A Fine-Grained Complexity View on Propositional Abduction — Algorithms and Lower Bounds

Extended Abstract*

Victor Lagerkvist[1], **Mohamed Maizia**[1,2], **Johannes Schmidt**[2]
[1]Department of Computer and Information Science, Linköping University, Sweden
[2]Department of Computer Science and Informatics, Jönköping University, Sweden
victor.lagerkvist@liu.se, mohamed.maizia@ju.se, johannes.schmidt@ju.se

The *Boolean satisfiability* problem is a well-known NP-complete problem. Due to the rapid advance of SAT solvers, many combinatorial problems are today solved by reducing to SAT, which can then be solved with off-the-shelf solvers. SAT fundamentally encodes a form of *monotonic* reasoning in the sense that conclusions remain valid regardless if new information is added. However, this turned out to be inappropriate for practical modeling, and thus *non-monotonic* logics emerged. Here, one is able to retract a statement if new data is added which violates the previous conclusion. One of the best known examples of non-monotonic reasoning is *abductive* reasoning where we are interested in finding explanations. More formally, an instance of the abduction problem is given by the triple (KB, H, M), where KB is the knowledge base, H the hypotheses, and M the manifestation. The question is whether there exists an explanation, that is, a set $E \subseteq H$ such that 1) $\mathrm{KB} \cup E$ is consistent, and 2) $\mathrm{KB} \cup E \models M$. Abduction has many practical applications, e.g., scientific discovery [Inoue *et al.*, 2009], network security [Alberti *et al.*, 2005], computational biology [Ray *et al.*, 2006], medical diagnosis [Obeid *et al.*, 2019], and XAI [Ignatiev, 2020; Racharak and Tojo, 2021]. The latter application could be especially important due to the continued emergence of AI in new and surprising applications, which need to be made GDPR compliant [Sovrano *et al.*, 2020]. The incitement for solving abduction fast, even when it is classically intractable, thus seems highly practically motivated.

Can non-monotonic reasoning be performed as efficiently as monotonic reasoning, or are there fundamental differences between the two? Classical complexity theory says that the two problems are different: SAT is NP-complete, while most forms of non-monotonic reasoning, including *propositional* abduction, are generally Σ_2^P-complete. However, modern complexity theory typically tells a different story, where classical hardness results do not imply that the problems are hopelessly intractable, but rather that different algorithmic schemes should be applied. For SAT, there is a healthy amount of theoretical research complementing the advances of SAT solvers, and k-SAT for every k can be solved substantially faster than 2^n (where n is the number of variables) via the resolution-based *PPSZ* algorithm [Paturi *et al.*, 2005]. There is a complementary theory of lower bounds where the central conjecture is that 3-SAT is not solvable in $2^{o(n)}$ time (*exponential-time hypothesis* (ETH) [Impagliazzo and Paturi, 2001]) and the *strong exponential-time hypothesis* (SETH) which implies that SAT with unrestricted clause length (CNF-SAT) cannot be solved in c^n time for any $c < 2$.

In contrast, the precise exponential time complexity of abduction is currently a blind spot, and *no* improved algorithms are known for the intractable cases. We thus issue a systematic attack on the complexity of abduction with a particular focus on the natural complexity parameter n, the number of variables in the knowledge base, sometimes supplemented by $|H|$ or $|M|$, the number elements in the hypothesis H or manifestation M. To obtain general results we primarily consider the setup where we are given a set of relations Γ (a *constraint language*) where the knowledge base of an instance is provided by a Γ-formula. We write $\mathrm{ABD}(\Gamma)$ for this problem and additionally also consider the variant where an explanation only consists of positive literals (P-$\mathrm{ABD}(\Gamma)$) since these two variants exhibit interesting differences. The classical complexity of abduction is either in P, NP-complete, coNP-complete, or Σ_2^P-complete [Nordh and Zanuttini, 2008], and the main question now is for which intractable Γ it is possible to beat exhaustive search. According to Cygan et al., tools to precisely analyze the exponential time complexity of NP-complete problems are in its infancy [2016]. For problems at higher levels of the polynomial hierarchy the situation is even more dire. Are algorithmic approaches for problems in NP still usable? Are the tools to obtain lower bounds still usable? Why are no sharp upper bounds known for problems in non-monotonic reasoning, and are these problems fundamentally different from e.g. satisfiability problems?

