

# Structural Decompositions of Epistemic Logic Programs<sup>\*</sup>

Markus Hecher<sup>1,2</sup>, Michael Morak<sup>3</sup>, and Stefan Woltran<sup>1</sup>

<sup>1</sup> TU Wien, Austria, {hecher, woltran}@dbai.tuwien.ac.at

<sup>2</sup> University of Potsdam, Germany

<sup>3</sup> University of Klagenfurt, Austria, michael.morak@aau.at

## 1 Introduction

Epistemic logic programs (ELPs) [5] are a popular generalization of standard Answer Set Programming (ASP) providing means for reasoning over answer sets within the language. This is accomplished by the use of modal operators such as **K** and **M** in rule bodies. Shen and Eiter [7] have shown that these operators can be conveniently expressed via a single negation-type operator (*epistemic negation not*), and have provided a new semantics for ELPs based on this operator. The richer formalism of ELPs comes at the price of higher computational complexity, namely checking whether an ELP has a world view, is  $\Sigma_3^P$ -complete, but problems higher on the polynomial hierarchy exist. In contrast to standard ASP, dedicated investigations towards tractability have not been undertaken yet. In this paper, we give first results in this direction and show that central ELP problems based on the semantics by Eiter and Shen [7] can be solved in linear time for ELPs exhibiting structural properties in terms of bounded treewidth. Then, we provide an algorithm that adheres to bounded treewidth. Finally, we show that applying treewidth to a novel dependency structure, given in terms of epistemic literals, allows to bound the number of ASP solver calls in ELP solving procedures.

## 2 Runtime Results for ELPs and Treewidth

Before we briefly discuss our results, we need to define some graph representation of ELPs. Let therefore  $\Pi$  be an ELP. We define the *primal graph*  $\mathbb{P}_\Pi$  as the graph, whose vertices comprises of all atoms of  $\Pi$  and we put an edge between two vertices whenever the two corresponding atoms appear together in at least one rule of  $\Pi$ . Then, we show that the world view existence for ELPs, whose primal graph has treewidth  $k$  can be solved in linear time. Further, we establish an algorithm and show the following theorem.

**Theorem 1 (Solving World View Existence for Treewidth in Linear Time).** *There is an algorithm for solving world view existence of a given epistemic program  $\Pi$  with  $n$  many atoms, running in time  $2^{2^{O(k)}} \cdot n$ , where  $k$  is the treewidth of the primal graph  $\mathbb{P}_\Pi$ .*

Interestingly, under reasonable assumptions in computational complexity, that is, the *exponential time hypothesis (ETH)*, one cannot significantly improve this runtime [3].

<sup>\*</sup> This is an extended abstract of a work that has been published [6] at AAAI'2020.

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

### 3 Bounding Calls to Standard ASP Solvers

Next, we present consequences of an algorithm for solving world view existence by means of calls to ASP solvers. Notably, our algorithm allows to bound the number of ASP solver calls such that for small treewidth only linearly many of these calls are required. To this end, we say for a given ELP  $\Pi$  that two vertices  $a, b$  of primal graph  $\mathbb{P}_\Pi$  are *non-epistemically connected* iff there is a path  $\langle a, v_1, \dots, v_n, b \rangle$  with  $n \geq 0$  in  $\mathbb{P}_\Pi$ , such no vertex  $v_i$  ( $1 \leq i \leq n$ ) appears under epistemic negation in  $\Pi$ . Now, the *epistemic primal graph*  $\mathbb{E}_\Pi$  of  $\Pi$  is a graph, whose vertices are only atoms of  $\Pi$  that appear under epistemic negation and there is an edge between two vertices of  $\mathbb{E}_\Pi$  iff the corresponding atoms are non-epistemically connected in  $\mathbb{P}_\Pi$ . Intuitively, two vertices form an edge in  $\mathbb{E}_\Pi$  iff there is a direct edge in  $\mathbb{P}_\Pi$  or they are connected in  $\mathbb{P}_\Pi$  via atoms that *do not* appear in epistemic literals. The concept of  $\mathbb{E}_\Pi$  is inspired by related work [4].

We establish an algorithm that allows to show the following theorem.

**Theorem 2.** *Given an ELP  $\Pi$  using  $n$  many atoms, world view existence can be solved with at most  $\mathcal{O}(2^k \cdot n)$  calls to an underlying ASP solver, where  $k$  is the treewidth of  $\mathbb{E}_\Pi$ .*

Indeed, our algorithm can be used for the classic *scholarship eligibility problem* [5]. We observed that this domain seems to be a “best case”-scenario for using our approach, where our algorithm naturally separates the ELPs into parts. However, standard ELP solvers seem to struggle in this setting when the program sizes increase; cf. e.g. [1].

### 4 Conclusions

This work provides the first parameterized complexity analysis of ELP solving w.r.t. treewidth. Our approach partitions an ELP according to a tree decomposition, and then solves the entire ELP by evaluating these parts in turn. Note that this is different from (ELP) splitting [2]. For future work, we aim to extend our algorithms to the formula evaluation problem, which should work in a similar fashion to our existing algorithms, given a suitable graph representation. Furthermore, we would like to apply our approach to other ELP semantics. There, we do not anticipate large obstacles, since most semantics are reduct-based, and the reduct is an easily exchangeable part in our algorithms.

### References

1. Bichler, M., Morak, M., Woltran, S.: Single-shot epistemic logic program solving. In: Proc. IJCAI. pp. 1714–1720 (2018)
2. Cabalar, P., Fandinno, J., Fariñas del Cerro, L.: Splitting epistemic logic programs. In: Proc. LPNMR. pp. 120–133 (2019)
3. Fichte, J.K., Hecher, M., Pfandler, A.: Lower bounds for QBFs of bounded treewidth. In: LICS. pp. 410–424. ACM (2020)
4. Galian, R., Ramanujan, M.S., Szeider, S.: Combining treewidth and backdoors for CSP. In: Proc. STACS. pp. 36:1–36:17 (2017)
5. Gelfond, M.: Strong introspection. In: Proc. AAAI. pp. 386–391. AAAI Press (1991)
6. Hecher, M., Morak, M., Woltran, S.: Structural decompositions of epistemic logic programs. In: AAAI. pp. 2830–2837. AAAI Press (2020)
7. Shen, Y., Eiter, T.: Evaluating epistemic negation in answer set programming. *Artif. Intell.* 237, 115–135 (2016)