Query $q_\sim$ is obtained from query q by replacing each term $t \in N_T(q)$ with an arbitrary, but fixed representative of the equivalence class of $\sim$ that contains t.

To check whether $q_\sim$ is aux-acyclic, we next introduce the *connection graph* cg for q and τ that contains a set E_s of edges $\langle v', v\rangle$ for each aux-simple atom $R(v', v) \in q_\sim$. In addition, cg also contains a set E_t of edges $\langle v', v\rangle$ that we later use to guess a skeleton for $\sigma(q_\sim)$ more efficiently. By the definition of aux-simple atoms, we have $E_s \subseteq E_t$.

Definition 7. *The* connection graph for q and τ *is a triple* $\mathsf{cg} = \langle V, E_s, E_t\rangle$ *where $E_s, E_t \subseteq V \times V$ are smallest sets satisfying the following conditions.*

- $V = \mathsf{ind}_{\mathsf{D}_\mathcal{K}} \cup \{z \in N_V(q_\sim) \mid \tau(z) \in \mathsf{aux}_{\mathsf{D}_\mathcal{K}}\}$.
- *Set E_s contains $\langle v', v\rangle$ for all $v', v \in V$ for which a role R exist such that $R(v', v)$ is an aux-simple atom in $q_\sim$.*
- *Set E_t contains $\langle v', v\rangle$ for all $v', v \in V$ such that individuals $\{u_1, \dots, u_n\} \subseteq \mathsf{aux}_{\mathsf{D}_\mathcal{K}}$ and roles $R_1, \dots, R_n$ exist with $n > 0$, $u_n = \tau(v)$, and $\mathsf{D}_\mathcal{K} \models \mathsf{d}_{R_i}(u_{i-1}, u_i)$ for each $i \in [1, n]$ and $u_0 = \tau(v')$.*

Function $\mathsf{isDSound}(q, \mathsf{D}_\mathcal{K}, \tau)$ from Definition 8 ensures that τ satisfies the constraints in $\sim$, and that $q_\sim$ does not contain cycles consisting only of aux-simple atoms.

Definition 8. *Function* $\mathsf{isDSound}(q, \mathsf{D}_\mathcal{K}, \tau)$ *returns* t *if and only if the two following conditions hold.*

1. *For all $s, t \in N_T(q)$, if $s \sim t$, then $\tau(s) = \tau(t)$.*
2. *$\langle V, E_s\rangle$ is a directed acyclic graph.*

We next define the notions of a variable renaming for q and τ, and of a skeleton for q and σ.

Definition 9. *A substitution σ with $\mathsf{dom}(\sigma) = V \cap N_V(q)$ and $\mathsf{rng}(\sigma) \subseteq \mathsf{dom}(\sigma)$ is a* variable renaming for q and τ *if*

1. *for each $v \in \mathsf{dom}(\sigma)$, we have $\tau(v) = \tau(\sigma(v))$,*
2. *for each $v \in \mathsf{rng}(\sigma)$, we have $\sigma(v) = v$, and*
3. *directed graph $\langle \sigma(V), \sigma(E_s)\rangle$ is a forest.*

Definition 10. *A* skeleton *for q and a variable renaming σ is a directed graph $\mathcal{S} = \langle \mathcal{V}, \mathcal{E}\rangle$ where $\mathcal{V} = \sigma(V)$, and $\mathcal{E}$ satisfies $\sigma(E_s) \subseteq \mathcal{E} \subseteq \sigma(E_t)$ and it is a forest whose roots are the individuals occurring in $\mathcal{V}$.*

Finally, we present function exist that checks whether one can satisfy the constraints imposed by the roles $L(v', v)$ labelling a skeleton edge $\langle v', v\rangle \in \mathcal{E}$.

Definition 11. *Given individuals u' and u, and a set of roles L, function* $\mathsf{exist}(u', u, L)$ *returns* t *if and only if individuals $\{u_1, \dots, u_n\} \subseteq \mathsf{aux}_{\mathsf{D}_\mathcal{K}}$ with $n > 0$ and $u_n = u$ exist where*

- *if $S \in L$ exists such that $\mathsf{trans}(S) \notin \mathcal{T}$, then $n = 1$; and*
- *$u_0 = u'$, and $\mathsf{D}_\mathcal{K} \models \mathsf{d}_R(u_{i-1}, u_i)$ for each $R \in L$ and each $i \in [1, n]$.*

Candidate answer τ' for q' over $\mathsf{D}_\mathcal{K}$ is sound, if the nondeterministic procedure $\mathsf{isSound}(q, \mathsf{D}_\mathcal{K}, \tau)$ from Algorithm 1 returns t, as shown by Theorem 12.

