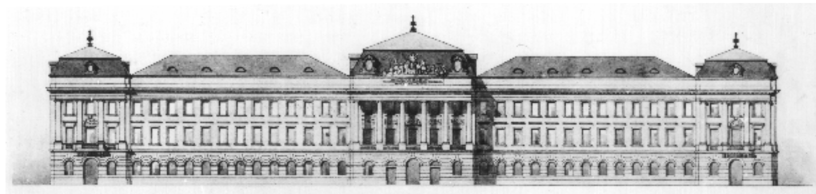
dbai

TECHNICAL REPORT



INSTITUT FÜR INFORMATIONSSYSTEME

ABTEILUNG DATENBANKEN UND ARTIFICIAL INTELLIGENCE

Multiparametric View on Answer Set Programming

DBAI-TR-2016-99

Johannes K. Fichte, Martin Kronegger, Stefan Woltran

Institut für Informationssysteme
Abteilung Datenbanken und
Artificial Intelligence
Technische Universität Wien
Favoritenstr. 9
A-1040 Vienna, Austria
Tel: +43-1-58801-18403
Fax: +43-1-58801-18493
sekret@dbai.tuwien.ac.at
www.dbai.tuwien.ac.at

DBAI TECHNICAL REPORT
2016

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

DBAI TECHNICAL REPORT
DBAI TECHNICAL REPORT DBAI-TR-2016-99, 2016

Multiparametric View on Answer Set Programming

Johannes K. Fichte [1] **Martin Kronegger** [1]
Stefan Woltran [1]

Abstract. Disjunctive answer set programming (ASP) is one of the fundamental reasoning tasks. Even the consistency problem, i.e., asking whether a given program has an answer set, is on the second level of the polynomial hierarchy. Thus, understanding the computational complexity and in particular the search for tractable fragments is of great importance for the design of efficient algorithms. During the last decades different approaches have been used find such tractable fragments. One such approach is by parameterized complexity theory. However, in the past, the full potential of this approach has not been used since only one or very few parameters have been considered at once.
In this paper, we close this gap by considering several natural parameters for the consistency problem of disjunctive ASP. Therefore, we also take the size of the answer set into account. Such a restriction is particularly interesting for applications that require small solutions. Further, we investigate on the main reasoning problems (brave and skeptical reasoning) of propositional answer set programming also taking the size of the answer sets into account. We show several novel fixed-parameter tractability (fpt) results based on combinations of parameters, XP-membership results and a variety of hardness results (para-NP, W[2], and W[1]-hardness) for the problems mentioned above. Several of these results are obtained by novel reductions to the Weighted Minimal Models Satisfiability problem (WMMSAT).

Keywords: answer set programming, propositional satisfiability

[1]TU Wien, Austria. E-mail: {jfichte, kronegger, woltran}@dbai.tuwien.ac.at

Acknowledgements: The authors gratefully acknowledge support by the Austrian Science Fund (FWF), Grant Y698. The first author is also affiliated with the Institute of Computer Science and Computational Science at University of Potsdam, Germany.

Contents

1 Introduction **3**

2 Preliminaries **7**
2.1 Propositional Satisfiability 7
2.2 Answer Set Programming 7
2.3 Parameterized Complexity 10

3 Considered Parameters **13**

4 Hardness Results **16**

5 Membership Results **22**

6 Conclusion **33**

1 Introduction

Answer set programming (ASP) is an important framework for declarative modelling and problem solving (Gebser et al., 2012; Marek and Truszczyński, 1999; Niemelä, 1999). In propositional ASP, a problem is described in terms of a logic program consisting of rules over propositional atoms. Answer sets, which are sometimes also referred as stable models, are then the solutions to such a logic program. Over the last years, many important problems, in particular, originating from the field of AI and reasoning have succinctly been represented and successfully been solved within the ASP framework, even at industrial scale, e.g., (Gebser et al., 2013), (Gebser et al., 2012b; Pontelli et al., 2012; Ricca et al., 2012), (Gebser et al., 2011). An important feature that contributes to the popularity of ASP as declarative modelling framework is its rich modeling language, which includes extended (choice, cardinality, weight constraint) rules and first-order programs. In fact, these programs can be transformed into propositional programs without extended rules (Abiteboul et al., 1995; Gebser et al., 2007; Niemelä et al., 1999; Syrjänen, 2009).

Several ASP solvers have been developed and their efficiency made huge improvements to ASP solving; among them Clasp (Drescher et al., 2008; Gebser et al., 2012a, 2013; Kaufmann et al., 2015), DLV (Leone et al., 2006), and WASP (Alviano et al., 2013). However, computational problems for disjunctive, propositional ASP (such as deciding whether a program has a solution, or whether a certain atom is contained in at least one or in all solutions) are complete for the second level of the Polynomial Hierarchy (Eiter and Gottlob, 1995). Thus, propositional ASP problems are harder than NP and therefore have a higher worst-case complexity compared to CSP and SAT.

When comparing the theoretical results with the performance of nowadays solvers, a huge gap becomes visible: theoretical results showing computational hardness and solvers being able to solve big real-world instances quickly. Unfortunately, there is little theoretical knowledge why modern solvers can deal with many real-world instances (see e.g., (Vardi, 2014) for SAT) efficiently. It is widely believed that these solvers exploit the presence of "hidden structure", see e.g., (Biere et al., 2009). Several results have been established to improve on the theoretical understanding of the effectiveness of modern SAT solvers. Among them are results by Ansótegui et al. (2008); Atserias et al. (2011); Biere et al. (2009); Gaspers and Szeider (2012); Gomes et al. (2008); Pipatsrisawat and Darwiche (2011); Williams et al. (2003a,b). In ASP theoretical results, in particular in computational complexity (Fichte and Szeider, 2015a,b; Gottlob et al., 2010; Jakl et al., 2009; Pichler et al., 2014), have been carried out to overcome this gap. However, most of these results consider hidden structure in terms of a single structural property. In the field of AI, a more fine-grained complexity analysis where hidden structure may consist of a combination of various structural properties has also been established for problems such as weighted minimal model satisfiability (WMMSAT) (Lackner and Pfandler, 2012b) and planning (Kronegger et al., 2013). The problem WMMSAT asks to decide given two propositional (CNF) formulas F_{min} and F_{cons} and an integer k whether there is a minimal model M of F_{min} that sets at most k variables to true and also satisfies F_{cons}.

In this paper, we study the computational complexity of propositional disjunctive ASP and consider several combinations of structural properties at once. Since the problem WMMSAT and ASP are quite related in terms of their problem questions, we start from results by Lackner and Pfandler (2012b) for WMMSAT, transform several of these results to ASP, point out limitations where the methods used for WMMSAT are insufficient and require to take additional structural properties into account, and finally extend them accordingly.

A mathematical framework that allows for capturing the notion of hidden structure is parameterized complexity theory (Cygan et al., 2015; Downey and Fellows, 1999, 2013; Downey et al., 1999; Flum and Grohe, 2006; Niedermeier, 2006). A complexity analysis in parameterized complexity provides a multivariate view of complexity by considering the size n of an instance together with an integer value (the parameter k) that describes properties of a given input instance. The parameter can describe (or limit) properties of an instance such as the solution size, the treewidth (Bodlaender, 1993; Robertson and Seymour, 1984), clique-width (Courcelle and Olariu, 2000), size of a smallest backdoor (Fichte and Szeider, 2015a; Williams et al., 2003a,b), or even combinations thereof. A more fine grained complexity analysis gives hope to identify techniques and structural properties of instances that enable us to solve a problem instance sufficiently fast in practice and thus provide a theoretically in-depth explanation of what makes a problem hard or easy to solve. A fundamental concept of parameterized complexity is fixed-parameter tractability, which relaxes classical polynomial-time tractability in such a way that all non-polynomial parts depend only on the size of the parameter and not on the size of the input. In more detail, fixed-parameter tractable problems can be solved in time $f(k) \cdot n^{\mathcal{O}(1)}$ by an algorithm, where $f(k)$ is a computable function depending on the parameter k only, such an algorithm is called *fpt-algorithm*. The exponential runtime is thus confined to the parameter, i.e., the function $f(k)$. An fpt-algorithm can be considered sufficiently fast as long as the parameter values of an instance are relatively low. In principle, any (natural) characteristic of problem instances is worth considering as parameter. Of course, there is no guarantee that a particular parameter permits an fpt-algorithm. However, parameterized complexity also offers methods to establish strong theoretical evidence that such an fpt-algorithm is not possible.

In this work, we use parameterized complexity to obtain a fine-grained complexity analysis and improve on the theoretical understanding of the computational complexity in ASP under a multiparametric view. Previous work in ASP mostly considered a parameter that describes a single structural property. In contrast, we identify several natural parameters and various combinations thereof based on results for WMMSAT (Lackner and Pfandler, 2012b). This allows us to draw a detailed map for various combined ASP parameters.

Main Contributions

Our main contributions can be summarized as follows:

1. We provide a parameterized complexity analysis for fundamental ASP problems that respects various combinations of natural ASP parameters, which allows us to draw a detailed map for a multivariate view on ASP complexity.

2. We study main ASP problems that also take the size of the answer set into account. Such a restriction is particularly interesting for applications that require small solutions. To the best of our knowledge, the analysis of ASP so far mainly focused on arbitrary large answer sets.