We successfully answer many of these questions. First, we show why enumerating all possible subsets of the hypothesis gives a bound 2^n for ABD and the (surprisingly bad) 3^n bound for P-ABD. Hence, any notion of improvement should be measured against 2^n for ABD and 3^n for P-ABD. Generally improving the factor $2^{|H|}$ (which may equal 2^n) seems difficult but we do manage this for languages Γ where all possible models of the knowledge base can be enumerated in c^n time, for some $c \leq 2$, which we call *sparsely enumerable* languages. We succeed with this for both $\mathrm{ABD}(\Gamma)$

*A full version has been accepted at IJCAI 2025.

and P-ABD(Γ), and while the algorithms for the two different cases share ideas, the details differ in intricate ways. It should be remarked that both algorithms solve the substantially more general problem of enumerating *all* (maximal) explanations which may open up further, e.g., probabilistic, applications for abduction. The enumeration algorithms in addition to exponential time also need exponential memory, but we manage to improve the naive 3^n bound for P-ABD(Γ) to 2^n with only polynomial memory. The sparsely enumerable property is strong: it fails even for 2-SAT and it is a priori not clear if it is ever true for intractable languages. Despite this, we manage to describe three properties implying sparse enumerability. This captures relations definable by equations $x_1 + \ldots + x_k = q \pmod p$ ($\textsc{Equations}^k$). The problem(s) (P-)ABD($\textsc{Equations}$) is Σ_2^P-complete and is, to the best of our knowledge, the first example of beating exhaustive search for a Σ_2^P-complete problem (under n). This yields improved algorithms for Σ_2^P-complete P-ABD(XSAT) (*exact satisfiability*) and NP-complete P-ABD($\mathrm{AFF}^{(\leq k)}$) (arity bounded equations over GF(2)).

(Type) Class	Classical complexity	Improved
$\textsc{Equations}^k$ ($k \geq 1$)	Σ_2^P-C	Yes
XSAT	Σ_2^P-C	$O^*(2^{\frac{n}{2}})$
(P) AFF^k ($k \geq 1$)	NP-C	Yes
(M) k-CNF$^+$ ($k \geq 1$)	NP-C	Yes
(P) k-CNF$^-$ $\cup$ IMP ($k \geq 1$)	NP-C	Yes
(P) finite 1-valid	coNP-C	Yes

Table 1: Upper bounds for P-ABD and ABD.

(Type) Class	Assumption	Bound
(M) 2-CNF$^+$	ETH	$(\frac{\lvert H\rvert}{\lvert M\rvert})^{o(\lvert M\rvert)}$
(P) 2-CNF$^+$ $\cup$ IMP	ETH	$(\frac{\lvert H\rvert}{\lvert M\rvert})^{o(\lvert M\rvert)}$
(M) k-CNF ($k \geq 4$)	SETH	2^n
(P) k-CNF ($k \geq 4$)	SETH	1.4142^n
CNF$^-$ $\cup$ IMP, Horn	SETH	1.2599^n
(M) CNF$^+$, DualHorn	SETH	1.2599^n

Table 2: Lower bounds for P-ABD and ABD.

We also consider more restricted types of abduction problems with a particular focus on (P-)ABD(k-CNF$^+$) where k-CNF$^+$ contains all positive clauses of arity k. Here, the problems are only NP-complete, in which case circumventing the $2^{|H|}$ barrier appears easier. For these, and similar problems, we construct an improved algorithm based on a novel reduction to a problem $\textsc{SimpleSAT}^p$ which can be solved by branching. For coNP, only P-ABD(Γ) becomes relevant, and we prove a simple but general improvement whenever a finite Γ is invariant under a constant Boolean operation.

We further prove lower bounds under (S)ETH for missing intractable cases. Let IMP $= \{\{(0,0),(0,1),(1,1)\}\}$. Under the ETH, we first prove that ABD(2-CNF$^+$) and ABD(2-CNF$^-$ $\cup$ IMP) cannot be solved in time $(\frac{|H|}{|M|})^{o(|M|)}$ under ETH, which asymptotically matches exhaustive search. For classical cases like k-CNF ($k \geq 4$) and NAE-k-SAT ($k \geq 5$) we establish sharp lower bounds of the form 2^n for ABD and 1.4142^n for P-ABD under the SETH. For (P-)ABD(CNF$^-$ $\cup$ IMP), we rule out improvements to 1.2599^n, $1.4142^{|H|}$, $2^{|M|}$, or $(|H|/|M|)^{|M|}$ under SETH. This transfers to Horn for (P-)ABD and to DualHorn for ABD. For ABD(2-CNF), we prove that sharp lower bounds under the SETH are unlikely unless NP $\subseteq$ P/Poly, leaving its precise fine-grained complexity as interesting open question.

The results are summarized in Table 1 and Table 2 where (P), respectively (M), indicates that the result only holds for P-ABD, respectively ABD. Thus, put together, we obtain a rather precise picture of the fine-grained complexity of ABD(Γ) and P-ABD(Γ) for almost all classical intractable languages Γ. Notably, we have proven that even Σ_2^P-complete problems can admit improved algorithms with respect to n, and that the barrier of exhaustively enumerating all possible explanations can be broken.

Concluding Remarks

We proved that propositional abduction, for many non-trivial cases *do* admit improvements over exhaustive search. We find it particularly interesting that even Σ_2^P-complete problems fall under the scope of our methods. Might it even be the case that Σ_2^P is not such an imposing barrier as classical complexity theory tells us? Nevertheless, despite many positive and negative results, there are still open cases remaining and many interesting directions for future research.