Theorem 12. *Let π' be a substitution. Then $\Xi_\mathcal{K} \models \pi'(q')$ iff $\mathcal{K}$ is unsatisfiable, or a candidate answer τ' to q' over $\mathsf{D}_\mathcal{K}$ exists such that $\tau'|_{\vec{x}} = \pi'$ and the following conditions hold:*

Algorithm 1: $\mathsf{isSound}(q, \mathsf{D}_\mathcal{K}, \tau)$

```
if isDSound(q, D_K, τ) = f then return f
return t if each R(s,t) ∈ q_∼ is good or aux-simple
guess a variable renaming σ for q and τ
guess a skeleton S = ⟨V, E⟩ for q, σ, and τ
for ⟨v', v⟩ ∈ E, let L(v', v) = ∅
for aux-simple atom R(s,t) ∈ σ(q_∼), add R to L(s,t)
for neither good nor aux-simple R(s,t) ∈ σ(q_∼) do
    guess role P s.t. D_K ⊨ P(τ(s), τ(t)) and P ⊑*_T R
    if ⟨s,t⟩ ∉ E and trans(P) ∉ T then return f
    if s reaches t in E then
        let v_0, ..., v_n be the path from s to t in E
    else
        let a_t be the root reaching t in E via v_0, ..., v_n
        if D_K ⊭ P(τ(s), a_t) then return f
    for i ∈ [1,n], add P to L(v_{i-1}, v_i)
for ⟨v', v⟩ ∈ E do
    if exist(τ(v'), τ(v), L(v', v)) = f then return f
return t
```

1. *for each $x \in \vec{x}$, we have $\tau'(x) \in N_I$, and*
2. *a nondeterministic computation exists such that function $\mathsf{isSound}(q, \mathsf{D}_\mathcal{K}, \tau)$ returns* t.

The following results show that our function isSound runs in nondeterministic polynomial time.

Theorem 13. *Function* $\mathsf{isSound}(q, \mathsf{D}_\mathcal{K}, \tau)$ *can be implemented so that*

1. *it runs in nondeterministic polynomial time,*
2. *if each binary atom in q is either good or aux-simple w.r.t. τ, it runs in polynomial time, and*
3. *if the TBox $\mathcal{T}$ and the query q are fixed, it runs in polynomial time in the size of the ABox $\mathcal{A}$.*

Each rule in $\mathsf{D}_\mathcal{K}$ contains a fixed number of variables, so we can compute all consequences of $\mathsf{D}_\mathcal{K}$ using polynomial time. Thus, we can compute CQ q and substitution τ in polynomial time, and by Proposition 2, we can also check whether $\mathcal{K}$ is unsatisfiable using polynomial time; hence, by Theorem 13, we can check whether a certain answer to q' over $\Xi_\mathcal{K}$ exists using nondeterministic polynomial time in combined complexity (i.e., when the ABox, the TBox, and the query are all part of the input), and in polynomial time in data complexity (i.e., when the TBox and the query are fixed, and only the ABox is part of the input).

The filtering procedure by Stefanoni, Motik, and Horrocks (2013) is polynomial, whereas the one presented in this paper introduces a source of intractability. In Theorem 14 we show that checking whether a candidate answer is sound is an NP-hard problem; hence, this complexity increase is unavoidable. We prove our claim by reducing the NP-hard problem of checking satisfiability of a 3CNF formula φ (Garey and Johnson 1979). Towards this goal, we define an $\mathcal{ELHO}^s$ KB $\mathcal{K}_\varphi$ and a Boolean CQ q_φ such that φ is satisfiable if and only if $\Xi_{\mathcal{K}_\varphi} \models q_\varphi$. Furthermore, we define a substitution τ_φ, and we finally show that τ_φ is a unique candidate answer to q_φ over $\mathsf{D}_{\mathcal{K}_\varphi}$.