More specifically, we provide the following novel contributions: After giving some preliminary explanations on propositional satisfiability, answer set programming, and parameterized complexity in Section 2, we formally define the considered parameters in Section 3. We then turn our attention to the hardness results presented in Section 4. It will turn out that the hardness results hold for the problems k-CONSISTENCY, CONSISTENCY, k-BRAVE REASONING, BRAVE REASONING, k-SKEPTICAL REASONING, and SKEPTICAL REASONING. In particular, we present a para-NP-, three W[2]- and three W[1]-hardness results for each of the problems. In Section 5, we then present several novel membership results for k-CONSISTENCY. In that section, we first show how known results for backdoors relate to our setting and then illustrate how to use known tractability results for WMMSAT in order to obtain tractability results for our settings.

Related Work

Gottlob et al. (2002) have provided fixed-parameter tractability results of several problems in artificial intelligence and non-monotonic reasoning. Gottlob and Szeider (2008) presented a survey on parameterized complexity of problems in artificial intelligence, database theory and automated reasoning. Various parameters have been considered in the literature for ASP.[1] Some of these results already provide fixed-parameter tractability. The considered parameters include the number of atoms of a normal program that occur in negative rule bodies (Ben-Eliyahu, 1996), the number of non-Horn rules of a normal program (Ben-Eliyahu, 1996), the size of a smallest feedback vertex set in the dependency digraph of a normal program (Gottlob et al., 2002), the number of cycles of even length in the dependency digraph of a normal program (Lin and Zhao, 2004), the maximum size of heads, positive, or negative bodies (Truszczyński, 2011), the treewidth of the incidence graph of a disjunctive program (Jakl et al., 2009; Morak et al., 2010), the size of backdoors into various target classes (Fichte and Szeider, 2015b), and the clique-width of the dependency graph, the incidence graph, or the signed incidence graph of a program (Bliem et al., 2016). and a combination of two parameters: the length of the longest cycle in the dependency digraph and the treewidth of the interaction graph of a head-cycle-free program (Ben-Eliyahu and Dechter, 1994). Furthermore, several experiments regarding the size of the backdoors of ASP instances were performed (Fichte, 2015). Interestingly, some of the instances were obtained through ASP-encodings of planning instances. Recently, backdoors have also been used to construct parameterized reductions to SAT for problems harder than NP such as ASP (Fichte and Szeider, 2015a) and Abduction (Pfandler et al., 2013).

In various areas of reasoning and graph theory, systematic parameterized complexity analyses by considering all combinations of several problem parameters have recently been conducted. For instance, the parameterized complexity of abduction was studied by Gottlob and Szeider (2008) and by Fellows et al. (2012), of circumscription by Lackner and Pfandler (2012a), of constraint satisfaction by Samer and Szeider (2010), of planning by Kronegger et al. (2013), of subgraph isomorphism by Marx and Pilipczuk (2014), and of handling minimal models by Lackner and Pfandler (2012b). So far there has been no rigorous study of disjunctive ASP within the framework of parameterized complexity.

[1] Several results are not stated in terms of parameterized complexity, but can be defined in terms of such.

The rich modeling language of ASP is an important feature that contributes to the popularity of ASP as declarative modelling framework, which includes in the core language extended and first-order programs. Extended programs allow for so-called choice rules, cardinality, and weighted constraint rules (Niemelä et al., 1999). First-order programs, which are also known as non-ground programs, allow for a restricted form of first-order variables. Usually, first-order variables are systematically instantiated by means of grounding techniques within common ASP solvers and hence are transformed into a propositional program (Abiteboul et al., 1995; Gebser et al., 2007; Syrjänen, 2009). Extended rules can be transformed into rules that contain no choice, no cardinality, and no weighted constraint rules (Bomanson et al., 2014; Janhunen and Niemelä, 2011; Niemelä et al., 1999). In this work, we assume that answer set programs contain no choice, no cardinality, no weighted constraint rules, and are grounded.

2 Preliminaries

In this section, we give definitions that are used throughout this work. We start with some general concept and pointers to the standard literature for complexity theory. Then, we give an introduction to answer set programming in Section 2.2, followed by an introduction to parameterized complexity theory, in Section 2.3.

We assume that the reader is familiar with complexity theory, in particular, algorithms, (decision) problems, and complexity classes (Arora and Barak, 2009; Goldreich, 2008; Papadimitriou, 1994).

2.1 Propositional Satisfiability

First, we need some notions from propositional satisfiability. We consider a universe U of propositional variables. Note that we usually say *variable* instead of atom in the context of formulas. A literal is a variable x or its negation $\neg x$. We sometimes use the notation x^0 for $\neg x$ and x^1 for x. A *clause* is a finite set of literals, interpreted as the disjunction of these literals. A propositional formula in *conjunctive normal form (CNF)* is a finite set of clauses, interpreted as the conjunction of its clauses. A *truth assignment* (or simply an *assignment*) is a mapping $\tau : X \to \{0,1\}$ defined for a set $X \subseteq U$ of variables (atoms). For $x \in X$, we define $\tau(\neg x) = 1 - \tau(x)$. By 2^X we denote the set of all truth assignments $\tau : X \to \{0,1\}$. By $\tau^{-1}(b)$ we denote the preimage $\tau^{-1}(b) := \{\, a : a \in X, \tau(a) = b \,\}$ of the truth assignment τ for some truth value $b \in \{0,1\}$. The *truth assignment reduct* of a CNF formula F with respect to $\tau \in 2^X$ is the CNF formula F_τ obtained from F by first removing all clauses c that contain a literal set to 1 by τ, and then removing from the remaining clauses all literals set to 0 by τ. A truth assignment τ *satisfies* a given CNF formula F if $F_\tau = \emptyset$. Moreover, F is *satisfiable* if it is satisfied by some truth assignment τ. We refer to other sources (Biere et al., 2009; Kleine Büning and Lettman, 1999) for further definitions.

2.2 Answer Set Programming

Let U be a universe of propositional *atoms*. A *literal* is an atom $a \in U$ or its negation $\neg a$. A *disjunctive logic program* (or simply a *program*) P is a set of *rules* of the form

$$a_1 \vee \ldots \vee a_l \leftarrow b_1, \ldots, b_n, \neg c_1, \ldots, \neg c_m$$

where $a_1, \ldots, a_l, b_1, \ldots, b_n, c_1, \ldots, c_m$ are atoms and l, n, m are non-negative integers. Further, let H, B^+, and B^- map rules to sets of atom such that for a rule r we have $H(r) = \{a_1, \ldots, a_l\}$ (the *head* of r), $B^+(r) = \{b_1, \ldots, b_n\}$ (the *positive body* of r), and $B^-(r) = \{c_1, \ldots, c_m\}$ (*negative body* of r). We denote the sets of atoms occurring in a rule r or in a program P by $\mathrm{at}(r) = H(r) \cup B^+(r) \cup B^-(r)$ and $\mathrm{at}(P) = \bigcup_{r \in P} \mathrm{at}(r)$, respectively. We write $\mathrm{occ}_P(a) := \{\, r \in P : a \in \mathrm{at}(r) \,\}$. We denote the number of rules of P by $|P| = |\{\, r : r \in P \,\}|$. The *size* $\|P\|$ of a program P is defined as $\sum_{r \in P} |H(r)| + |B^+(r)| + |B^-(r)|$.

A rule r is *positive* (basic/negation-free) if $B^-(r) = \emptyset$, r is *normal* if $|H(r)| \leq 1$, r is a *constraint* (integrity rule) if $|H(r)| = 0$, r is *constraint-free* if $|H(r)| > 0$, r is *Horn* if it is positive and normal or a constraint, r is *definite Horn* if it is Horn and constraint-free, r is *tautological* if

$B^+(r) \cap (H(r) \cup B^-(r)) \neq \emptyset$, and *non-tautological* if it is not tautological, r is *positive-body-free* if $B^+(r) = \emptyset$, and r is a *fact* if r is definite and $(B^+(r) \cup B^-(r)) = \emptyset$. We say that a program has a certain property if all its rules have the property. **Horn** refers to the class of all Horn programs. We denote the class of all normal programs by **Normal**. **Pos+Cons** refers to the class of all programs where positive rules and arbitrary constraint rules (may also contain negative atoms) are allowed. A normal program P is *stratified* if there is a mapping $str : \text{at}(P) \to \mathbb{N}$, called *stratification*, such that for each rule r in P the following holds: (i) if $x \in H(r)$ and $y \in B^+(r)$, then $str(x) \leq str(y)$ and (ii) if $x \in H(r)$ and $y \in B^-(r)$, then $str(x) < str(y)$ (see, for example, (Apt et al., 1988; Chandra and Harel, 1985; Gelder, 1989)). We denote the class of all stratified programs by **Strat**. Let P and P' be programs. We say that P' is a *subprogram* of P (in symbols $P' \subseteq P$) if for each rule $r' \in P'$ there is some rule $r \in P$ with $H(r') \subseteq H(r)$, $B^+(r') \subseteq B^+(r)$, $B^-(r') \subseteq B^-(r)$. Let $P \in$ **Horn**, we write $\text{Constr}(P)$ for the set of constrains of P and $\text{DH}(P) = P \setminus \text{Constr}(P)$. We also identify the parts of a program P consisting of proper rules as $P_r = \{ r \in P : H(r) \neq \emptyset \}$ and constraints as $P_c = P \setminus P_r$. In this paper we are particularly interested in the following class. We occasionally write $\perp$ if $H(r) = \emptyset$. If $B^+(r) \cup B^-(r) = \emptyset$, we simply write $H(r)$ instead of $H(r) \leftarrow \emptyset, \emptyset$. We also write $H(P) := \bigcup_{r \in P} H(r)$, $B^-(P) := \bigcup_{r \in P} B^-(r)$.