Faster Enumeration? We proved that finite subsets of AFF and $\textsc{Equations}$ are susceptible to enumeration. It is easy to see that Pol(AFF) contains the *Maltsev* operation $x - y + z(\pmod 2)$, while $\textsc{Equations}$ is exactly the set of symmetric relations invariant under a *partial Maltsev* operation [Lagerkvist and Wahlström, 2022]. Is this a coincidence, or could universal algebra be applied even further? For example, one can prove that if a language is *not* preserved by partial Maltsev, then it can not be sparsely enumerable. Extending this further, if one allows e.g. a polynomial-time preprocessing, could it even be the case that a Boolean (possibly non-symmetric) language is sparsely enumerable if and only if it is invariant under partial Maltsev?

2- and 3-CNF. While (P-)ABD(4-CNF) is unlikely to admit improved algorithms, (P-)ABD(k-CNF) for $k \leq 3$ is wide open. These languages are not sparsely enumerable, so the enumeration algorithms are not applicable. Yet, it appears highly challenging to prove sharp lower bounds for them (we have shown that CNF-SAT does not admit an LV-reduction to (P-)ABD(2-CNF) unless NP $\subseteq$ P/Poly).

Other Parameters? Related to the above question one could more generally ask when (P-)ABD(Γ) admits an improved algorithm with complexity parameter m (number of clauses/constraints), which we observe in general can be much larger than n. Do any of the algorithmic results carry over, and can lower bounds be obtained? For the related quantified Boolean formula problems, Williams [2002] constructed an $O(1.709^m)$ time algorithm, so one could be cautiously optimistic about analyzing abduction with m.

References

[Alberti *et al.*, 2005] Marco Alberti, Federico Chesani, Marco Gavanelli, Evelina Lamma, Paola Mello, and Paolo Torroni. Security protocols verification in abductive logic programming: A case study. In *Engineering Societies in the Agents World VI, 6th Internat. Workshop, ESAW'05*, pages 106–124, 2005.

[Cygan *et al.*, 2016] Marek Cygan, Holger Dell, Daniel Lokshtanov, Dániel Marx, and Jesper Nederlof et al. On problems as hard as CNF-SAT. *ACM Trans. Algorithms*, 12(3):41:1–41:24, 2016.

[Ignatiev, 2020] Alexey Ignatiev. Towards trustable explainable AI. In Christian Bessiere, editor, *Proc. 29th Internat. Joint Conf. on Artificial Intelligence (IJCAI'20)*, pages 5154–5158. ijcai.org, 2020.

[Impagliazzo and Paturi, 2001] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *J. Comput. Syst. Sci*, 62(2):367 – 375, 2001.

[Inoue *et al.*, 2009] Katsumi Inoue, Taisuke Sato, Masakazu Ishihata, Yoshitaka Kameya, and Hidetomo Nabeshima. Evaluating abductive hypotheses using an EM algorithm on bdds. In *Proc. 21st Internat. Joint Conf. on Artificial Intelligence (IJCAI'09)*, pages 810–815, 2009.

[Lagerkvist and Wahlström, 2022] Victor Lagerkvist and Magnus Wahlström. The (coarse) fine-grained structure of NP-hard SAT and CSP problems. *ACM Trans. Comput. Theory.*, 14(1), 2022.

[Nordh and Zanuttini, 2008] Gustav Nordh and Bruno Zanuttini. What makes propositional abduction tractable. *Artif. Intell.*, 172(10):1245–1284, 2008.

[Obeid *et al.*, 2019] Mariam Obeid, Zeinab Obeid, Asma Moubaiddin, and Nadim Obeid. Using description logic and abox abduction to capture medical diagnosis. In *Proc. 32nd Internat. Conf. on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2019*, pages 376–388, 2019.

[Paturi *et al.*, 2005] Ramamohan Paturi, Pavel Pudlák, Michael E. Saks, and Francis Zane. An improved exponential-time algorithm for k-sat. *J. ACM*, 52(3):337–364, 2005.

[Racharak and Tojo, 2021] Teeradaj Racharak and Satoshi Tojo. On explanation of propositional logic-based argumentation system. In *Proc. 13th Internat. Conf. on Agents and Artificial Intelligence (ICAART'21)*, pages 323–332, 2021.

[Ray *et al.*, 2006] Oliver Ray, Athos Antoniades, Antonis C. Kakas, and Ioannis Demetriades. Abductive logic programming in the clinical management of HIV/AIDS. In *Proc. 17th European Conf. on Artificial Intelligence*, pages 437–441, 2006.

[Sovrano *et al.*, 2020] Francesco Sovrano, Fabio Vitali, and Monica Palmirani. Modelling GDPR-compliant explanations for trustworthy AI. In *Proc. 9th Internat. Conf. on Electronic Government and the Information Systems Perspective (EGOVIS'20)*, pages 219–233, 2020.

[Williams, 2002] Ryan Williams. Algorithms for quantified boolean formulas. In *Proc. 13th Annual ACM-SIAM Symp. on Discrete Algorithms (SODA'02)*, page 299–307, 2002.