u) \wedge \bigwedge_{\langle u,v \rangle \in E} E(u, v),$$

where every element in N is an existentially quantified variable. The result of translation TR$_r(P)$ is defined as TR(P) with the only difference that r is now a constant. For a child-only tree pattern with attribute comparisons $P = (N, E, r, l)$, translations TR(P) and TR$_r(P)$ are defined similarly:

$$\bigwedge_{v \in N, p \in l(v)} p(v) \wedge \bigwedge_{u \in N, @_a \mathrm{op}\ c \in l(u)} (a(u, x_{u,a}) \wedge x_{u,a}\ \mathrm{op}\ c)$$
$$\wedge \bigwedge_{\langle u,v \rangle \in E} E(u, v).$$

Note that the result of translation of a tree pattern with label negation (att. comparisons) is a connected Boolean Berge-acyclic CQ with unary negated atoms (comparisons). Moreover, negation in this query is *guarded* (i.e., when for every negated atom $\neg R(\bar{x})$ in a query there is an atom $P(\bar{y})$ in the query such that every variable in $\bar{x}$ occurs in $\bar{y}$).

Proposition 1. *Let P_1 and P_2 be child-only tree patterns with label negation (attribute comparisons). Then*

(i) $P_1 \subseteq P_2$ *iff* TR$(P_1) \subseteq$ TR(P_2),
(ii) $P_1 \subseteq_{root} P_2$ *iff* TR$_r(P_1) \subseteq$ TR$_r(P_2)$.

We will use item (i) of the above proposition to derive lower bounds in Section 3. In Section 4 we show that root-to-root containment for child-only tree patterns with label negation or comparisons is in PTime, implying the same upper bound for the fragment of acyclic CQs corresponding to the translation of child-only tree patterns, by item (ii).

3. Containment for acyclic conjunctive queries with negated atoms or comparisons

We first state the known result on the containment for CQs with negated atoms or comparisons.

Theorem 1. [10,15,12] *The containment problem for* CQ$^{\neg}$ *and* CQ^{comp} *queries is* Π_2^P*-complete.*

As noted in the Introduction, the known proofs for the Π_2^P lower bound involve conjunctive queries that are *cyclic*. In this section we show that containment for α*-acyclic* conjunctive queries with negated atoms of bounded arity (comparisons) is coNP-complete. We also provide several coNP lower bounds which help to identify the sources of intractability. Without loss of generality we can consider containment for *Boolean* acyclic CQs. Indeed, the containment problem for non-Boolean CQs can be reduced in PTime to containment of Boolean CQs while preserving the acyclicity restriction.

Proposition 2. *Let $\mathcal{P}$ and $\mathcal{Q}$ be* ACQ$^{\neg}$*s* (ACQ^{comp}*s*). *Then there exist* PTime *computable Boolean* ACQ$^{\neg}$*s (Boolean* ACQ^{comp}*s) $\mathcal{P}'$ and $\mathcal{Q}'$ such that*

$\mathcal{P} \subseteq \mathcal{Q}$ iff $\mathcal{P}' \subseteq \mathcal{Q}'$.

This also holds for Berge-acyclic queries.

Proof. Let $\mathcal{P}(\bar{x})$ and $\mathcal{Q}(\bar{y})$ be ACQ$^{\neg}$*s* (ACQ^{comp}*s*). We check if $\mathcal{P}$ and $\mathcal{Q}$ are consistent which can be done in PTime [16,15]. If $\mathcal{P}$ is not consistent, let $\mathcal{P}' = \mathcal{Q}' = \exists x.P(x)$. Otherwise, if $\mathcal{Q}$ is not consistent or the length of $\bar{x}$ and the length $\bar{y}$ are different, let $\mathcal{P}' = \exists x.P_1(x)$ and $\mathcal{Q}' = \exists x.P_2(x)$ for $P_1 \neq P_2$. Let $\mathcal{P}$ and $\mathcal{Q}$ now be consistent, $\bar{x} = (x_1, \ldots, x_n)$ and $\bar{y} = (y_1, \ldots, y_n)$, and $P_1, \ldots, P_n$ unary relational names that do not appear in $\mathcal{P}$ or $\mathcal{Q}$. Then we define $\mathcal{P}' = \exists \bar{x}.P_1(x_1) \wedge \ldots \wedge P_n(x_n) \wedge \mathcal{P}(\bar{x})$ and $\mathcal{Q}' = \exists \bar{y}.P_1(y_1) \wedge \ldots \wedge P_n(y_n) \wedge \mathcal{Q}(\bar{y})$. Clearly, $\mathcal{P}'$ and $\mathcal{Q}'$ are PTime computable. Moreover, if $\mathcal{P}$ and $\mathcal{Q}$ are p-acyclic ($p \in \{\alpha, \text{Berge}\}$), then $\mathcal{P}'$ and $\mathcal{Q}'$ are p-acyclic as well. It is straightforward to show that $\mathcal{P} \subseteq \mathcal{Q}$ iff $\mathcal{P}' \subseteq \mathcal{Q}'$. □

Thus, in the rest of the article we only consider containment for *Boolean* acyclic CQs with negated atoms or comparisons.

Now we show that restricting CQs with negated atoms to be acyclic and with negated atoms of bounded arity makes the containment problem coNP-complete.

Theorem 2. *The containment problem for α-acyclic conjunctive queries with negated atoms of bounded arity (or with arithmetic comparisons), is in* coNP.

