The Price of Selfishness: Conjunctive Query Entailment for $\mathcal{ALC}_{\mathsf{Self}}$ is $2\mathrm{ExpTime}$-hard (Extended Abstract)[⋆]

Bartosz Bednarczyk[1,2] 🆔✉ and Sebastian Rudolph[1] 🆔✉

[1] Computational Logic Group, Technische Universität Dresden, Germany
[2] Institute of Computer Science, University of Wrocław, Poland
{bartosz.bednarczyk, sebastian.rudolph}@tu-dresden.de

Various modelling features of DLs affect the complexity of conjunctive query (CQ) entailment in a rather strong sense. For the most popular basic description logic (DL), $\mathcal{ALC}$, the complexity of CQ entailment is known to be ExpTime-complete, as is that of knowledge-base satisfiability. It was first shown in [9, Thm. 2] that CQ entailment becomes exponentially harder when $\mathcal{ALC}$ is extended with inverse roles ($\mathcal{I}$), while the complexity of satisfiability remains the same. Shortly after, a combination of transitivity and role-hierarchies ($\mathcal{SH}$) was shown to be another culprit of higher worst-case complexity of reasoning [5, Thm. 1]. Finally, also nominals ($\mathcal{O}$) turned out to be equally problematic [10, Thm. 9]. On the other hand, there are "benign" DL constructs that do not affect the complexity of CQ entailment. Examples are counting ($\mathcal{Q}$) [9, Thm. 4] (the complexity stays the same even for expressive arithmetical constraints [1, Thm. 21]), role-hierarchies alone ($\mathcal{H}$) [6, Cor. 3], or even a tamed use of higher-arity relations [2, Thm. 20].

Our result. We study CQ entailment in $\mathcal{ALC}_{\mathsf{Self}}$, an extension of $\mathcal{ALC}$ with the Self operator, *i.e.* a modelling feature that allows us to specify the situation when an element is related to it*self* by a binary relationship. Among other things, this allows us to formalise the concept of a "narcissist", *e.g.* Narcissist $\equiv \exists loves.\mathsf{Self}$. The Self operator is supported by the OWL 2 Web Ontology Language and the DL $\mathcal{SROIQ}$ [7]. Due to its simplicity (it only refers to one element), it is easy to handle by automata techniques [4] or consequence-based methods [11] and thus, so far, there has been no real indication that the added expressivity provided by Self may change anything, complexity-wise. Arguably, this impression is further corroborated by the observation that Self features in two profiles of OWL 2 (EL/RL), without harming tractability [8].

However, we present a rather counter-intuitive result, namely that CQ entailment for $\mathcal{ALC}_{\mathsf{Self}}$ is exponentially harder than for $\mathcal{ALC}$, thus placing the seemingly innocuous Self operator among the "malign" modelling features.

Theorem 1. *CQ entailment over $\mathcal{ALC}_{\mathsf{Self}}$ TBoxes is 2ExpTime-hard.*

Our proof goes via encoding of computational trees of alternating Turing machines working in exponential space and follows a general hardness-proof-scheme by Lutz [9, Section 4]. However, to adjust the schema to $\mathcal{ALC}_{\mathsf{Self}}$, novel ideas are required: the ability to speak about self-loops is exploited to produce a single query that traverses trees in a root-to-leaf manner and to simulate disjunction inside CQs, useful to express that certain paths are repeated inside the tree.

To illustrate our proof technique, assume that the structures under consideration are two full binary trees of height n with their roots connected via the *next* role. The crucial task of the to-be-constructed query is to "synchronise" the two trees, which requires to identify and connect leaves in corresponding positions. To this end, the structures are axiomatised to satisfy a few additional properties: (a) each node is labelled with a unique Lvl_i concept (meaning that its distance from its root is equal to i), (b) every node from the $(i{-}1)$-th level is connected to its left (resp. right) child by ℓ_i (resp. r_i) role, (c) every node has a ℓ_i- and a r_i-self-loop for all $1 \leq i \leq n$, and (d) all leaves (and only they) have a *next*-self-loop. The case of $n = 2$ is depicted below.
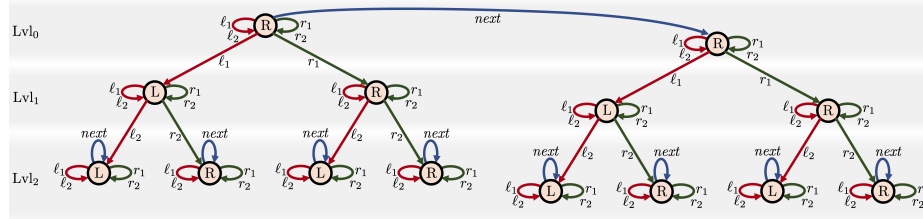
Then, one can produce a single CQ q with distinguished variables x, y such that the set of pairs $(\pi(x), \pi(y))$ over all matches π of q over the above-depicted structure is equal to the set of all leaf-leaf pairs between two trees of the same addresses (when numbered from left to right). A quick check can ensure the reader that, for $n = 2$, the following conjunctive query does the job:[3]