A set M of atoms *satisfies* a rule r if $(H(r) \cup B^-(r)) \cap M \neq \emptyset$ or $B^+(r) \setminus M \neq \emptyset$. M is a *model* of P if it satisfies all rules of P. The *Gelfond-Lifschitz (GL) reduct* of a program P under a set M of atoms is the program P^M obtained from P by first removing all rules r with $B^-(r) \cap M \neq \emptyset$ and then removing all $\neg z$ where $z \in B^-(r)$ from the remaining rules r (Gelfond and Lifschitz, 1991). M is an *answer set* (or *stable model*) of a program P if M is a minimal model of P^M. In other words, we consider a subset M of atoms of P as a “candidate” for an answer set. By default we interpret all atoms in M as “positive literals” and all others as “negative literals”. The GL reduct establishes a semantics which treats the negative body of a rule in such a way that the positive literals naturally behave as exceptions for the implication, e.g., the rule $a \leftarrow b, \neg c$ reads as b implies a unless c belongs to M. The negative literals that occur in negative bodies of a rule have no effect on the rule (negative atoms are simply removed to construct P^M). The positive literals that occur in a literal of the negative body of a rule, however, effect the entire rule as an occurring exception (the entire rule is removed to construct P^M). M is an answer set if M is a minimal model after considering exceptions. We denote by $\text{AS}(P)$ the set of all answer sets of P and for some integer $k \geq 0$ by $\text{AS}_k(P)$ the set of all answer sets of P of size at most k.

In the following, we restrict ourselves for simplicity of exposition to programs that do not contain any tautological rules. This restriction is not significant as tautological rules can be omitted from a program without changing its answer sets (Brass and Dix, 1998), i.e., $\text{AS}(P) = \text{AS}(P')$ where P' is a program obtained from P' by removing all tautological rules (Brass and Dix, 1998; Eiter et al., 2004).

It is well known that normal Horn programs have a unique answer set or no answer set and that this set can be found in linear time. Note that every definite Horn program P has a unique minimal model which equals the least model $LM(P)$ (Gelfond and Lifschitz, 1988). Dowling and Gallier (1984) have established a linear-time algorithm for testing the satisfiability of propositional Horn formulas which easily extends to Horn programs. Further, Niemelä and Rintanen (1994) have shown that stratified programs have at most one answer set, which can be found in linear-time. In the following, we state the well-known linear-time results.

Lemma 1 *Every program in* $\{\mathbf{Horn}, \mathbf{Strat}\}$ *has at most one minimal model which can be found in linear time.*

Observation 2 (Folkore) *Let* P *be a program and* M *be an answer set of* P*, then*

1. $M \subseteq \bigcup_{r \in H(r)} H(r)$ *and*
2. $|M| \leq |P_r|$.

Observation 3 (Folklore) *Let* P *be a program and* M *be a minimal model of* P^M*. Then* M *is a minimal model of* P*.*

Proof. Assume that M is a minimal model of P^M. By definition of an answer set for each rule $r \in P$ we have (i) $B^-(r) \cap M \neq \emptyset$ or (ii) there is a corresponding rule $r' \in P^M$ such that $H(r) = H(r')$, $B^+(r) = B^+(r')$, and $B^-(r') = \emptyset$. If Case (i) holds, M satisfies r. If Case (ii) holds, M satisfies r' as M is a minimal model of P^M. Thus, M also satisfies r. Consequently, M satisfies every $r \in P$ and is hence a model of P.

In order to show that no proper subset of M is a model of P choose arbitrarily a proper subset $N \subsetneq M$. Since M is a minimal model of P^M, N cannot be a minimal model of P^M. Consequently, there must be a rule $r \in P$ such that $B^-(r) \cap M = \emptyset$ (i.e., r is not deleted by forming P^M), $B^+(r) \subseteq N$ and $H(r) \cap N = \emptyset$. Since $N \subsetneq M$ and $B^-(r) \cap M = \emptyset$, we obtain $B^-(r) \cap N = \emptyset$. Hence, $(H(r) \cap B^-(r)) \cap N = \emptyset$ and $B^+(r) \setminus N \neq \emptyset$. Thus, N does not satisfy r and is consequently not a model of P. We conclude that M is a minimal model of P^M. □

In this work, we consider the following fundamental ASP problems.

Problem:	k-CHECKING
Input:	A program P, an integer k, and a set $M \subseteq \text{at}(P)$ such that $\lvert M \rvert \leq k$.
Task:	Decide whether M is an answer set of P.

Problem:	k-CONSISTENCY
Input:	A program P and an integer k.
Task:	Decide whether P has an answer set of size at most k.

Problem:	k-BRAVE REASONING
Input:	A program P, an atom $a \in \text{at}(P)$, and an integer k.
Task:	Decide whether P has an answer set M of size at most k such that $a \in M$.

Problem:	k-SKEPTICAL REASONING
Input:	A program P, an atom $a \in \text{at}(P)$, and an integer k.
Task:	Decide whether a belongs to every answer set of size at most k of P.

We refer to the problems as CHECKING, CONSISTENCY, BRAVE REASONING and SKEPTICAL REASONING, respectively, if the integer k can be arbitrarily large.

2.3 Parameterized Complexity

In this section we give some background on parameterized complexity. For more detailed information we refer to other sources (Cygan et al., 2015; Downey and Fellows, 1999, 2013; Flum and Grohe, 2006; Niedermeier, 2006). An instance of a *parameterized problem* L is a pair $(I,k) \in \Sigma^* \times \mathbb{N}$ for some finite alphabet Σ. For an instance $(I,k) \in \Sigma^* \times \mathbb{N}$ we call I the *main part* and k the *parameter*. $\|I\|$ denotes the size of I. L is *fixed-parameter tractable* if there exist a computable function f and a constant c such that we can decide by an algorithm whether $(I,k) \in L$ in time $\mathcal{O}(f(k)\|I\|^c)$. Such an algorithm is called an *fpt-algorithm*. If L is a decision problem, then we identify L with the set of all yes-instances (I,k). FPT is the class of all fixed-parameter tractable decision problems.

Let $L \subseteq \Sigma^* \times \mathbb{N}$ and $L' \subseteq \Sigma'^* \times \mathbb{N}$ be two parameterized decision problems for some finite alphabets Σ and Σ'. An *fpt-reduction* r from L to L' is a many-to-one reduction from $\Sigma^* \times \mathbb{N}$ to $\Sigma'^* \times \mathbb{N}$ such that for all $I \in \Sigma^*$ we have $(I,k) \in L$ if and only if $r(I,k) = (I',k') \in L'$ such that $k' \leq g(k)$ for a fixed computable function $g : \mathbb{N} \to \mathbb{N}$ and there is a computable function f and a constant c such that r is computable in time $\mathcal{O}(f(k)\|I\|^c)$ (Flum and Grohe, 2006). Thus, an fpt-reduction is, in particular, an fpt-algorithm. It is easy to see that the class FPT is closed under fpt-reductions. It is clear for parameterized problems L_1, and L_2 that if $L_1 \in$ FPT and there is an fpt-reduction from L_2 to L_1, then $L_2 \in$ FPT. We would like to note that the theory of fixed-parameter intractability is based on fpt-reductions (Downey and Fellows, 1999, 2013; Flum and Grohe, 2006).

Parameterized complexity also facilitates hardness theory to rule out the existence of fpt-algorithms. Next, we will define several parameterized complexity classes capturing fixed-parameter intractability needed in this work. For this we first define the model-checking problem over $\Sigma_{t,u}$ formulas, $\mathsf{MC}[\Sigma_{\mathsf{t,u}}]$. The class $\Sigma_{t,u}$ contains all first-order formulas of the form $\exists \bar{x}_1 \forall \bar{x}_2 \exists \bar{x}_3 \ldots Q\bar{x}_t \varphi(\bar{x}_1, \ldots, \bar{x}_t)$, where the formula φ is quantifier free and the quantifier Q is an $\exists$ if t is odd and a $\forall$ if t is even, and the quantifier blocks – with the exception of the first $\exists$ block – are of length at most u. We write Σ_1 to denote $\Sigma_{1,u}$ for arbitrary $u \geq 1$. The problem $\mathsf{MC}[\Sigma_{\mathsf{t,u}}]$ is then defined as follows.

Problem:	$\mathsf{MC}[\Sigma_{\mathsf{t,u}}]$
Input:	A finite structure $\mathscr{A}$ and a formula $\psi \in \Sigma_{t,u}$.
Task:	Decide whether $\mathscr{A}$ is a model of ψ, i.e., $\mathscr{A} \models \psi$.

The so-called W-hierarchy can be defined with help of the problem $\mathsf{MC}[\Sigma_{\mathsf{t,u}}]$. For $t \geq 1, u \geq 1$, the class W[t] contains all problems that are fpt-reducible to $\mathsf{MC}[\Sigma_{\mathsf{t,u}}]$ when parameterized by the length of ψ (Downey et al., 1998; Flum and Grohe, 2005). Showing W[1]-hardness for a problem rules out the existence of a fixed-parameter algorithm under the usual complexity theoretic assumption FPT $\neq$ W[1].

The class XP of *non-uniform* polynomial-time tractable problems consists of all parameterized decision problems that can be solved in polynomial time if the parameter is considered constant. That is, $(I,k) \in L$ can be decided in time $\mathcal{O}(\|I\|^{f(k)})$ for some computable function f.