Proof. Let $\mathcal{P}$ and $\mathcal{Q}$ be input queries. A coNP algorithm then works as follows. We first guess a potential counterexample I, and, second, check whether $I \models \mathcal{P}$ and

$I \not\models \mathcal{Q}$. Lemma 1 below guarantees that it is enough to guess a counterexample of size polynomial in the sizes of $\mathcal{P}$ and $\mathcal{Q}$. By Lemma 2 below, the second step can be done in PTime. The acyclicity condition is not used in the proof of Lemma 1, but it is crucial in the proof of Lemma 2. □

Lemma 1. *Let $\mathcal{Q}_1$ and $\mathcal{Q}_2$ be Boolean α-acyclic* CQ*s with negated atoms of bounded arity (resp. with arithmetic comparisons). If $\mathcal{Q}_1 \not\subseteq \mathcal{Q}_2$, then there is an instance I such that $I \models \mathcal{Q}_1$, $I \not\models \mathcal{Q}_2$, and the size of I is polynomial in the sizes of $\mathcal{Q}_1$, $\mathcal{Q}_2$.*

Proof. We first consider the case of CQ*s* with negated atoms. By the assumption, for every negated atom $\neg R$ in $\mathcal{Q}_2$, the arity of R is bounded by a constant k. Let I' be a counterexample for $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$. Since $I' \models \mathcal{Q}_1$, there is a satisfying assignment $\theta : Var(\mathcal{Q}_1) \to dom(I')$. By $\theta(Var(\mathcal{Q}_1))$ we denote the range of θ. Furthermore, by $\theta(\mathcal{Q}_1)$ we denote the image of positive atoms in $\mathcal{Q}_1$ wrt θ, i.e., the set $\{R(\theta(\bar{x})) \mid R(\bar{x}) \in \mathcal{Q}_1\}$. We then define the instance I as the set

$$\theta(\mathcal{Q}_1) \cup \{N(\theta(y_1), \ldots, \theta(y_m))\} \cup$$
$$\cup \{P(\bar{a}) \in I' \mid P \text{ occurs negatively in } \mathcal{Q}_2,$$
$$\bar{a} \subseteq \theta(Var(\mathcal{Q}_1)) \cup Const(\mathcal{Q}_1) \cup Const(\mathcal{Q}_2)\},$$

where N is a fresh relational name and $\bar{y} = y_1, \ldots, y_m$ are the variables of $\mathcal{Q}_1$ that appear in a negated atom but not in a positive atom in $\mathcal{Q}_1$. We add the N-fact to I in order to retain the image of the "unsafe" variables appearing in a negated atom in $\mathcal{Q}_1$.

Note that the size of I is bounded by

$$|\mathcal{Q}_1| + |Var(Q_1)| + |\mathcal{Q}_2| \cdot (|Term(\mathcal{Q}_1)| + |Const(\mathcal{Q}_2)|)^k.$$

Firstly, θ is a satisfying assignment for $\mathcal{Q}_1$ in I. Indeed, the positive atoms are preserved since $\theta(\mathcal{Q}_1) \subseteq I$. Furthermore, no negated atom in $\mathcal{Q}_1$ becomes true under θ: if $\neg R(\bar{x}) \in \mathcal{Q}_1$, then $\theta(\bar{x}) \in dom(I)^{|\bar{x}|}$ (since every $\theta(x_i)$ is either in a fact from $\theta(\mathcal{Q}_1)$ or in $N(\theta(\bar{y}))$) and $R(\theta(\bar{x})) \notin I$ (since $R(\theta(\bar{x})) \notin I'$ and $R \neq N$).

Secondly, we show $I \not\models \mathcal{Q}_2$. Suppose the opposite. This means there is a satisfying assignment $h : Var(\mathcal{Q}_2) \to dom(I)$. We show that h is also a satisfying assignment for $\mathcal{Q}_2$ in I' which contradicts the assumption.

- Let $R(\bar{x}) \in \mathcal{Q}_2$. Then $R(h(\bar{x})) \in I \setminus \{N(\theta(\bar{y}))\} \subseteq I'$.
- Let $\neg R(\bar{x}) \in \mathcal{Q}_2$. Then $R(h(\bar{x})) \notin I$. Note that $h(\bar{x}) \subseteq \theta(Var(\mathcal{Q}_1)) \cup Const(\mathcal{Q}_1) \cup Const(\mathcal{Q}_2)$. Thus, because of that and the fact that R occurs negatively in $\mathcal{Q}_2$, it follows that $R(\bar{a}) \notin I'$ by the definition of I.

We now prove the lemma for the case of arithmetic comparisons. Let θ be a satisfying assignment for $\mathcal{Q}_1$ in I'. We take I as $\theta(\mathcal{Q}_1) = \{R(\theta(\bar{x})) \mid R(\bar{x}) \in \mathcal{Q}_1\}$. The size of I is obviously polynomial. $I \models \mathcal{Q}_1$ holds because θ is a satisfying assignment for $\mathcal{Q}_1$ in I. Furthermore, $I \not\models \mathcal{Q}_2$ holds since any satisfying assignment for $\mathcal{Q}_2$ in I is a satisfying assignment for $\mathcal{Q}_2$ in I'. □

The *evaluation problem* for a class of Boolean queries $\mathcal{C}$ is the following decision problem. Given an instance I, a Boolean query $\mathcal{Q} \in \mathcal{C}$, decide whether $\mathcal{Q}$ evaluates to true in I, i.e., $I \models \mathcal{Q}$.

Lemma 2. *The evaluation problem is in* PTime *for each of the following classes of Boolean queries:*

(i) Boolean α-acyclic conjunctive queries with negated atoms of bounded arity, and
(ii) Boolean α-acyclic conjunctive queries with arithmetic comparisons.

Proof. We prove item (i). Let I be an instance and $\mathcal{Q}$ a Boolean α-acyclic CQ where each negated atom is bounded by a constant k. We make a polynomial reduction to the evaluation problem for (positive) α-acyclic Boolean CQ*s* which is known to be in PTime [17,8].