Theorem 14. *Checking whether a candidate answer is sound is* NP*-hard.*

		Inds.	Unary atoms	Binary atoms	Total atoms	Ratio
U5	before	100,848	169,079	296,941	466,020	
	after	100,873	511,115	1,343,848	1,854,963	3.98
U10	before	202,387	339,746	598,695	938,441	
	after	202,412	1,026,001	2,714,214	3,740,215	3.98
U20	before	426,144	714,692	1,259,936	1,974,628	
	after	426,169	2,157,172	5,720,670	7,877,842	3.99

Table 2: Evaluation results

	q_3^l				q_1^t				q_2^t				q_3^t				q_4^t			
	C	U	F	N	C	U	F	N	C	U	F	N	C	U	F	N	C	U	F	N
U5	10	0	0.06	0	73K	12	1.71	7.55	3K	0	0.01	0	157K	66	1.07	8.6	30K	63	2.44	10.9
U10	22	0	0.06	0	149K	12	1.68	7.54	6K	0	0.01	0	603K	81	1.20	9.6	61K	63	2.44	10.9
U20	43	0	0.07	0	313K	12	1.66	7.55	12K	0	0.01	0	2.6M	90	1.28	10.3	129K	63	2.44	10.9

(C) total number of candidate answers (U) percentage of unsound answers (F) average filtering time in ms
(N) average number of nondeterministic choices required for each candidate answer

Preliminary Evaluation

We implemented our algorithm in a prototypical system, and we conducted a preliminary evaluation with the goal of showing that the number of consequences of $\mathsf{D}_\mathcal{K}$ is reasonably small, and that the nondeterminism of the filtering procedure is manageable. Our prototype uses the RDFox (Motik et al. 2014) system to materialise the consequences of $\mathsf{D}_\mathcal{K}$. We ran our tests on a MacBook Pro with 4GB of RAM and a 2.4Ghz Intel Core 2 Duo processor.

We tested our system using the version of the LSTW benchmark (Lutz et al. 2013) by Stefanoni, Motik, and Horrocks (2013). The TBox of the latter is in $\mathcal{ELHO}$, and we extended it to $\mathcal{ELHO}^s$ by making the role *subOganizationOf* transitive and by adding an axiom of type 5 and an axiom of type 7. We used the data generator provided by LSTW to generate KBs U5, U10, and U20 of 5, 10, and 20 universities, respectively. Finally, only query q_3^l from the LSTW benchmark uses transitive roles, so we have manually created four additional queries. Our system, the test data, and the queries are all available online.[1] We evaluated the practicality of our approach using the following two experiments.

First, we compared the size of the materialised consequences of $\mathsf{D}_\mathcal{K}$ with that of the input data. As the left-hand side of Table 2 shows, the ratio between the two is four, which, we believe, is acceptable in most practical scenarios.

Second, we measured the 'practical hardness' of our filtering step on our test queries. As the right-hand side of Table 2 shows, soundness of a candidate answer can typically be tested in as few as several milliseconds, and the test involves a manageable number of nondeterministic choices. Queries q_3^t and q_4^t were designed to obtain a lot of candidate answers with auxiliary individuals, so they retrieve many unsound answers. However, apart from query q_3^t, the percentage of the candidate answers that turned out to be unsound does not change with the increase in the size of the ABox. Therefore, while some queries may be challenging, we believe that our algorithm can be practicable in many cases.

4 Acyclic and Arborescent Queries

In this section, we prove that answering a simple class of tree-shaped acyclic CQs—which we call *arborescent*—over $\mathcal{ELHO}$ KBs is tractable, whereas answering acyclic queries is NP-hard. In addition, we show that extending $\mathcal{EL}$ with transitive or reflexive roles makes answering arborescent queries NP-hard. This is in contrast with the recent result by Bienvenu et al. (2013), who show that answering acyclic CQs over $\mathcal{ELH}$ KBs is tractable. We start by introducing acyclic and arborescent queries.

Definition 15. *For q a Boolean CQ, $\mathsf{dg}_q = \langle N_V(q), E\rangle$ is a directed graph where $\langle x, y\rangle \in E$ for each $R(x, y) \in q$. Query q is* acyclic *if the graph obtained from dg_q by removing the orientation of edges is acyclic; q is* arborescent *if q contains no individuals and dg_q is a rooted tree with all edges pointing towards the root.*

Definition 16 and Theorem 17 show how to answer arborescent CQs over $\mathcal{ELHO}$ KBs in polynomial time. Intuitively, we apply the fork rule (cf. Definition 6) bottom-up, starting with the leaves of q and spread constraints upwards.

Definition 16. *Let $\mathcal{K}$ be an $\mathcal{ELHO}$ KB, let $\mathsf{D}_\mathcal{K}$ be the datalog program for $\mathcal{K}$, let $\mathsf{ind}_{\mathsf{D}_\mathcal{K}}$ and $\mathsf{aux}_{\mathsf{D}_\mathcal{K}}$ be as specified in Definition 4, and let q be an arborescent query rooted in $r \in N_V(q)$. For each $y \in N_V(q)$ with $y \neq r$, and each $V \subseteq N_V(q)$, sets r_y and P_V are defined as follows.*

$$\mathsf{r}_y = \{R \in N_R \mid R(y, x) \in q \text{ with } x \text{ the parent of } y \text{ in } \mathsf{dg}_q\}$$
$$\mathsf{P}_V = \{y \in N_V(q) \mid \exists x \in V \text{ with } x \text{ the parent of } y \text{ in } \mathsf{dg}_q\}$$

Set RT is the smallest set satisfying the following conditions.