Parameterized complexity theory also offers complexity classes for problems that lie higher in the polynomial hierarchy. Let C be a classical complexity class, e.g., NP. The parameterized complexity class para-C is then defined as the class of all parameterized problems $L \subseteq \Sigma^* \times \mathbb{N}$, for some finite alphabet Σ, for which there exist an alphabet Π, a computable function $f : \mathbb{N} \to \Pi^*$, and a problem $P \subseteq \Sigma^* \times \Pi^*$ such that $P \in C$ and for all instances $(x,k) \in \Sigma^* \times \mathbb{N}$ of L we have that $(x,k) \in L$ if and only if $(x, f(k)) \in P$. Intuitively, the class para-C consists of all problems that are in C after a precomputation that only involves the parameter (Flum and Grohe, 2003). The class para-NP can also be defined via non-deterministic fpt-algorithms, i.e., para-NP contains all parameterized decision problems L such that $(I,k) \in L$ can be decided *non-deterministically* in time $\mathcal{O}(f(k)\|I\|^c)$ for some computable function f and constant c (Flum and Grohe, 2006). A parameterized decision problem is para-NP-complete if it is in NP and NP-complete when restricted to finitely many parameter values (Flum and Grohe, 2006). For parameterizations of problems that are harder than NP (like the main reasoning problems of propositional disjunctive ASP) para-NP-completeness is a desirable property as it allows us to exploit the parameter to solve the problem for small parameter values more efficiently.

For our results we will need the definition of the following two problems. The problem WSAT$_{\leq}$ is defined as follows.

Problem:	WEIGHTED SATISFIABILITY (WSAT$_{\leq}$)
Input:	A CNF formula φ and some integer k
Task:	Decide whether φ has a model $M \subseteq var(\varphi)$ of cardinality $|M| \leq k$

It is well-known that different variations of WSAT$_{\leq}$ can be used to define the W-hierarchy (see, e.g., the work of Flum and Grohe (2006)). The problem WMMSAT is defined as follows.

Problem:	WEIGHTED MINIMAL MODEL SATISFIABILITY (WMMSAT)
Input:	Two CNF formulas φ and π where $var(\pi) \subseteq var(\varphi)$ and some integer k
Task:	Decide whether φ has a minimal model $M \subseteq var(\varphi)$ of cardinality $|M| \leq k$ such that M is also a model of π.

The parameterized complexity of these problems has been studied in the work of (Lackner and Pfandler, 2012b). In their work, they have considered the parameters as listed in Table 1. Several hardness and tractability results for combined parameter turn out to be useful to show hardness and tractability results for the considered ASP problems.

k	maximum weight of the minimal model
d	maximum clause size
d^+, d^-	maximum positive / negative clause size
h	number of non-horn clauses
b	minimum size of strong Horn backdoor set
o	maximum number of occurrences of a variable in φ
p	maximum number of positive occurrences of a variable in φ
v^+, v^-	number of variables that occur as positive / negative literals in φ or in π
$\|\varphi\|$	number of clauses in φ
$\|\pi\|$	number of clauses in π
d^+_π	maximum positive clause size in π
$\|\|\pi\|\|$	length of π, i.e., the total number of variable occurrences in π

Table 1: List of considered parameters in the work of Lackner and Pfandler (2012b) for the problems WSAT and WMMSAT.

k	maximum size of an answer set
$\text{maxsize}^r_{H,B^+,B^-}$	maximum size of a non-constraint rule
$\text{maxsize}^r_{H,B^-}$	maximum size of the head and negative body of a rule
maxsize_H	maximum size of the head of a rule
$\text{maxsize}^r_{B^+}$	maximum size of the positive body of a non-constraint rule
$\text{maxsize}^r_{B^-}$	maximum size of the negative body of a rule
$\text{maxsize}^c_{B^+}$	maximum size of the positive body of a constraint
$\text{maxsize}^c_{B^-}$	maximum size of the negative body of a constraint
$\#\text{non-Horn}^r$	number of non-(definite Horn) rules
$\text{maxocc}^r_{H,B^+,B^-}$	maximum number of occurrences of a variable in P_r
maxocc^r_{H,B^-}	maximum number of occurrences of a variable in P_r when only the head and negative-body occurrences are counted
$\#\text{at}_H$	number of atoms that occur in the head
$\#\text{at}_{B^+}$	number of atoms that occur in the positive body
$\#\text{at}_{B^-}$	number of atoms that occur in the negative body
$\|P_r\|$	number of rules in P_r
$\|P_c\|$	number of rules in P_c
$\|\|P_c\|\|$	the total number of variable occurrences in P_c

Table 2: List and informal description of the considered parameters.

3 Considered Parameters

In this section, we introduce a list of ASP-parameters, which mainly originate from earlier work for WMMSAT, for our parameterized complexity analysis. In particular, we are interested in parameter combinations. Table 2 contains a list of the considered parameters and their intuitive description. Note that all considered parameters can be computed in polynomial time. A more formal descriptions is given below. Therefore, let P be a program and $X \subseteq \{H, B^+, B^-\}$ where H, B^+, and B^- are mappings defined as in Section 2.2. We omit P if the program is clear from the context. Further, let

$$\begin{aligned}
\text{at}_{X,r} &:= \cup_{f \in X} f(r)\\
\#\text{at}_X &:= |\cup_{r \in P} \text{at}_{X,r}|\\
\text{maxsize}^r_X &:= \max\Big\{ \sum_{f \in X, r' \in P} |f(r')| : |H(r')| > 0 \Big\}\\
\text{maxsize}^c_X &:= \max\Big\{ \sum_{f \in X, r' \in P} |f(r')| : |H(r')| = 0 \Big\}\\
\#\text{non-Horn}^r &:= |\{ r' : r' \in P, r' \text{ not Horn}\}|\\
\text{maxocc}^r_X &:= \max\Big\{ i : a \in at(P), i = \sum_{f \in X,\, r' \in P,\, |H(r')| > 0} |\{ a : a \in f(r') \}| \Big\}
\end{aligned}$$

Parameter Comparison

Let p and q be ASP parameters. We say that p *dominates* q (in symbols $p \preceq q$) if there is a function f such that $p(P) \leq f(q(P))$ holds for all programs P. The parameters p and q are *similar* (in symbols $p \sim q$) if $p \preceq q$ and $q \preceq p$. The parameter p *strictly dominates* q (in symbols $p \prec q$) if $p \preceq q$ but not $q \preceq p$, and p and q are *incomparable* (in symbols $p \bowtie q$) if neither $p \preceq q$ nor $q \preceq p$.

We now list some parameter dependencies that are useful for our results. Note, however, that this list is not complete.

Lemma 4 *The following parameter dependencies hold:*

(1) $\text{maxsize}^r_{H,B^-} \preceq \text{maxsize}^r_{H,B^+,B^-}$

(2) $\text{maxsize}_H \preceq \text{maxsize}^r_{H,B^+,B^-}$

(3) $\text{maxsize}^r_{B^+} \preceq \text{maxsize}^r_{H,B^+,B^-}$

(4) $\text{maxsize}^r_{B^-} \preceq \text{maxsize}^r_{H,B^+,B^-}$

(5) $\text{maxsize}_H \preceq \text{maxsize}^r_{H,B^-}$

(6) $\text{maxsize}^r_{B^-} \preceq \text{maxsize}^r_{H,B^-}$

(7) $\text{maxsize}^r_{B^+} \preceq \#\text{at}_{B^+}$

(8) $\text{maxsize}_H \preceq \#\text{at}_H$

(9) $k \preceq |P_r|$

(10) $\#\text{non-Horn}^r \preceq |P_r|$

(11) $\text{maxocc}^r_{H,B^-} \preceq \text{maxocc}^r_{H,B^+,B^-}$

(12) $\text{maxocc}^r_{H,B^+,B^-} \preceq |P_r|$

(13) $\text{maxsize}^c_{B^-} \preceq ||P_c||$

(14) $|P_c| \preceq ||P_c||$

Proof. Most dependencies trivially follow from the definition of the parameters. Concerning (12), notice that $\text{maxocc}^r_{H,B^+,B^-} \leq 3 \cdot |P_r|$. □

Problem	Parameter	Result	Reference
	$\mathrm{maxsize}^r_{H,B^+,B^-} + \mathrm{maxocc}^r_{H,B^+,B^-} + \#\mathrm{at}_{B^+} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	para-NP	Thm. 6 (Statement 1, p. 16)
	$k + \mathrm{maxsize}^r_{B^-} + \mathrm{maxsize}^c_{B^+} + \mathrm{maxsize}^c_{B^-}$	W[2]	Thm. 6 (Statement 2, p. 16)
	$k + \mathrm{maxsize}^r_{B^+} + \#\mathrm{at}_{B^+} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	W[2]	Thm. 6 (Statement 3, p. 16)
k-CONSISTENCY	$k + \mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^+,B^-} + \#\mathrm{at}_{B^+} + \lvert P_r\rvert$	W[2]	Thm. 6 (Statement 4, p. 16)
	$k + \mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^-} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	W[1]	Thm. 6 (Statement 5, p. 16)
	$\mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^+,B^-} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	W[1]	Thm. 6 (Statement 6, p. 16)
	$k + \mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^+,B^-} + \lvert P_r\rvert + \mathrm{maxsize}^c_{B^-}$	W[1]	Thm. 6 (Statement 7, p. 16)
CONSISTENCY,	$\mathrm{maxsize}^r_{H,B^+,B^-} + \mathrm{maxocc}^r_{H,B^+,B^-} + \#\mathrm{at}_{B^+} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	para-NP	Cor. 7 (Statement 1, p. 19)
k-BRAVE REASONING,	$k + \mathrm{maxsize}^r_{B^-} + \mathrm{maxsize}^c_{B^+} + \mathrm{maxsize}^c_{B^-}$	W[2]	Cor. 7 (Statement 2, p. 19)
BRAVE REASONING,	$k + \mathrm{maxsize}^r_{B^+} + \#\mathrm{at}_{B^+} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	W[2]	Cor. 7 (Statement 3, p. 19)
k-SKEPTICAL	$k + \mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^+,B^-} + \#\mathrm{at}_{B^+} + \lvert P_r\rvert$	W[2]	Cor. 7 (Statement 4, p. 19)
REASONING, and	$k + \mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^-} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	W[1]	Cor. 7 (Statement 5, p. 19)
SKEPTICAL	$\mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^+,B^-} + \lvert P_c\rvert + \mathrm{maxsize}^c_{B^-} + \lVert P_c\rVert$	W[1]	Cor. 7 (Statement 6, p. 19)
REASONING	$k + \mathrm{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \mathrm{maxocc}^r_{H,B^+,B^-} + \lvert P_r\rvert + \mathrm{maxsize}^c_{B^-}$	W[1]	Cor. 7 (Statement 7, p. 19)

Table 3: Summary of hardness results for *k*-CONSISTENCY, CONSISTENCY, *k*-BRAVE REASONING, BRAVE REASONING, *k*-SKEPTICAL REASONING, and SKEPTICAL REASONING.