For every relational name R that occurs negatively in $\mathcal{Q}$, we introduce a new relational name $\widetilde{R}$ of the same arity as R. By $\widetilde{\mathcal{Q}}$ we denote the result of replacement of each $\neg R(\bar{x})$ in $\mathcal{Q}$ by $\widetilde{R}(\bar{x})$. Note that $\widetilde{\mathcal{Q}}$ is now an ordinary CQ. Moreover, $\widetilde{\mathcal{Q}}$ is α-acyclic because $\mathcal{Q}$ is α-acyclic. We then define the instance

$$\widetilde{I} = I \cup \{\widetilde{R}(\bar{a}) \mid \bar{a} \subseteq dom(I), \neg R(\bar{x}) \in \mathcal{Q}, \text{ and } R(\bar{a}) \notin I\}.$$

Note that the size of $\widetilde{I}$ is bounded by $|I| + |\mathcal{Q}| \cdot |dom(I)|^k$ which is polynomial in the sizes of I and $\mathcal{Q}$. We claim that $I \models \mathcal{Q}$ if and only if $\widetilde{I} \models \widetilde{\mathcal{Q}}$.

($\Rightarrow$). Suppose $I \models \mathcal{Q}$, i.e., there is a satisfying variable assignment $\theta : Var(\mathcal{Q}) \to dom(I)$. Note that $Var(\mathcal{Q}) = Var(\widetilde{\mathcal{Q}})$ and $dom(I) = dom(\widetilde{I})$. We show that θ is a satisfying assignment for $\widetilde{\mathcal{Q}}$ in $\widetilde{I}$. The positive atoms from $\mathcal{Q}$ are still preserved since we did not remove any facts from I. Let $\widetilde{R}(\bar{x}) \in \widetilde{\mathcal{Q}}$. This means that $\neg R(\bar{x}) \in \mathcal{Q}$. Hence, $R(\theta(\bar{x})) \notin I$. Since also $\theta(\bar{x}) \subseteq dom(I)$, we have that $R(\theta(\bar{x})) \in \widetilde{I}$, as needed.

($\Leftarrow$). Suppose $\widetilde{I} \models \widetilde{\mathcal{Q}}$, i.e., there is a satisfying assignment $\theta : Var(\widetilde{\mathcal{Q}}) \to dom(\widetilde{I})$. We show that θ is a satisfying assignment for $\mathcal{Q}$ in I. Positive atoms in $\mathcal{Q}$ are preserved since they are positive atoms in $\widetilde{\mathcal{Q}}$ as well and θ preserves them in $\widetilde{I}$ and thus in I. Let $\neg R(\bar{x}) \in \mathcal{Q}$. Then $\widetilde{R}(\bar{x}) \in \widetilde{\mathcal{Q}}$ and $\widetilde{R}(\theta(\bar{x})) \in \widetilde{I}$. Then by definition of $\widetilde{I}$, it follows that $R(\theta(\bar{x})) \notin I$, as desired.

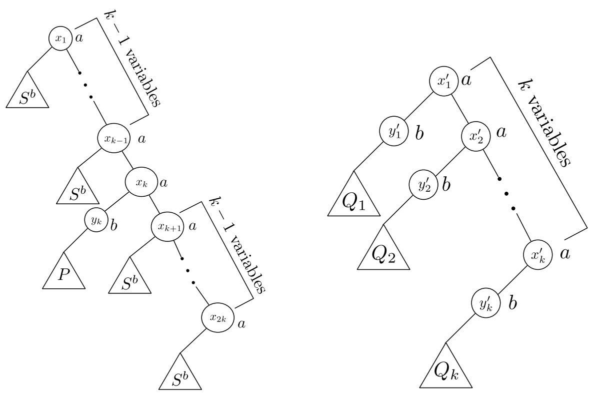
Item (ii) is shown similarly. Now each arithmetic comparison $x \text{ op } c$ that occurs in $\mathcal{Q}$ is replaced with a new unary atom $P_{\text{op } c}(x)$. Let $\widetilde{\mathcal{Q}}$ be the result of this replacement. Let Σ_c be the constants occurring in the comparisons of $\mathcal{Q}$. Note that $|\Sigma_c| \leq |\mathcal{Q}|$. We define the instance

$$\widetilde{I} = I \cup \{P_{\text{op } c}(a) \mid a \in dom(I), c \in \Sigma_c,$$
$$\text{op} \in \{=, \neq, <, >, \leq, \geq\} \text{ and } a \text{ op } c\}.$$

Note that the size of $\widetilde{I}$ is bounded by $|I| + 6 \cdot |\mathcal{Q}| \cdot |dom(I)|$, which is polynomial in the sizes of I and $\mathcal{Q}$. It is straightforward to show that $I \models \mathcal{Q}$ if and only if $\widetilde{I} \models \widetilde{\mathcal{Q}}$. □

Lower bound

We show that the corresponding coNP lower bound for containment already holds for child-only tree patterns with label negation (or att. comparisons). For this, we first



1(a) Query P'.

1(b) Query Q'.

Fig. 1. Queries P' and Q' from Lemma 3.

show that we can allow disjunction on the right hand side query of the containment problem. We extend the definition of containment for *unions* of tree patterns with label negation (att. comparisons), i.e., expressions of the form $\bigvee_{i=1}^{k} Q_i$ where each Q_i is a tree pattern with label negation (att. comparisons). Let P be a tree pattern and $Q = \bigvee_{i=1}^{k} Q_i$ a union of tree patterns. We say that P is *contained* (resp. *root-to-root contained*) in Q if for every G it holds that $G \models P$ ($G \models_{root} P$) implies that there is a $j \in \{1, \dots, k\}$ such that $G \models Q_j$ ($G \models_{root} Q_j$).