- $\{r\} \in \mathsf{RT}$ *and the level of* $\{r\}$ *is* 0.
- *For each set $V \in \mathsf{RT}$ with level n, we have $\mathsf{P}_V \in \mathsf{RT}$ and the level of P_V is $n + 1$.*
- *For each set $V \in \mathsf{RT}$ with level n and each $y \in \mathsf{P}_V$, we have $\{y\} \in \mathsf{RT}$ and the level of $\{y\}$ is $n + 1$.*

For each $V \in \mathsf{RT}$, set c_V contains each $u \in \mathsf{aux}_{\mathsf{D}_\mathcal{K}} \cup \mathsf{ind}_{\mathsf{D}_\mathcal{K}}$ such that $\mathsf{D}_\mathcal{K} \models B(u)$ for each unary atom $B(x) \in q$ with $x \in V$. By reverse-induction on the level of the sets in RT, each $V \in \mathsf{RT}$ is associated with a set $\mathsf{A}_V \subseteq \mathsf{ind}_{\mathsf{D}_\mathcal{K}} \cup \mathsf{aux}_{\mathsf{D}_\mathcal{K}}$.

- *For each set $V \in \mathsf{RT}$ of maximal level, let $\mathsf{A}_V = \mathsf{c}_V$.*
- *For $V \in \mathsf{RT}$ a set of level n where A_V is undefined but A_W has been defined for each $W \in \mathsf{RT}$ of level $n + 1$, let $\mathsf{A}_V = \mathsf{c}_V \cap (\mathsf{i}_V \cup \mathsf{a}_V)$, where i_V and a_V are as follows.*

$$\mathsf{i}_V = \{u \in \mathsf{ind}_{\mathsf{D}_\mathcal{K}} \mid \forall y \in \mathsf{P}_V \exists u' \in \mathsf{A}_{\{y\}}.\mathsf{D}_\mathcal{K} \models \bigwedge_{R \in \mathsf{r}_y} R(u', u)\}$$
$$\mathsf{a}_V = \{u \in \mathsf{aux}_{\mathsf{D}_\mathcal{K}} \mid \exists u' \in \mathsf{A}_{\mathsf{P}_V} \forall y \in \mathsf{P}_V.\mathsf{D}_\mathcal{K} \models \bigwedge_{R \in \mathsf{r}_y} \mathsf{d}_R(u', u)\}$$

Function $\mathsf{entails}(\mathsf{D}_\mathcal{K}, q)$ returns t if and only if $\mathsf{A}_{\{r\}}$ is nonempty.

Theorem 17. *For $\mathcal{K}$ a satisfiable $\mathcal{ELHO}$ KB and q an arborescent query, function $\mathsf{entails}(\mathsf{D}_\mathcal{K}, q)$ returns t if and only if $\Xi_\mathcal{K} \models q$. Furthermore, function $\mathsf{entails}(\mathsf{D}_\mathcal{K}, q)$ runs in time polynomial in the input size.*

[1] http://www.cs.ox.ac.uk/isg/tools/EOLO/

Finally, we show that (unless PTIME = NP), answering arbitrary acyclic queries over $\mathcal{ELHO}$ KBs is harder than answering arborescent queries, and we show that adding transitive or reflexive roles to the DL $\mathcal{EL}$ makes answering arborescent queries intractable.

Theorem 18. *For $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ a KB and q a Boolean CQ, checking $\mathcal{K} \models q$ is* NP*-hard in each of the following cases.*

1. *The query q is acyclic and the TBox $\mathcal{T}$ is in $\mathcal{ELHO}$.*
2. *The query q is arborescent and the TBox $\mathcal{T}$ consists only of axioms of type* 1 *and* 7*, and of one axiom of type* 8*.*
3. *The query q is arborescent and the TBox $\mathcal{T}$ consists only of axioms of type* 1 *and* 7*, and of one axiom of type* 9*.*

5 Outlook

In future, we shall adapt our filtering procedure to detect unsound answers already during query evaluation. Moreover, we shall extend Algorithm 1 to handle complex role inclusions, thus obtaining a practicable approach for OWL 2 EL.

