Finding Good Proofs for Description Logic Entailments Using Recursive Quality Measures (Extended Abstract)*

Christian Alrabbaa, Franz Baader, Stefan Borgwardt, Patrick Koopmann, and Alisa Kovtunova

Theoretical Computer Science, TU Dresden, Dresden, Germany
firstname.lastname@tu-dresden.de

Decisions made by an automated system, especially in sensitive domains involving interaction with humans, must be explained. Such explanations should be transparent and detailed enough to verify the work of the system. In this way, a user does not have to trust the manufacturers and developers implicitly, but can verify that the system indeed does what it is supposed to do. One way to explain logical reasoning is by providing a proof for the established entailments, composed of individual, easy to understand reasoning steps leading from the asserted knowledge to the inferred conclusion. Since such proofs may be arbitrarily complex, ideally we would like to pick a proof that is easiest to understand.

This theoretical paper investigates the complexity of finding proofs of a given quality among all possible alternative proofs. The quality of a proof is determined by a *proof measure* $\mathfrak{m}$ and an upper bound q that filters out bad proofs w.r.t. the measure. That is, we want to decide the existence of a proof $\mathcal{P}$ with $\mathfrak{m}(\mathcal{P}) \leq q$. Our analysis is based on the notion of a *deriver*, which generates a so-called *derivation structure* consisting of possible proof steps, from which all proofs of the given consequence can be constructed. This structure can be thought of as containing all possible instantiations of a specific set of inference rules, for some specific KB $\mathcal{K}$ and entailment η to be proven. Practical inference systems often restrict the shape of derived axioms, e.g. to be in a certain normal form or to use only subconcepts from the input, thus giving a bound on the number of steps in a derivation structure based on $\mathcal{K}$ and η. Based on this bound, we can distinguish *polynomial* and *exponential* derivers, examples of which are consequence-based reasoners for the DLs $\mathcal{EL}$ [3,9] and $\mathcal{ELI}$ [4,8]. The complexity of generating the derivation structure is not considered part of the complexity of finding good proofs since we do not want to be limited by the choice of the underlying logic or the inference rule set. In many cases, a derivation structure can also be generated in a goal-oriented way, without having to fully explore all possible rule instances.

Table 1 gives an overview of known and new results, going from general classes of measures to specific cases. This paper extends the results of [1,2], where

---

* Abstract of a paper published at CADE-28. This work is supported by the DFG grant 389792660 (TRR 248) (see https://perspicuous-computing.science, and QuantLA, GRK 1763 (https://lat.inf.tu-dresden.de/quantla).

Table 1. Overview over existing and new complexity results for deciding the existence of good proofs, w.r.t. polynomial/exponential derivers and unary/binary encoding of the bound (known results in gray).

Measure	polynomial unary	polynomial binary	exponential unary	exponential binary
Ψ-measure	≤ NP [2]	≤ NP [2]		
Size	NP [1]	NP [1]	NP [1]	NExpTime [1]
Monotone recursive Φ-measure	≤ P	≤ P	≤ ExpTime	≤ ExpTime
Local Ψ-measure	≤ P [2]	≤ P [2]	≤ ExpTime	≤ ExpTime
Tree size	P [1]	P	NP [1]	PSpace
Depth	P	P	PSpace	ExpTime
Logarithmic depth	P	P	ExpTime	ExpTime

we had established results concerning some proof measures $\mathfrak{m}$ with respect to polynomial/exponential derivers with both unary and binary encoding of the upper bound q. In particular, in our first work in this direction [1], we focused our attention on *size* as the measure of proof quality. We could show that the above decision problem is NP-complete even for polynomial derivers and unary encoding of numbers. For exponential derivers, the complexity depends on the encoding of numbers: NP-complete (NExpTime-complete) for unary (binary) encoding. For the related measure *tree size* (which assumes that the proof hypergraphs are tree-shaped, i.e. cannot reuse already derived consequences), the complexity turned out to be lower, due to the fact that a Dijkstra-like greedy algorithm can be applied. In [2], we generalized the results by introducing a class of measures called *Ψ-measures*, which contains both size and tree size and for which the same complexity upper bounds as for size could be shown for polynomial derivers. We also lifted the better upper bounds for tree size (for polynomial derivers) to *local Ψ-measures*, a natural class of proof measures.