Lemma 3. *Let P be a child-only tree pattern with label negation (resp. attribute comparisons) and $Q = \bigvee_{i=1}^{k} Q_i$ a union of child-only tree patterns with label negation (attribute comparisons). There exist* PTIME *computable child-only tree patterns with label negation (attribute comparisons) P' and Q' such that*

$$P \subseteq_{root} Q \text{ if and only if } P' \subseteq Q'.$$

Proof. The proof is similar to the one of Lemma 3 in [11]. Let a, b be node labels not occurring in P or Q. Let S^b be the child-only tree pattern corresponding to (written as a CQ$^{\neg}$) $b(x) \wedge \bigwedge_{i=1}^{k}(E(x, y_i) \wedge \mathrm{TR}(Q_i))$, where y_i is the variable corresponding to the root of Q_i, x is not among the variables of every $\mathrm{TR}(Q_i)$ and E is the child relation. We define P' and Q' as in Fig. 1. In this figure, a circle denotes a node, a triangle denotes the tree pattern written inside the triangle, and a line connecting two circles (or a circle and a triangle) denotes the child relation between the two nodes (resp. between the node and the root of the tree pattern corresponding to the triangle). Clearly P' and Q' are child-only tree patterns with label negation and PTIME computable. We verify that $P \subseteq_{root} Q$ if and only if $P' \subseteq Q'$.

($\Rightarrow$). Assume $P \subseteq_{root} Q$. Let I be a graph such that $I \models P'$. That means there is an embedding $\theta : Nodes(P') \to dom(I)$. Note that θ is also an embedding of P in I as well, since P is a subquery of P'. Thus, $I \models Q$, i.e., there exists an index j, $1 \le j \le k$, such that $I \models Q_j$. The latter implies there is an embedding h' of Q_j in I. Moreover, since the containment of P in Q is root-to-root, we must have that h' maps the root of Q_j to the θ-image of the root of P. Then we define a mapping $\theta' : Nodes(Q') \to I$ as the composition of h with θ, where h is a mapping from $Nodes(Q')$ to $Nodes(P')$ that extends h' and is defined for the other nodes as follows. For every i, $1 \le i \le k$, we define $h(x'_i) := x_{k-j+i}$, $h(y'_i) = y_{k-j+i}$ and the nodes of Q_i ($i \neq j$) in Q' are mapped “canonically” to the corresponding nodes in S^b. It is easy to see that θ' is indeed an embedding of Q' in I.

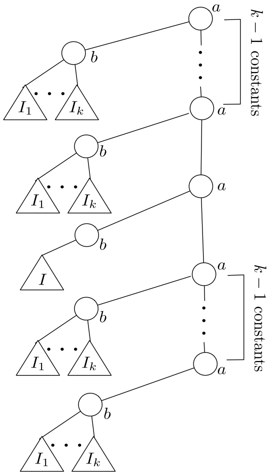


Fig. 2. The instance I' from Lemma 3.

($\Leftarrow$). Assume $P' \subseteq Q'$. Let I be an instance such that $I \models P$ and h an embedding of P in I. Let also I_i be the canonical tree (instance) of Q_i (i.e., the instance containing only the positive atoms of Q_i and replacing each variable by a fresh constant). Then we construct the instance I' depicted in Fig. 2. Note that I is connected to the b-node via the h-image of the root of P. By the assumption, it holds that $I' \models Q'$, i.e., there is an embedding $\theta : Nodes(Q') \to dom(I')$. In particular, since a only appears in the vertical span of $2k - 1$ nodes in Fig. 2, the span of k a-nodes of Q' can only be mapped on that vertical span in I'. Because of this and the fact that a b-node in Q' must be mapped to a b-node in I', there must exist an index j for which θ is an embedding of Q_j in I. Moreover, this embedding maps the root of Q_j to the h-image of the root of P. Thus, $I \models_{root} Q_j$ and, therefore, $I \models_{root} Q$.

The case of child-only tree patterns with attribute comparisons is proved similarly. □

Lemma 4. *The containment problem $P \subseteq_{root} \bigvee_{i=1}^{m} Q_i$ is* coNP*-hard for each of the following cases:*

(i) P, Q_i are child-only tree patterns with label negation,
(ii) P, Q_i are descendant-only tree patterns with label negation,
(iii) P, Q_i are child-only tree patterns with attribute comparisons.

Proof. (i). We reduce 3SAT to the complement of the containment problem for the stated fragment. Let φ be a conjunction of clauses $C_i = (b_1^i \vee b_2^i \vee b_3^i)$, $1 \le i \le k$, over the variables $\{x_1, \dots, x_n\}$, where b_j^i are literals, i.e., variables or their negations. For every clause C_i we introduce a node label c_i. Then we define P and Q_i over the node labels c_i, $1 \leq i \leq k$, and x_j, $1 \leq j \leq n$, as follows. We define $P = (\{r\}, \emptyset, \emptyset, r, l^+, l^-)$, $l^+(r) = \{c_1, \ldots, c_k\}$ and $l^-(r) = \emptyset$; $Q_i = (\{r_i\}, \emptyset, \emptyset, r_i, l_i^+, l_i^-)$, $c_i \in l^+(r_i)$, $x \in l^-(r_i)$ if $b_j^i = x$ and $x \in l^+(r_i)$ if $b_j^i = \neg x$ for x a variable and every j, $j = 1, 2, 3$.