Acknowledgements

This work was supported by Alcatel-Lucent; the EU FP7 project OPTIQUE; and the EPSRC projects MASI^3, Score!, and DBOnto.

References

Baader, F.; Calvanese, D.; McGuinness, D.; Nardi, D.; and Patel-Schneider, P. F., eds. 2007. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press.

Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the $\mathcal{EL}$ envelope. In Kaelbling, L. P., and Saffiotti, A., eds., *IJCAI 2005*, 364–369.

Bienvenu, M.; Ortiz, M.; Simkus, M.; and Xiao, G. 2013. Tractable queries for lightweight description logics. In *IJCAI 2013*.

Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2007. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. of Automated Reasoning* 39(3):385–429.

Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; Poggi, A.; Rodriguez-Muro, M.; Rosati, R.; Ruzzi, M.; and Savo, D. F. 2011. The Mastro system for ontology-based data access. *Semantic Web Journal* 2(1):43–53.

Cuenca Grau, B.; Horrocks, I.; Motik, B.; Parsia, B.; Patel-Schneider, P.; and Sattler, U. 2008. OWL 2: The next step for OWL. *Journal of Web Semantics* 6(4):309–322.

Eiter, T.; Lutz, C.; Ortiz, M.; and Simkus, M. 2009. Query answering in description logics with transitive roles. In *IJCAI 2009*, 759–764.

Eiter, T.; Ortiz, M.; Simkus, M.; Tran, T.-K.; and Xiao, G. 2012. Query rewriting for Horn-$\mathcal{SHIQ}$ plus rules. In *AAAI 2012*.

Fitting, M. 1996. *First-order logic and automated theorem proving (2nd ed.)*. Springer-Verlag New York, Inc.

Garey, M. R., and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co.

Glimm, B.; Horrocks, I.; Lutz, C.; and Sattler, U. 2008. Conjunctive query answering for the description logic $\mathcal{SHIQ}$. *Journal of Artif. Intell. Res.* 31:151–198.

Gottlob, G.; Kikot, S.; Kontchakov, R.; Podolskii, V. V.; Schwentick, T.; and Zakharyaschev, M. 2014. The price of query rewriting in ontology-based data access. *Artif. Intell.* 213:42–59.

Kontchakov, R.; Lutz, C.; Toman, D.; Wolter, F.; and Zakharyaschev, M. 2011. The combined approach to ontology-based data access. In Walsh, T., ed., *IJCAI 2011*, 2656–2661.

Krötzsch, M.; Rudolph, S.; and Hitzler, P. 2007. Conjunctive queries for a tractable fragment of OWL 1.1. In Aberer, K., et al., eds., *ISWC 2007*, 310–323.

Krötzsch, M.; Rudolph, S.; and Hitzler, P. 2008. ELP: Tractable rules for OWL 2. In Sheth, A., et al., eds., *ISWC 2008*, 649–664.

Krötzsch, M. 2010. Efficient inferencing for OWL EL. In *JELIA 2010*, volume 6341, 234–246.

Lutz, C.; Seylan, I.; Toman, D.; and Wolter, F. 2013. The combined approach to OBDA: Taming role hierarchies using filters. In *ISWC 2013*, volume 8218, 314–330.

Motik, B.; Nenov, Y.; Piro, R.; Horrocks, I.; and Olteanu, D. 2014. Parallel materialisation of datalog programs in centralised, main-memory RDF systems. In *AAAI 2014*.

Ortiz, M.; Rudolph, S.; and Simkus, M. 2011. Query answering in the Horn fragments of the description logics $\mathcal{SHOIQ}$ and $\mathcal{SROIQ}$. In Walsh, T., ed., *IJCAI 2011*, 1039–1044.

Rodriguez-Muro, M.; Kontchakov, R.; and Zakharyaschev, M. 2013. Ontology-based data access: Ontop of databases. In *ISWC 2013*, 558–573. Springer.

Stefanoni, G., and Motik, B. 2014. Answering conjunctive queries over $\mathcal{EL}$ knowledge bases with transitive and reflexive roles. *CoRR* abs/1411.2516.

Stefanoni, G.; Motik, B.; and Horrocks, I. 2013. Introducing nominals to the combined query answering approaches for EL. In *AAAI 2013*.

Venetis, T.; Stoilos, G.; and Stamou, G. B. 2014. Query extensions and incremental query rewriting for OWL 2 QL ontologies. *J. Data Semantics* 3(1):1–23.

Yannakakis, M. 1981. Algorithms for acyclic database schemes. In *VLDB 1981*, 82–94.