4 Hardness Results

In this section, we present several hardness results for the problems k-CONSISTENCY, CONSISTENCY, k-BRAVE REASONING, BRAVE REASONING, k-SKEPTICAL REASONING, and SKEPTICAL REASONING, which are summarized in Table 3. To this aim, we establish reductions from the problems $\text{WSAT}_{\leq}$ and WMMSAT.

In the next proposition, we summarize known hardness results for WMMSAT which turn out to be useful for showing hardness for several combined parameters for k-CONSISTENCY.

Proposition 5 (Lackner and Pfandler (2012b)) WMMSAT *is* para-NP-*hard when parameterized by the following combined parameter*

1. $d+d^{+}+d^{-}+o+p+v^{-}+|\pi|+d^{+}_{\pi}+||\pi||$

WMMSAT *is* W[2]-*hard when parameterized by the following combined parameters*

2. $k+d^{-}+v^{-}+|\pi|+d^{+}_{\pi}+||\pi||$
3. $k+d^{-}+h+o+p+v^{-}+|\varphi|$

WMMSAT *is* W[1]-*hard when parameterized by the following combined parameters*

4. $k+d^{-}+h+p+|\pi|+d^{+}_{\pi}+||\pi||$
5. $d^{-}+h+o+p+|\pi|+d^{+}_{\pi}+||\pi||$
6. $k+d^{-}+h+o+p+|\varphi|+d^{+}_{\pi}$

Now we are ready to present several hardness results for k-CONSISTENCY. The main idea is to provide a polynomial-time reduction from WSAT or WMMSAT that preserves the considered parameters. In fact, one reduction is sufficient for all the considered parameters.

Theorem 6 k-CONSISTENCY *is* para-NP-*hard when parameterized by the following parameters*

1. $\text{maxsize}^{r}_{H,B^{+},B^{-}}+\text{maxocc}^{r}_{H,B^{+},B^{-}}+\#\text{at}_{B^{+}}+|P_c|+\text{maxsize}^{c}_{B^{-}}+||P_c||$

k-CONSISTENCY *is* W[2]-*hard when parameterized by the following parameters*

2. $k+\text{maxsize}^{r}_{B^{-}}+\text{maxsize}^{c}_{B^{+}}+\text{maxsize}^{c}_{B^{-}}$
3. $k+\text{maxsize}^{r}_{B^{+}}+\#\text{at}_{B^{+}}+|P_c|+\text{maxsize}^{c}_{B^{-}}+||P_c||$
4. $k+\text{maxsize}^{r}_{B^{+}}+\#\text{non-Horn}^{r}+\text{maxocc}^{r}_{H,B^{+},B^{-}}+\#\text{at}_{B^{+}}+|P_r|$

k-CONSISTENCY *is* W[1]-*hard when parameterized by the following parameters*

5. $k+\text{maxsize}^{r}_{B^{+}}+\#\text{non-Horn}^{r}+\text{maxocc}^{r}_{H,B^{-}}+|P_c|+\text{maxsize}^{c}_{B^{-}}+||P_c||$
6. $\text{maxsize}^{r}_{B^{+}}+\#\text{non-Horn}^{r}+\text{maxocc}^{r}_{H,B^{+},B^{-}}+|P_c|+\text{maxsize}^{c}_{B^{-}}+||P_c||$
7. $k+\text{maxsize}^{r}_{B^{+}}+\#\text{non-Horn}^{r}+\text{maxocc}^{r}_{H,B^{+},B^{-}}+|P_r|+\text{maxsize}^{c}_{B^{-}}$

Proof. We proceed by a reduction from the problem WMMSAT for Statements (1) and (3)–(7) and WSAT for Statement (2). Let (φ,π,k) be an instance of WMMSAT. We assume w.l.o.g. that φ contains no clauses without positive literals, since otherwise we can shift such clauses into π without effecting the size of the models and hence the minimality.[2] We now construct an instance (P,k) of k-CONSISTENCY as follows. For a clause C and $i \in \{0,1\}$ we define $C^i := \{\, a^i : x^i \in C, x \in var(C)\,\}$ where a is a fresh atom and $a^0 = \neg a$ and $a^1 = a$. Now, let $P_\varphi := \{\, C^1 \leftarrow C^0 : C \in \varphi \,\}$ and $P_\pi := \{\, \leftarrow \neg C^1, C^0 : C \in \pi \,\}$ and we define a program $P := P_\varphi \cup P_\pi$. Next, we show that φ has a minimal model M of size at most k such that M is also a model of π if and only if P has an answer set of size at most k.

($\Rightarrow$): Let M be a minimal model of φ of size at most k such that M is also a model of π. For every rule $r \in P_\varphi$, there is a corresponding clause $C \in \varphi$. Since for each clause $C \in \varphi$ it is true that (i) $C^1 \cap M \neq \emptyset$, or (ii) $C^0 \setminus M \neq \emptyset$, we obtain by construction of P_φ that (i) $H(r) \cap M \neq \emptyset$, or (ii) $B^+(r) \setminus M \neq \emptyset$ holds. Hence, M is a model of P_φ. Since $(P_\varphi)^M = P_\varphi$, the set M is also a model of $(P_\varphi)^M$. For every rule $r \in P_\pi$, there is a corresponding clause $C \in \pi$. Since for each clause $C \in \varphi$ it holds that (i) $C^1 \cap M \neq \emptyset$, or (ii) $C^0 \setminus M \neq \emptyset$, we have by construction of P_π that (i) $B^-(r) \cap M \neq \emptyset$, or (ii) $B^+(r) \setminus M \neq \emptyset$. Hence, M is a model of P_π. Then, for every rule $r \in P_\pi$ there is either (i) a corresponding rule $r' \in (P_\pi)^M$ with $B^+(r) = B^+(r')$ and $B^+(r') \setminus M \neq \emptyset$, since $B^+(r) \setminus M \neq \emptyset$, or (ii) $B^-(r) \setminus M \neq \emptyset$ and the rule r has been removed from P_π when constructing $(P_\pi)^M$. Consequently, M is also a model of $(P_\pi)^M$. It remains to observe that M is also a minimal model of P^M. For proof by contradiction assume that P^M has a model N such that $N \subsetneq M$. Let now $r \in P^M$. Then, by the construction of P there is a corresponding clause C_r such that either (i) $C_r \in \varphi$, $C_r^1 = H(r)$ and $C_r^0 = B^+(r)$, or (ii) $C_r \in \pi$ and $C_r^0 = B^+(r)$. Since N is a model of P for every rule $r \in P^M$, it holds that $H(r) \cap N \neq \emptyset$ or $B^+(r) \setminus N \neq \emptyset$. Thus we can conclude in Case (i) $C_r^1 \cap N \neq \emptyset$ or $C_r^0 \setminus N \neq \emptyset$ and thus N is also a model of φ, which however contradicts the assumption that M is a minimal model of φ. Further, we can conclude in Case (ii) that $C_r^0 \setminus N \neq \emptyset$, which however contradicts the assumption that M is a model of π. Consequently, M is an answer set of P of size at most k.

($\Leftarrow$): Conversely, assume that M is an answer set of P of size at most k. For each rule $r \in P$ there is a corresponding clause (i) $C_r \in \pi$ such that $C_r^0 = B^+(r)$ and $C_r^1 = B^-(r)$ if $H(r) = \emptyset$, or (ii) $C_r \in \varphi$ such that $C_r^1 = H(r)$ and $C_r^0 = B^+(r)$ if $B^-(r) = \emptyset$. We proceed with Case (i): By definition of an answer set, M is a model of P. Hence, for rules where $H(r) = \emptyset$, we have $B^+(r) \setminus M \neq \emptyset$ or $B^-(r) \cap M \neq \emptyset$. Thus, we obtain $C_r^1 \cap B^-(r) \neq \emptyset$ or $C_r^0 \setminus M \neq \emptyset$, which yields that M is a model of π. We proceed with Case (ii): By definition of an answer set, the set M is a model of P. Hence, for rules where $H(r) \neq \emptyset$, we have $H(r) \cap M \neq \emptyset$ or $B^+(r) \setminus M \neq \emptyset$. Since $H(r) = C_r^1$ and $B^+(r) = C_r^0$, we have $C_r^1 \cap M \neq \emptyset$ or $C_r^0 \setminus M \neq \emptyset$. Hence, M is a model of φ. For proof by contradiction assume that there is some model N of φ such that $N \subsetneq M$ and N is also a model of π. By construction of P for a clause $C \in \varphi$ there is a corresponding rule $r_c \in P$ such that $H(r_c) = C^1$, $B^+(r_c) = C^0$, and $B^-(r_c) = \emptyset$. Since $B^-(r') = \emptyset$ for every rule $r \in P_\varphi$, we have that N is also a model of $(P_\varphi)^M$, which contradicts the assumption that M is an answer set of P. Further, by construction of P for a clause $C \in \pi$ there is a corresponding rule $r_c \in P$ such that $H(r_c) = \emptyset$, $B^+(r_c) = C^0$, and

[2]Note that this has also no effect to the results we use for WMMSAT, since the parameters used in the proofs for WMMSAT remain uneffected (it only effects d and d^-, however, there d^- is already bounded by v^-; see the proofs of Theorems 16 and 17 in (Lackner and Pfandler, 2012b)).