We claim that φ is satisfiable iff $P \not\subseteq_{root} \bigvee_{i=1}^k Q_i$. Indeed, $P \not\subseteq_{root} \bigvee_{i=1}^k Q_i$ iff there is a graph $G = (dom(G), E', \rho, r')$ such that $G \models_{root} P$ and $G \not\models_{root} Q_i$ for every $i \in 1, \ldots, k$. This is equivalent to respectively the fact that $\{c_1, \ldots, c_k\} \subseteq \rho(r')$ and for every i, $i = 1, \ldots, k$, it holds that $c_i \notin \rho(r')$ or there is a $j \in \{1, 2, 3\}$ such that $x \in \rho(r')$ when $b_j^i = x$ or $x \notin \rho(r')$ when $b_j^i = \neg x$. In turn this is equivalent to the fact that $\rho(r')$ gives rise to a satisfying variable assignment for φ. Note that P and Q_i are also descendant-only tree patterns with label negation, and thus item (ii) holds.

Item (iii) is proved as follows. Let φ be as above. A child-only tree pattern P is defined as $(\{r, m_1, \ldots, m_n\}, \{\langle r, m_1\rangle, \ldots, \langle r, m_n\rangle\}, r, l)$, where $l(m_w) = \{p_w, @_a \neq 2\}$ for every $w \in \{1, \ldots, n\}$ and $l(r) = \emptyset$. Each Q_i is defined as $(\{r_i, n_i^1, n_i^2, n_i^3\}, \{\langle r_i, n_i^1\rangle, \langle r_i, n_i^2\rangle, \langle r_i, n_i^3\rangle\}, r_i, l_i)$ with $l_i(r_i) = \emptyset$, $l_i(n_i^j) = \{p_w, B_j^i\}$, where w is such that $b_j^i = x_w$ or $b_j^i = \neg x_w$ in C_i, and B_j^i is $(@_a = 0)$ iff $b_j^i = x_w$ and B_j^i is $(@_a \neq 0)$ iff $b_j^i = \neg x_w$ in C_i. It is straightforward to show that φ is satisfiable iff $P \not\subseteq_{root} \bigvee_{i=1}^k Q_i$. Indeed, every counter-example T for the containment gives rise to a satisfying variable assignment V for φ, namely $V(x_w) = 0$ if $\rho_{att}(e(m_w)) = 0$ and $V(x_w) = 1$ otherwise, where ρ_{att} is the attribute function of T and e an embedding of P into T. Vice versa, for every satisfying variable assignment for φ, we can construct a canonical tree for P which satisfies P and falsifies every Q_i. □

Corollary 1. *The following problems are* coNP*-hard:*

a) Containment for child-only tree patterns with label negation (attribute comparisons).

b) Containment for Berge-acyclic queries with unary negated atoms (with comparisons) that are unconnected and contain a constant.

c) Containment for α-acyclic conjunctive queries with unary negated atoms (with comparisons) that are connected and contain a constant.

Proof. Item a) follows from Lemmas 3 and 4.

Item b) follows from a). Let $P_1 \subseteq P_2$ be an instance of the containment problem for child-only tree patterns with label negation (att. comparisons). We then apply Proposition 1 to obtain the equivalent instance of the containment problem $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$, where $\mathcal{Q}_1$ and $\mathcal{Q}_1$ are connected Berge-acyclic queries with unary negated atoms (comparisons). We then define $\mathcal{Q}_i' = R(\texttt{root}) \wedge \mathcal{Q}_i$, $i = 1, 2$, where $\texttt{root}$ is a constant and R a unary relational name. Note that $\mathcal{Q}_i'$ is not a connected query. It is straightforward to show that $\mathcal{Q}_1 \subseteq \mathcal{Q}_2$ if and only if $\mathcal{Q}_1' \subseteq \mathcal{Q}_2'$.

For item c) we use a). Let $P \subseteq Q$ be an instance of the containment problem for child-only tree patterns with label negation (att. comparisons). We apply Proposition 1, item (i), to obtain an equivalent problem $\mathcal{P} \subseteq \mathcal{Q}$, where $\mathcal{P}$ and $\mathcal{Q}$ are Berge-acyclic queries with negated atoms (comparisons). We construct Boolean α-acyclic conjunctive queries $\mathcal{P}'$ and $\mathcal{Q}'$ as follows. Let $\{x_1, \ldots, x_n\}$ be the variables of $\mathcal{P}$, P a fresh $(n+1)$-ary relational name, $\texttt{root}$ a constant, G a fresh binary relational name, and r the root (variable) of $\mathcal{Q}$. Then $\mathcal{P}' = P(\texttt{root}, x_1, \ldots, x_n) \wedge \mathcal{P} \wedge \bigwedge_{i=1}^n G(\texttt{root}, x_i)$ and $\mathcal{Q}' = G(\texttt{root}, r) \wedge \mathcal{Q}$. Note that both $\mathcal{P}'$ and $\mathcal{Q}'$ are α-acyclic queries. In particular, $\mathcal{P}'$ is α-acyclic because all its hyperedges are contained in the hyperedge for P. We show that $\mathcal{P} \subseteq \mathcal{Q}$ iff $\mathcal{P}' \subseteq \mathcal{Q}'$.

($\Rightarrow$). Let I be an instance such that $I \models \mathcal{P}'$. Thus there exists a satisfying variable assignment $\theta : Var(\mathcal{P}') \to dom(I)$. In particular, θ is satisfying for $\mathcal{P}$ in I. Let $I' = I{\restriction}\theta(\{x_1, \ldots, x_n\})$, i.e., the sub-instance of I obtained by restricting the domain to $\theta(\{x_1, \ldots, x_n\})$. Since $I' \models \mathcal{P}$, it holds that $I' \models \mathcal{Q}$. Let θ' be a satisfying assignment for Q in I'. Then r is mapped to one of $\theta(x_j)$. Since $\texttt{root}$ must be mapped to $\texttt{root}$ and $G(\texttt{root}, \theta(x_i))$ holds in I for every i, we have that θ' is a satisfying assignment for $\mathcal{Q}'$ in I.