In this paper, we extend the latter line of research by providing a more general notion of measures, *monotone recursive Φ-measures*, which also allow to measure the *depth* and *logarithmic depth* of a proof. We prove that every local Ψ-measure is a monotone recursive Φ-measure, but not vice versa. Therefore, in this paper we improve the previous results from [2]. This generalization is particularly important since the depth is related to how much of the proof a (human or automated) proof checker needs to keep in memory at the same time. We analyze these measures not only for polynomial derivers, but this time also consider exponential derivers, thus giving insights on how our complexity results transfer to more expressive logics. In addition to upper bounds for the general class of monotone recursive Φ-measures, we show improved bounds for the specific measures *depth* and *tree size*, in the latter case improving results from [1]. Overall, we thus obtain a comprehensive picture of the complexity landscape for the problem of finding good proofs for DL and other entailments.

In future work, we plan to investigate more fine-grained measures of proof quality, inspired by the weighted justification measure of [7], and their connection to explaining proofs in practice.

References

1. Alrabbaa, C., Baader, F., Borgwardt, S., Koopmann, P., Kovtunova, A.: Finding small proofs for description logic entailments: Theory and practice. In: Albert, E., Kovacs, L. (eds.) LPAR-23: 23rd International Conference on Logic for Programming, Artificial Intelligence and Reasoning. EPiC Series in Computing, vol. 73, pp. 32–67. EasyChair (2020). https://doi.org/10.29007/nhpp
2. Alrabbaa, C., Baader, F., Borgwardt, S., Koopmann, P., Kovtunova, A.: On the complexity of finding good proofs for description logic entailments. In: Borgwardt, S., Meyer, T. (eds.) Proceedings of the 33rd International Workshop on Description Logics (DL 2020). CEUR Workshop Proceedings, vol. 2663. CEUR-WS.org (2020), `http://ceur-ws.org/Vol-2663/paper-1.pdf`
3. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Kaelbling, L.P., Saffiotti, A. (eds.) Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI'05). pp. 364–369. Professional Book Center (2005), `http://ijcai.org/Proceedings/09/Papers/053.pdf`
4. Baader, F., Horrocks, I., Lutz, C., Sattler, U.: An Introduction to Description Logic. Cambridge University Press (2017). https://doi.org/10.1017/9781139025355
5. Borgida, A., Franconi, E., Horrocks, I.: Explaining $\mathcal{ALC}$ subsumption. In: ECAI 2000, Proceedings of the 14th European Conference on Artificial Intelligence, Berlin, Germany, August 20-25, 2000. pp. 209–213 (2000), `http://www.frontiersinai.com/ecai/ecai2000/pdf/p0209.pdf`
6. Engström, F., Nizamani, A.R., Strannegård, C.: Generating comprehensible explanations in description logic. In: Informal Proceedings of the 27th International Workshop on Description Logics, Vienna, Austria, July 17-20, 2014. pp. 530–542 (2014), `http://ceur-ws.org/Vol-1193/paper_17.pdf`
7. Horridge, M., Bail, S., Parsia, B., Sattler, U.: Toward cognitive support for OWL justifications. Knowl. Based Syst. **53**, 66–79 (2013). https://doi.org/10.1016/j.knosys.2013.08.021
8. Kazakov, Y.: Consequence-driven reasoning for horn SHIQ ontologies. In: Boutilier, C. (ed.) IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009. pp. 2040–2045 (2009), `http://ijcai.org/Proceedings/09/Papers/336.pdf`
9. Kazakov, Y., Krötzsch, M., Simancik, F.: The incredible ELK – from polynomial procedures to efficient reasoning with $\mathcal{EL}$ ontologies. J. Autom. Reasoning **53**(1), 1–61 (2014). https://doi.org/10.1007/s10817-013-9296-3
10. Nguyen, T.A.T., Power, R., Piwek, P., Williams, S.: Measuring the understandability of deduction rules for OWL. In: Proceedings of the First International Workshop on Debugging Ontologies and Ontology Mappings, WoDOOM 2012, Galway, Ireland, October 8, 2012. pp. 1–12 (2012), `http://www.ida.liu.se/~patla/conferences/WoDOOM12/papers/paper4.pdf`