$B^-(r_c) = C^1$. Since N is a model of π we conclude that (i) $B^-(r_c) \cap N \neq \emptyset$, or (ii) $B^+(r_c) \setminus N \neq \emptyset$. Hence, N is also a model of $(P_\pi)^M$. Thus, by Statement 1 of Observation 2 the set N is also an answer set of P, which contradicts our assumption. Consequently, M is a minimal model of φ, has size at most k, and is also a model of π.

We have established the claim that φ has a minimal model M of size at most k such that M is also a model of π if and only if P has an answer set of size at most k.

Next, we can employ the construction and proofs from above to establish a reduction from an instance (φ, k) of WSAT for Statement 2. Note that WSAT is a well known to be W[2]-hard, e.g., (Downey and Fellows, 2013). Therefore, observe that φ has a model M of size at most k if and only if P_φ has an answer set of size at most k.

Finally, it remains to observe that our reduction preserves the parameters:

- k: directly corresponds to the maximum weight of a minimal model (k)
- $\text{maxsize}^r_{H,B^+,B^-}$: directly corresponds to the maximum clause size (d)
- $\text{maxsize}^r_{B^+}$: directly corresponds to the maximum negative clause size (d^+)
- $\#\text{non-Horn}^r$: directly corresponds to the number of non-horn clauses (h)
- $\text{maxocc}^r_{H,B^+,B^-}$: directly corresponds to the maximum number of occurrences of a variable in φ (o)
- maxocc^r_{H,B^-}: directly corresponds to the maximum number of positive occurrences of a variable in φ (p)
- $\#\text{at}_{B^+}$: directly corresponds to the number of variables that occur as negative literals in φ (v^-) or in π (v^-)
- $|P_r|$: directly corresponds to the number of clauses in φ ($|\varphi|$)
- $|P_c|$: directly corresponds to the number of clauses in π ($|\pi|$)
- $\text{maxsize}^c_{B^-}$: directly corresponds to the maximum positive clause size in π (d^+_π)
- $\text{maxsize}^c_{B^+}$: directly corresponds to the maximum negative clause size in π (d^-)
- $||P_c||$: directly corresponds to the length of π, i.e., the total number of variable occurrences in π ($\|\pi\|$)

The runtime follows from Proposition 5. This concludes the proof. □

The hardness results for k-CONSISTENCY trivially extend to CONSISTENCY. Furthermore, the results also extend to k-BRAVE REASONING because one can solve k-CONSISTENCY by calling for each $a \in \text{at}(P)$ the problem k-BRAVE REASONING and return yes if at least one of the $|\text{at}(P)|$ calls of k-BRAVE REASONING returns yes. Clearly, these results then also extend to BRAVE REASONING, k-SKEPTICAL REASONING, and SKEPTICAL REASONING. Thus, we obtain the following corollary.

Corollary 7 CONSISTENCY*,* k-BRAVE REASONING*,* BRAVE REASONING*,* k-SKEPTICAL REASONING*, and* SKEPTICAL REASONING *are* para-NP*-hard when parameterized by the following combined parameter*

1. $\text{maxsize}^r_{H,B^+,B^-} + \text{maxocc}^r_{H,B^+,B^-} + \#\text{at}_{B^+} + |P_c| + \text{maxsize}^c_{B^-} + ||P_c||$

CONSISTENCY*,* k-BRAVE REASONING*,* BRAVE REASONING*,* k-SKEPTICAL REASONING*, and* SKEPTICAL REASONING *are* W[2]*-hard when parameterized the following combined parameters*

2. $k + \text{maxsize}^r_{B^-} + \text{maxsize}^c_{B^+} + \text{maxsize}^c_{B^-}$

3. $k + \text{maxsize}^r_{B^+} + \#\text{at}_{B^+} + |P_c| + \text{maxsize}^c_{B^-} + ||P_c||$

4. $k + \text{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \text{maxocc}^r_{H,B^+,B^-} + \#\text{at}_{B^+} + |P_r|$

CONSISTENCY*,* k-BRAVE REASONING*,* BRAVE REASONING*,* k-SKEPTICAL REASONING*, and* SKEPTICAL REASONING *are* W[1]*-hard when parameterized the following combined parameters*

5. $k + \text{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \text{maxocc}^r_{H,B^-} + |P_c| + \text{maxsize}^c_{B^-} + ||P_c||$

6. $\text{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \text{maxocc}^r_{H,B^+,B^-} + |P_c| + \text{maxsize}^c_{B^-} + ||P_c||$

7. $k + \text{maxsize}^r_{B^+} + \#\text{non-Horn}^r + \text{maxocc}^r_{H,B^+,B^-} + |P_r| + \text{maxsize}^c_{B^-}$

Problem	Parameter	Result	Reference
k-CONSISTENCY	$\#\text{at}_H$	FPT	Obs. 19 (p. 24)
	$\text{maxsize}_H + \lvert P_r \rvert$	FPT	Cor. 20 (p. 25)
	$k + \text{maxsize}^r_{H,B^-}$	FPT	Thm. 22 (p. 25)
	$\#\text{non-Horn}^r + \text{maxsize}^r_{H,B^-}$	FPT	Thm. 22 (p. 25)
	$k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$	FPT	Thm. 29 (p. 30)
	$\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$	FPT	Thm. 29 (p. 30)
	$k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \lvert P_c \rvert + \#\text{at}_{B^-}$	FPT	Thm. 29 (p. 30)
	$\#\text{at}_{B^+} + \#\text{non-Horn}^r + \lvert P_c \rvert + \#\text{at}_{B^-}$	FPT	Thm. 29 (p. 30)
	$\lvert P_r \rvert + \lvert P_c \rvert + \#\text{at}_{B^-}$	FPT	Thm. 29 (p. 30)
	$\text{maxsize}^r_{H,B^-} + \#\text{non-Horn}^r$	FPT	Thm. 32 (p. 31)
	$\lvert P_r \rvert$	XP	Thm. 34 (p. 32)
	k	XP	Obs. 16 (p. 23)
	$\text{sb}_{\textbf{Horn}}$	FPT	Cor. 13 (p. 23)
	$\text{sb}_{\textbf{Strat}}$	FPT	Cor. 13 (p. 23)
	$\text{sb}_{\textbf{no-DBEC}}$	W[2]	Cor. 13 (p. 23)
k-CONSISTENCY restricted to programs from **Pos+Cons**	$k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-}$	FPT	Thm. 27 (p. 29)
	$\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-}$	FPT	Thm. 27 (p. 29)
	$k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \lvert P_c \rvert$	FPT	Thm. 27 (p. 29)
	$\#\text{at}_{B^+} + \#\text{non-Horn}^r + \lvert P_c \rvert$	FPT	Thm. 27 (p. 29)
	$\lvert P_r \rvert + \lvert P_c \rvert$	FPT	Thm. 27 (p. 29)

Table 4: Summary of membership results for k-CONSISTENCY and k-CONSISTENCY when restricted to programs from **Pos+Cons**.

Problem	Parameter	Result	Reference
k-BRAVE REASONING	$\#\text{at}_H$	FPT	Obs. 19 (p. 24)
	$\text{maxsize}_H + \lvert P_r \rvert$	FPT	Cor. 20 (p. 25)
	$k + \text{maxsize}^r_{H,B^-}$	FPT	Cor. 23 (p. 28)
	$\#\text{non-Horn}^r + \text{maxsize}^r_{H,B^-}$	FPT	Cor. 23 (p. 28)
	$k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$	FPT	Cor. 30 (p. 31)
	$\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$	FPT	Cor. 30 (p. 31)
	$k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \lvert P_c \rvert + \#\text{at}_{B^-}$	FPT	Cor. 30 (p. 31)
	$\#\text{at}_{B^+} + \#\text{non-Horn}^r + \lvert P_c \rvert + \#\text{at}_{B^-}$	FPT	Cor. 30 (p. 31)
	$\lvert P_r \rvert + \lvert P_c \rvert + \#\text{at}_{B^-}$	FPT	Cor. 30 (p. 31)
	$\text{maxsize}^r_{H,B^-} + \#\text{non-Horn}^r$	FPT	Cor. 33 (p. 32)
	$\lvert P_r \rvert$	XP	Obs. 34 (p. 32)
	k	XP	Obs. 16 (p. 23)
	$\text{sb}_{\textbf{Horn}}$	FPT	Cor. 13 (p. 23)
	$\text{sb}_{\textbf{Strat}}$	FPT	Cor. 13 (p. 23)
	$\text{sb}_{\textbf{no-DBEC}}$	W[2]	Cor. 13 (p. 23)
k-SKEPTICAL REASONING	$\#\text{at}_H$	FPT	Obs. 19 (p. 24)
	$\text{maxsize}_H + \lvert P_r \rvert$	FPT	Cor. 20 (p. 25)
	$\lvert P_r \rvert$	XP	Obs. 34 (p. 32)
	k	XP	Obs. 16 (p. 23)
	$\text{sb}_{\textbf{Horn}}$	FPT	Cor. 13 (p. 23)
	$\text{sb}_{\textbf{Strat}}$	FPT	Cor. 13 (p. 23)
	$\text{sb}_{\textbf{no-DBEC}}$	W[2]	Cor. 13 (p. 23)

Table 5: Summary of membership results for k-BRAVE REASONING and k-SKEPTICAL REASONING.