($\Leftarrow$). Let I be an instance such that $I \models \mathcal{P}$. Let θ be a satisfying assignment for $\mathcal{P}$ in I. W.l.o.g. we can assume that $I = I{\restriction}\theta(Var(\mathcal{P}))$. We then construct the instance I' as

$$I' = I \cup \{P(\texttt{root}, \theta(x_1), \ldots, \theta(x_n)),$$
$$G(\texttt{root}, \theta(x_1)), \ldots, G(\texttt{root}, \theta(x_n))\}.$$

Clearly, θ is a satisfying assignment for $\mathcal{P}'$ in I'. Then it holds that $I' \models \mathcal{Q}'$. Let θ' be a satisfying assignment for $\mathcal{Q}'$ in I'. In particular, $Var(\mathcal{Q})$ must be mapped to $\theta(\{x_1, \ldots, x_n\})$ since G is fresh and $\texttt{root}$ is mapped to $\texttt{root}$. Thus, θ' is a satisfying assignment for $\mathcal{Q}$ in I. Thus $I \models \mathcal{Q}$. □

We say that a conjunctive query with negated atoms or arithmetic comparisons $\mathcal{Q}$ is *pointed Berge-acyclic* if $\mathcal{Q}$

(i) contains a constant,
(ii) is connected,
(iii) is Berge-acyclic.

Corollary 1 together with Proposition 1 shows that containment for every class of conjunctive queries with negated atoms (arithmetic comparisons) that satisfies at most two of the conditions (i)–(iii) is coNP-hard.

We leave it as an open question whether containment for pointed Berge-acyclic queries is in PTime. However, we are able to obtain PTime results for root-to-root containment of child-only tree patterns with label negation (comparisons), which by Proposition 1 entails PTime containment for a particular fragment of pointed Berge-acyclic queries.

4. Polynomial-time algorithms for containment

In this section we show that *root-to-root* containment for child-only tree patterns with label negation is solvable in PTime. In view of Proposition 1 it implies PTime containment for a restricted fragment of pointed Berge-acyclic CQs with negated atoms.

We characterize root-to-root containment using *homomorphisms*. Let $P = (N, E, r, l^+, l^-)$ and $Q = (N', E', r', l'^+, l'^-)$ be child-only tree patterns with label negation. A mapping $h : N' \to N$ is called a *homomorphism* from Q to P, if the following are satisfied:

(i) $h(r') = r$,
(ii) If $(x, y) \in E'$, then $(h(x), h(y)) \in E$,
(iii) $l'^+(x) \subseteq l^+(h(x))$, for every $x \in N'$,
(iv) $l'^-(x) \subseteq l^-(h(x))$, for every $x \in N'$.

Note that the above definition without item (iv) coincides with the usual definition of homomorphism for tree patterns without negation [11].

Theorem 3. *Let P and Q be consistent child-only tree patterns with label negation. Then P is root-to-root contained in Q if and only if there exists a homomorphism from Q to P.*

Proof. Let $P = (N, E, r, l^+, l^-)$ and $Q = (N', E', r', l'^+, l'^-)$ be consistent child-only tree patterns with label negation.

($\Leftarrow$). Assume $h : N' \to N$ is a homomorphism. Let $G = (dom(G), E'', \rho, r'')$ be a graph such that $G \models_{root} P$, i.e., there is an embedding $e : N \to dom(G)$ with $e(r) = r''$. We claim that $e' = e \circ h$ is an embedding of Q in G. We check conditions (0)–(4) except (2) (which concerned the transitive closure of E) from the definition of embedding:

(0) $e'(r') = e \circ h(r') = e(r) = r''$,
(1) Let $(x, x') \in E'$. Then $(h(x), h(x')) \in E$ which implies $(e(h(x)), e(h(x'))) \in E''$,
(3) Let $x \in N'$ and $p \in l'^+(x)$. Then $p \in l^+(h(x))$, which implies that $p \in \rho(e(h(x)))$, as needed,
(4) Let $x \in N'$ and $p \in l'^-(x)$. Then $p \in l^-(h(x))$, which implies that $p \notin \rho(e(h(x)))$, as needed.

($\Rightarrow$). We show the contrapositive. Suppose there is no homomorphism from Q to P. We then construct a counter-example by taking a tree with the same structure as P. For R a tree pattern and n a node in R, we use $R.n$ to denote the subtree of R rooted in n. By induction on $depth(P.y)$, the depth of $P.y$, we show

(IH) If there is no homomorphism from $Q.x$ to $P.y$, then there exists a tree T such that $T \models_{root} P.y$ and $T \not\models_{root} Q.x$.