$$(\mathrm{Lvl}_2?; r_2^-; \ell_2^-; r_1^-; \ell_1^-; \mathrm{Lvl}_0?; next; \mathrm{Lvl}_0?; \ell_1; r_1; \ell_2; r_2; \mathrm{Lvl}_2?)(x, y)$$
$$\wedge \ (r_2^-; \ell_2^-; \ell_1^-; next; \ell_1; \ell_2; r_2; \mathrm{Lvl}_2?; r_2^-; \ell_2^-; r_1^-; next; r_1; \ell_2; r_2)(x, y)$$
$$\wedge \ (\ell_2^-; r_1^-; \ell_1^-; next; \ell_1; r_1; \ell_2; \mathrm{Lvl}_2?; r_2^-; r_1^-; \ell_1^{-1}; next; \ell_1; r_1; r_2)(x, y)$$

The first line of the query is responsible for selecting a leaf x in the left tree and a leaf y in the second tree. The second (resp. third) line ensures that the first (resp. second) bits of their addresses coincide. In the full paper [3], we show that a similar query can be produced for any n with only a polynomial blow-up.

---

[3] To write CQs in a concise way we employ the *2-way path syntax* of CQs, letting

$$(\mathrm{A}_0?; r_1; \mathrm{A}_1?; r_2; \mathrm{A}_2?; \ldots; \mathrm{A}_{n-1}?; r_n; \mathrm{A}_n?)(x_0, x_n)$$

denote the CQ $\bigwedge_{i=0}^{n} \mathrm{A}_i(x_i) \wedge \bigwedge_{i=1}^{n} r_i(x_{i-1}, x_i)$ with concepts A_i and (possibly inverted) roles r_i, where $r^-(x, y)$ stands for $r(y, x)$. Whenever A_i happens to be $\top$, it will be removed from the expression; this does not create ambiguities. Note that path CQs are just syntactic sugar and should not be mistaken *e.g.* with regular path queries.

Acknowledgements

References

1. Baader, F., Bednarczyk, B., Rudolph, S.: Satisfiability and Query Answering in Description Logics with Global and Local Cardinality Constraints. In: ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020. Frontiers in Artificial Intelligence and Applications, vol. 325, pp. 616–623. IOS Press (2020)
2. Bednarczyk, B.: Exploiting Forwardness: Satisfiability and Query-Entailment in Forward Guarded Fragment. In: Logics in Artificial Intelligence - 17th European Conference, JELIA 2021, Virtual Event, May 17-20, 2021, Proceedings. Lecture Notes in Computer Science, vol. 12678, pp. 179–193. Springer (2021)
3. Bednarczyk, B., Rudolph, S.: The Price of Selfishness: Conjunctive Query Entailment for ALCSelf is 2ExpTime-hard. CoRR **abs/2106.15150** (2021), https://arxiv.org/abs/2106.15150
4. Calvanese, D., Eiter, T., Ortiz, M.: Regular Path Queries in Expressive Description Logics with Nominals. In: IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009. pp. 714–720 (2009)
5. Eiter, T., Lutz, C., Ortiz, M., Simkus, M.: Query Answering in Description Logics with Transitive Roles. In: IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009. pp. 759–764 (2009)
6. Eiter, T., Ortiz, M., Simkus, M.: Conjunctive query answering in the description logic SH using knots. J. Comput. Syst. Sci. **78**(1), 47–85 (2012)
7. Horrocks, I., Kutz, O., Sattler, U.: The Even More Irresistible SROIQ. In: Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006. pp. 57–67. AAAI Press (2006)
8. Krötzsch, M., Rudolph, S., Hitzler, P.: ELP: Tractable Rules for OWL 2. In: The Semantic Web - ISWC 2008, 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5318, pp. 649–664. Springer (2008)
9. Lutz, C.: The Complexity of Conjunctive Query Answering in Expressive Description Logics. In: Automated Reasoning, 4th International Joint Conference, IJCAR 2008, Sydney, Australia, August 12-15, 2008, Proceedings. Lecture Notes in Computer Science, vol. 5195, pp. 179–193. Springer (2008)
10. Ngo, N., Ortiz, M., Simkus, M.: Closed Predicates in Description Logics: Results on Combined Complexity. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016, Cape Town, South Africa, April 25-29, 2016. pp. 237–246. AAAI Press (2016)
11. Ortiz, M., Rudolph, S., Simkus, M.: Worst-Case Optimal Reasoning for the Horn-DL Fragments of OWL 1 and 2. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010. AAAI Press (2010)