5 Membership Results

In this section, we present several novel tractability results for k-CONSISTENCY, which are summarized in Table 4 and 5. Further, we state conditions under which we can extend known results for the main ASP problems where k can be arbitrarily large.

First, we need some definitions.

Definition 8 *An* ASP parameter *is a function p that assigns to every program P some non-negative integer $p(P)$.*

Problem:	k-ENUM
Input:	A program P, an atom $a \in \mathrm{at}(P)$, and an integer k.
Task:	List all answer sets of size at most k of P.

We refer to the problems as ENUM if the integer k can be arbitrarily large.

The following proposition states that a fixed-parameter tractability result for the ENUM problem when parameterized by some parameter directly extends to a fixed-parameter tractability result with the same parameter for our main ASP problems, where we are interested only in answer sets of size at most k. Hence, known results for backdoors, see (Fichte and Szeider, 2015b), immediately apply to our main ASP problems for answer sets of size at most k.

Proposition 9 *Let p be an ASP parameter. If the problem* ENUM *is fixed-parameter tractable when parameterized by p, then for every problem $L \in \{k$-CONSISTENCY, k-BRAVE REASONING, k-SKEPTICAL REASONING$\}$, L is fixed-parameter tractable when parameterized by p.*

Proof. If the problem ENUM is fixed-parameter tractable when parameterized by p, then the problem k-ENUM is fixed-parameter tractable with respect to p. Thus, we can simply enumerate all answer sets of size at most k in fpt-time and decide any of the listed problems in fpt-time. Hence, the claim holds. □

Next, we define the concept of a *truth assignment reduct*.

Definition 10 (cf. Fichte and Szeider (2015b), Definition 3.1) *Let P be a program, $M \subseteq \mathrm{at}(P)$, and $N \subseteq \mathrm{at}(P) \setminus M$. The* truth assignment reduct *of P under (M,N) is the logic program $P_{M,N}$ obtained from P by*

(i) removing all rules r with $H(r) \cap M \neq \emptyset$;

(ii) removing all rules r with $B^+(r) \cap N \neq \emptyset$;

(iii) removing all rules r with $B^-(r) \cap M \neq \emptyset$;

(iv) removing from the heads and negative bodies of the remaining rules all atoms a with $a \in N$;

(v) removing from the positive bodies of the remaining rules all atoms a with $a \in M$.

Definition 11 *Let $\mathscr{C}$ be a class of programs. A set X of atoms is a* strong $\mathscr{C}$-backdoor *of a program P if $P_{M,N} \in \mathscr{C}$ for all truth assignments $\tau \in 2^X$, $M = \tau^{-1}(1)$, and $N = \tau^{-1}(1)$. For a program P let $\mathrm{sb}_{\mathscr{C}}(P)$ denote the size of a smallest strong $\mathscr{C}$-backdoor. A class $\mathscr{C}$ of programs is* enumerable *if for each $P \in \mathscr{C}$ we can compute $\mathrm{AS}(P)$ in polynomial time.*

Proposition 12 (Fichte and Szeider (2015b)) *Let $\mathscr{C}$ be an enumerable class of normal programs. The problem* ENUM *is fixed-parameter tractable when parameterized by the size of a strong $\mathscr{C}$-backdoor.*

Corollary 13 *Let $\mathscr{C}$ be an enumerable class of normal programs. Every problem $L \in \{k$-*CONSISTENCY*, k-*BRAVE REASONING*, k-*SKEPTICAL REASONING*$\}$ is fixed-parameter tractable when parameterized by the size of a strong $\mathscr{C}$-backdoor.*

The following propositions states that if the parameter yields a less restrictive result, namely, only the decision problems are fixed-parameter tractable when parameterized by some fixed parameter and the parameter is not effected by a standard "at most k" construction using a sequential counter, then our main ASP problems for answer sets of size at most k are fixed-parameter tractable when parameterized by a combined parameter that consists of the parameter together with the size k.

Definition 14 *Let p be an ASP parameter. Then we call p* counter-preserving *if $p(P) = f(p(P_k))$ for some computable function f, an integer k and $P'_k := P \cup \{\, \bot \leftarrow \neg c_{1,k+1} \,\} \cup \{\, c_{i,j} \leftarrow c_{i+1,j}, a_i;\ c_{i,j} \leftarrow c_{i+1,j} : 1 \leq i \leq n, 0 \leq j \leq k+1 \,\} \cup \{\, c_{n+1,0} \leftarrow \top \,\}$ where $a_1, \ldots, a_n$ are the atoms of P.*

Proposition 15 *Let p be counter-preserving ASP parameter. If the problem $L \in \{$*CONSISTENCY*,* BRAVE REASONING*,* SKEPTICAL REASONING*$\}$ is fixed-parameter tractable when parameterized by p, then its corresponding problem k-L, which decides the question of L when restricted to answer sets of size at most k, is fixed-parameter tractable when parameterized by p.*

Proof. Let P be a program. We restrict the decision question to answer sets of size at most k by means of a simple counter. Therefore, apply the construction from Definition 14, which uses a standard approach as described in the literature (Gebser et al., 2012) to ensure that at most k atoms are set to true and hence belong to an answer set of P. In P_k we introduce auxiliary atoms $c_{i,j}$ for $1 \leq i \leq n+1$ and $0 \leq j \leq k+1$ resulting in $\mathscr{O}(n \cdot k)$ additional auxiliary atoms and $\mathscr{O}(n \cdot k)$ additional rules. We can then simply decide L on P_k instead of P and obtain the result for our initial problem k-L. Since L is fixed-parameter tractable, $p(P) = p(P_k)$, and $\|P_k\|$ is polynomial in $n \cdot k$, the overall construction gives an fpt-algorithm with respect to k. Hence, the proposition sustains. □

Observation 16 *For each problem $L \in \{k$-*CONSISTENCY*, k-*BRAVE REASONING*, k-*SKEPTICAL REASONING*$\}$ we have $L \in \mathrm{XP}$ when parameterized by k.*

ALGORITHM 1: ALL-SIZE-K-MOD(P,k,M)

Input: A program P and an integer k.
Output: A family of sets of size at most k.

if $k \leq 0$ *or there is some rule* $\bot \in P$ **then return** $\emptyset$
if $(H(r) \cup B^-(r)) \cap M$ *or* $B^+(r) \setminus M \neq \emptyset$ *for every* $r \in P$ **then return** $\{M\}$
foreach $r \in R$ **do**
  foreach $a \in H(r)$ **do**
    $\mathsf{M} := \mathsf{M} \cup$ ALL-SIZE-K-MOD$(P_{M,\emptyset}, k, M \cup \{a\})$
return M

Proof. Let P be a program, $n = \text{at}(P)$, and k some integer. Then, we have at most $\sum_{i=1}^{k} \binom{n}{k}$ answer sets of size at most k. For each of these answer set candidates, the minimality check can be done in time $\mathcal{O}(2^k)$ by first checking whether the candidate is a model and then try all smaller models. Since it holds that $\binom{n}{k} \leq \frac{n^k}{k!}$, the algorithm runs in time $\mathcal{O}(n^k)$. □

Before we are able to show the next two fpt-results, we need to define answer set candidates of the original program.

Definition 17 (cf. Fichte and Szeider (2015b), Definition 3.5) *Let P be a program, M and N be a set of atoms. We define*

$$\text{AS}(P,M,N) = \{\, O \cup \tau^{-1}(1) : \tau^{-1}(1) \in 2^{(M \cup N) \cap \text{at}(P)}, O \in \text{AS}(P_{M,N}) \,\}.$$

Lemma 18 *Let P be a program and $H_p = \bigcup_{r \in P} H(r)$. Then,* $\text{AS}(P) \subseteq \{\, \text{AS}(P,M,N) : M \in 2^{H_p}, N = H_P \setminus M \,\}$.

Proof. The lemma is a special case of Lemma 3.6 in earlier work by Fichte and Szeider (2015b) where we simply set $X := M \cup N$. □

Observation 19 *For each problem $L \in \{k$-CONSISTENCY, k-BRAVE REASONING, k-SKEPTICAL REASONING$\}$ we have L is fixed-parameter tractable when parameterized by* $\#\text{at}_H$.

Proof. First, we show the result for k-CONSISTENCY. Therefore, let $h := \#\text{at}_H$. By Statement 1 of Observation 2 for every answer set $M \in \text{AS}_k(P)$ holds that $M \subseteq \bigcup_{r \in H(r)} H(r)$. Hence, we use a simple bounded search tree approach. We construct a complete binary search tree T of depth h. Therefore,

(1) we label the root of the tree with the triple $(P, \emptyset, \emptyset)$.

(2) Then, we label the remaining nodes of the tree recursively as follows: Let (R,M,N) be the label of a node t of T whose two children are not labeled yet. Choose an atom $a \in \text{at}(P)_H \setminus (M \cup N)$.