Base of induction: $depth(P.y) = 0$. Then there is no homomorphism from $Q.x$ to $P.y$ if either

- there exists a label $p \in \Sigma$ such that either (i) $p \in l'^+(x)$ and $p \notin l^+(y)$, or (ii) $p \in l'^-(x)$ and $p \notin l^-(y)$.
  Let $T = (N_1, E_1, r_1, \rho_1)$ be the canonical tree of $P.y$. Then $T \models_{root} P.y$. In case (i) holds, $T \not\models_{root} Q.x$ since for every mapping $e : Nodes(Q.x) \to N_1$ with $e(x) = r_1$, we have that $p \in l'^+(x)$ and $p \notin \rho_1(r_1)$ thus violating condition (3) in the definition of embedding. In case (ii) holds, we change T in that we also add p to the label of r_1. Then we still have $T \models_{root} P.y$ since $p \notin l^-(y)$, and $T \not\models_{root} Q.x$ since for every mapping $e : Nodes(Q.x) \to N_1$ with $e(x) = r_1$, we have that $p \in l'^-(x)$ and $p \in \rho_1(r_1)$ thus violating condition (4) in the definition of embedding.
- or there exists x' in N' such that $(x, x') \in E'$.
  In this case, $T \not\models_{root} Q.x$, since T is of depth 0 and thus for every mapping $e : Nodes(Q.x) \to N_1$ we have $(x, x') \in E'$ and $(e(x), e(x')) \notin E_1$, which violates condition (1) in the definition of embedding.

Step of induction: $depth(P.y) > 0$ and there is no homomorphism from $Q.x$ to $P.y$. It is because either

- there exists a label p such that either (i) $p \in l'^+(x)$ and $p \notin l^+(y)$, or (ii) $p \in l'^-(x)$ and $p \notin l^-(y)$. This case is treated exactly as the first case in the base of induction.
- or there exists x' such that $(x, x') \in E'$ and for all y_i with $(y, y_i) \in E$ it holds that there is no homomorphism from $Q.x'$ to $P.y_i$.
  Since $depth(P.y_i) < depth(P.y)$ for every such y_i, by the induction hypothesis (IH), it holds that there exists a tree $T_i = (N_i, E_i, r_i, \rho_i)$ such that $T_i \models_{root} P.y_i$ and $T_i \not\models_{root} Q.x'$. We can assume that these trees are pairwise disjoint. We then define the tree $T = (N_T, E_T, r_T, \rho_T)$ such that $N_T = \{r_T\} \cup \bigcup_i N_i$, $E_T = \bigcup_i (E_i \cup \{(r_T, r_i)\})$, and $\rho(u) = \begin{cases} l^+(y) & \text{if } u = r_T, \\ \rho_i(u) & \text{if } u \in N_i. \end{cases}$
  We claim that $T \models_{root} P.y$ and $T \not\models_{root} Q.x$. The former is by the construction of T. For the latter, suppose e is an embedding of $Q.x$ in T with $e(x) = r_T$. In particular, that means that e is an embedding of $Q.x'$ to one of T_i. Thus, $T_i \models_{root} Q.x'$, which is a contradiction.

Thus, if there is no homomorphism from $Q.r'$ to $P.r$, there exists a tree such that $T \models_{root} P$ and $T \not\models_{root} Q$. □

Corollary 2. *Root-to-root containment for child-only tree patterns with label negation is in* PTime.

Proof. Let $P \subseteq_{root} Q$ be an instance of the containment problem, where P and Q are child-only tree patterns with label negation. We first check if P is consistent. If not, we output "yes". If it is consistent, we check if Q is consistent. If not, we output "no". Both checks can be done in PTime. Otherwise, by Theorem 3 it is enough to check existence of a homomorphism from Q to P. To this purpose, we reduce the problem to checking existence of a homomorphism for child-only tree patterns without label negation. The latter can be done e.g., using a bottom-up procedure [11]. For each negated label $\neg p$ occurring in P or Q, we introduce a new label $\tilde{p}$. For a tree pattern with label negation Q, by $\tilde{Q}$ we denote the result of replacing each negated label $\neg p$ with the corresponding label $\tilde{p}$. It is straightforward to verify that there is a homomorphism from Q to P if and only if there is a homomorphism from $\tilde{Q}$ to $\tilde{P}$. □

Interestingly, using a similar homomorphism characterization, we can prove PTime results for containment of descendant-only tree patterns with label negation and tree patterns with attribute comparisons. For each of the cases we introduce the corresponding notion of a homomorphism.

Let $P = (N, E_{//}, r, l^+, l^-)$ and $Q = (N', E'_{//}, r', l'^+, l'^-)$ be descendant-only tree patterns. A mapping $h : N' \to N$ is called a *d-homomorphism* from Q to P if it satisfies the conditions (i), (iii) and (iv) of the definition of homomorphism, and, furthermore, the following condition:

(ii') If $(x, y) \in E'_{//}$, then $(h(x), h(y)) \in E^+_{//}$.

Let $P = (N, E, r, l)$ and $Q = (N', E', r', l')$ be child-only tree patterns with attribute comparisons. Then a mapping $h : N' \to N$ is called an *a-homomorphism* from Q to P if it satisfies the conditions (i), (ii) and (iii') (where (iii') is obtained from (iii) by replacing l'^+ and l^+ with l' and l), and, furthermore, the following condition:

(iv') For every $x \in N'$, if $@_a \mathrm{op} c \in l'(x)$ then there must exist $@_a \mathrm{op}' c' \in l(h(x))$ for some op' and c', and, $C \models @_a \ \mathrm{op} \ c$, where C is the set of comparisons of a-attribute in $l(h(x))$.

The above condition $C \models @_a \ \mathrm{op} \ c$ denotes the fact that the first-order logic formula

$$\forall x.(\bigwedge_{@_a \mathrm{op}' c' \in C} x \ \mathrm{op}' \ c' \to x \ \mathrm{op} \ c)$$

is valid with respect to the theory of dense linear orders.

If P and Q are descendant-only tree patterns with attribute comparisons, then h is *da-homomorphism* from Q to P if it satisfied the conditions (i), (ii'), (iii') and (iv').

The following theorem is proved similarly to Theorem 3.

Theorem 4. *The following statements hold.*

(i) For P and Q consistent child-only tree patterns with attribute comparisons, it holds that $P \subseteq_{root} Q$ if and only if there exists an a-homomorphism from Q to P.
(ii) For P and Q consistent descendant-only tree patterns with label negation (resp. with attribute comparisons), it holds that $P \subseteq_{root} Q$ if and only if there exists a d-homomorphism (resp. da-homomorphism) from Q to P.

Since existence of a d-, da- and a-homomorphism can be checked in PTime, we obtain the following.

Corollary 3. *The root-to-root containment problem is in* PTime *for the following classes of queries.*

- *Descendant-only tree patterns with label negation,*
- *Descendent-only tree patterns with attribute comparisons,*
- *Child-only tree patterns with attribute comparisons.*

As we mentioned before, child-only tree patterns with label negation can be viewed as a fragment of pointed Berge-acyclic conjunctive queries. The precise complexity for the latter fragment is an open question. In the end of this section, we give an example showing that the homomorphism characterization fails for this fragment.

Example 1. Let Q_1 be the Boolean query $R(c, x) \wedge R(c, y) \wedge Q(y) \wedge R(z, x) \wedge R(z, w) \wedge \neg Q(w)$ and Q_2 the Boolean query $R(c, u) \wedge Q(u) \wedge R(v, u) \wedge R(v, t) \wedge \neg Q(t)$, where c is a constant and all variables are existentially quantified. It can be seen that there is no homomorphism but the containment holds. Indeed, let I be an instance such that $I \models Q_1$, i.e., there is a satisfying variable assignment $\theta : Var(Q_1) \to dom(I)$. We then define a variable assignment $\theta' : Var(Q_2) \to dom(I)$ as the composition of $g : Var(Q_2) \to Var(Q_1)$ with θ, where g is defined according to the following cases.

- $I \models Q(\theta(x))$. In this case we define $g = \{u \to x, v \to z, t \to w\}$.
- $I \not\models Q(\theta(x))$. In this case we define $g = \{u \to y, v \to c, t \to x\}$.

It is straightforward to verify that θ is a satisfying assignment, thus $I \models Q_2$.

5. Conclusion and future work

In this article we have considered several restrictions on conjunctive queries with negated atoms or arithmetic comparisons. We have shown that complexity of containment can be lowered to coNP if the arity of negated atoms is bounded. We have also shown several coNP lower bound proofs that indicate that much stronger restrictions than α-acyclicity need to be imposed to make containment tractable. For one particular restricted fragment, namely child-only tree patterns with label negation, root-to-root containment is in PTime. We have also shown that root-to-root containment for child-only tree patterns with attribute comparisons is in PTime. The two main remaining open problems are:

- What is the complexity of containment for pointed Berge-acyclic CQ*s*?
- What is the complexity of containment for α-acyclic CQ*s* with negated atoms (with no bound on the arity of negated atoms)?

Acknowledgements

We thank the anonymous referees for their valuable comments. This research was supported by NWO under project number 612.001.012 (DEX). First author is supported by EPSRC under an Impact Acceleration Award (IAA).

References

[1] F.N. Afrati, S. Cohen, G.M. Kuper, On the complexity of tree pattern containment with arithmetic comparisons, Inf. Process. Lett. 111 (15) (2011) 754–760.

[2] S. Amer-Yahia, S. Cho, L. Lakshmanan, D. Srivastava, Tree pattern query minimization, VLDB J. 11 (2002) 315–331.

[3] M. Benedikt, W. Fan, G.M. Kuper, Structural properties of XPath fragments, Theor. Comput. Sci. 336 (1) (2005) 3–31.

[4] A.K. Chandra, P.M. Merlin, Optimal implementation of conjunctive queries in relational data bases, in: Proc. 9th ACM Symp. on Theory of Computing, 1977.

[5] C. Chekuri, A. Rajaraman, Conjunctive query containment revisited, Theor. Comput. Sci. 239 (2) (2000) 211–229.

[6] A. Facchini, Y. Hirai, M. Marx, E. Sherkhonov, Containment for conditional tree patterns, Log. Methods Comput. Sci. 11 (2) (2015) 4.

[7] R. Fagin, Degrees of acyclicity for hypergraphs and relational database schemes, J. ACM 30 (3) (1983) 514–550.

[8] G. Gottlob, N. Leone, F. Scarcello, The complexity of acyclic conjunctive queries, J. ACM 48 (3) (2001) 431–498.

[9] G. Gottlob, N. Leone, F. Scarcello, Hypertree decompositions and tractable queries, J. Comput. Syst. Sci. 64 (3) (2002) 579–627.

[10] A.C. Klug, On conjunctive queries containing inequalities, J. ACM 35 (1) (1988) 146–160.

[11] G. Miklau, D. Suciu, Containment and equivalence for a fragment of XPath, J. ACM 51 (1) (2004) 2–45.

[12] W. Nutt, Ontology and database systems: foundations of database systems, teaching material, http://www.inf.unibz.it/~nutt/Teaching/ODBS1314/ODBSSlides/3-conjQueries.pdf, 2013.

[13] E. Sherkhonov, M. Marx, Containment for tree patterns with attribute value comparisons, in: WebDB 2013, 2013.

[14] J.D. Ullman, Information integration using logical views, Theor. Comput. Sci. 239 (2) (2000) 189–210.

[15] R. van der Meyden, The complexity of querying indefinite data about linearly ordered domains, J. Comput. Syst. Sci. 54 (1) (1997) 113–135.

[16] F. Wei, G. Lausen, Containment of conjunctive queries with safe negation, in: ICDT 2003, 2003, pp. 343–357.

[17] M. Yannakakis, Algorithms for acyclic database schemes, in: Proc. 7th Internat. Conf. on Very Large Data Bases, 1981, pp. 82–94.