  (i) Label the left child of t with $(R_{M,N}, M \cup \{a\}, N)$.

(ii) Label the right child of t with $(R_{M,N}, M, N \cup \{a\})$.

(3) If there exists a node labeled with (R, M, N) such that R has no rules then M is a model of P, it remains to check whether there is some $M' \subsetneq M$ such that M' is a model of P^M. Therefore,

(a) we check for each atom $m \in M$ whether $M \setminus \{m\}$ is still a model of P^M; if so, we discard M; and

(b) we check whether $|M| \leq k$; if the answer is no, we can discard M otherwise, M is an answer set of P.

Since the depth of T is bounded by h, the size of M is at most h. We conclude that the above algorithm solves the problem k-CONSISTENCY in time $\mathcal{O}(2^h \cdot h \cdot n^c)$ for $n = |\text{at}(P)|$ and some constant c. Notice that this results trivially extends to k-BRAVE REASONING by adding a rule that consists of an empty head, an empty positive body and a negative body that contains only the atom we are interested in. For k-SKEPTICAL REASONING we can simply traverse the entire bounded search tree and obtain the same running time. □

As #at(P) is bounded by $\text{maxsize}_H \cdot |P_r|$ the next result follows directly from the previous one.

Corollary 20 *Let $L \in \{k$-CONSISTENCY, k-BRAVE REASONING, k-SKEPTICAL REASONING$\}$. Then, L is fixed-parameter tractable when parameterized by* $\text{maxsize}_H + |P_r|$.

We now proceed to two tractability results that can be obtained by a reduction to WMMSAT. Lackner and Pfandler (2012b) presented several fixed-parameter tractability results that turn out to be useful for showing fixed-parameter tractability for k-CONSISTENCY. To proof the next theorem we need the following results.

Proposition 21 (Lackner and Pfandler (2012b)) WMMSAT *is fixed-parameter tractable when parameterized by at least one of the following combined parameters*

1. $k + d^+$

2. $d^+ + h$

Now we are ready to show two fixed-parameter tractability results.

Theorem 22 k-CONSISTENCY *is fixed-parameter tractable when parameterized by at least one of the following combined parameters*

1. $k + \text{maxsize}^r_{H,B^-}$

2. $\#\text{non-Horn}^r + \text{maxsize}^r_{H,B^-}$

Proof. The main idea of the proof is a reduction to WMMSAT. This reduction runs in linear time and preserves all necessary parameters.

In more detail, for a program P this reduction consists of two reductions:

1. from P we construct in linear time a program $P^* = P' \cup P^{\text{supset}}$
2. from P^* we construct in linear time an instance of WMMSAT.

With these reductions in hand we will then be able to proof the following claims. For this, let P be a program, $M \subseteq \text{at}(P)$ and $M' = \{\, a' \mid a \ \in M \,\}$.

Claim 1: From P, we can construct in linear time the program P^* such that M is an answer set of P if and only if $M \cup M'$ is a minimal model of P' and $M \cup M'$ is a model of P^{supset}.

Claim 2: From P^*, we can construct in linear time two CNF formulas $F_{\text{P'}}$ and F_{supset} such that $M \cup M'$ is a minimal model of P' and $M \cup M'$ is a model of P^{supset} if and only if $V_{M \cup M'}$ is a minimal model of $F_{\text{P'}}$ and $V_{M \cup M'}$ is a model of F_{supset} where $V_{M \cup M'} := \{\, v[a] : a \in M \,\} \cup \{\, v[a'] : a' \in M' \,\}$.

From these claims it immediately follows that P is a yes-instance, i.e., M is an answer set of P, if and only if the constructed WMMSAT instance with $\varphi = F_{\text{P'}}$ and $\pi = F_{\text{supset}}$ is a yes-instance, i.e., $V_{M \cup M'}$ is a minimal model of φ and $V_{M \cup M'}$ is a model of π.

Let us now proof Claim 1. For a set X, with the help of the macro $(X)'$ we denote the set $\{a' \mid a \in X\}$. Let us now construct the following set of rules.

$$
\begin{aligned}
P^{\text{min}} &:= \left\{\, H(r) \leftarrow B^+(r), \left(B^-(r)\right)' : r \in P \,\right\} \\
P^{\text{subset}} &:= \left\{\, a' \leftarrow a : a \in \text{at}(P) \,\right\} \\
P' &:= P^{\text{min}} \cup P^{\text{subset}} \\
P^{\text{supset}} &:= \left\{\, a \leftarrow a' : a \in \text{at}(P) \,\right\} \\
P^* &:= P' \cup P^{\text{supset}}
\end{aligned}
$$

It is straightforward to see that this reduction runs in linear time in the size of P. Now we show now that M is an answer set of P if and only if $M \cup M'$ is a minimal model of P'.
($\Rightarrow$): Let M be an answer set of P. By Observation 3 it suffices to show that $M \cup M'$ is a minimal model of $(P')^{M \cup M'}$. Hence, for proof by contradiction assume that there is some $N \subsetneq (M \cup M')$ such that N is a model of $(P')^{M \cup M'}$.

(i) Let $b \in M \setminus N$: By definition of an answer set and the rules in P^{subset}, we have that $b \in M$ implies $b' \in M \cup M'$. However, then there is some $N' \subseteq M$ and $b \in M \setminus N'$ such that N' is a model of P^M, which contradicts the assumption that M is an answer set of P.

(ii) Let $b' \in M' \setminus N$: Since every atom $a' \in \text{at}(P)$ occurs only in the head of rules in P^{subset}, we conclude by Statement 1 of Observation 2 that there is a corresponding atom $b \in M$. Hence, we proceed as in Case (i), which yields a contradiction.

($\Leftarrow$): Let $M \cup M'$ be an answer set of $(P')^{M \cup M'}$. For proof by contradiction assume that there is some $N \subsetneq M$ such that N is a model of P^M. However, by Statement 1 of Observation 2 there exists a set $N' \subseteq M \cup M'$ such that N' is a minimal model of $(P')^{M \cup M'}$, which contradicts the assumption that $M \cup M'$ is an answer set of $(P')^{M \cup M'}$.

Now we show that if $M \cup M'$ is a minimal model of P' it follows that $M \cup M'$ is a model of P^{supset}. For this, let $M \cup M'$ be a minimal model of P'. The rules in P^{subset} ensure by definition of an answer set that if $a \in M \cup M'$ then $a' \in M \cup M'$. By Statement 1 of Observation 2 we also obtain that if $a' \in M \cup M'$ then $a \in M \cup M'$. Consequently, $M \cup M'$ satisfies each rule in P^{subset}. Hence, Claim 1 holds.

We now proceed to show Claim 2. From program P^* we construct the following CNF formulas.

$$
\begin{aligned}
F_{\min} &:= \bigwedge_{r \in P^{\min}} \left(\bigvee_{a \in B^+(r)} \neg v[a] \vee \bigvee_{a \in H(r)} v[a] \right) && \text{(encodes } P^{\min}\text{)}\\
F_{\text{subset}} &:= \bigwedge_{r \in P^{\text{subset}}} \left(\bigvee_{a \in B^+(r)} \neg v[a] \vee \bigvee_{a \in H(r)} v[a] \right) && \text{(encodes } P^{\text{subset}}\text{)}\\
F_{\text{P'}} &:= F_{\min} \wedge F_{\text{subset}} && \text{(encodes } P'\text{)}\\
F_{\text{supset}} &:= \bigwedge_{r \in P^{\text{supset}}} \left(\bigvee_{a \in B^+(r)} \neg v[a] \vee \bigvee_{a \in H(r)} v[a] \right) && \text{(encodes } P^{\text{supset}}\text{)}
\end{aligned}
$$

The WMMSAT instance is then given by $\varphi = F_{\text{P'}}$ and $\pi = F_{\text{supset}}$. It is straight-forward to see that $M \cup M'$ is a minimal model of P' and $M \cup M'$ is a model of P^{supset} if and only if $V_{M \cup M'}$ is a minimal model of $F_{\text{P'}}$ and $V_{M \cup M'}$ is a model of F_{supset} where $V_{M \cup M'} := \{ v[a] : a \in M \} \cup \{ v[a'] : a' \in M' \}$. Hence, Claim 2 holds.

It remains to observe that the reduction preserves all parameters.

- $\text{maxsize}^r_{H,B^-}$: Let $d \geq 2$ be some integer. Moreover, assume that $\text{maxsize}^r_{H,B^-} \leq d$, by construction of $F_{\text{P'}}$ each clause in F_{subset} contains at most 1 positive literal and the maximum number of positive literals in a clause of $F_{\min}$ is at most d. Moreover, each clause in F_{supset} contains at most 1 positive literal. Hence, maximum number of positive literals in each clause of the resulting formulas is at most d.

- k: Let $k \geq 0$ be some integer. Moreover, assume that $|M| \leq k$. By construction of $F_{\text{P'}}$, $M \subseteq \text{at}(P)$ is an answer set of P if and only if $V_{M \cup M'}$ is a minimal model of F_{P} and $V_{M \cup M'}$ is a model of F_{supset}. Hence, we have $|V_{M \cup M'}| \leq 2k$ by construction. Consequently, the maximum weight of the minimal model of F_{P} is bounded by $2k$.

- #non-Horn^r: Let $h \geq 0$ be some integer and assume that #non-Horn$^r \leq h$. By construction of F_{subset} and F_{supset} contain only Horn clauses. Moreover, a rule is not Horn if and only if the corresponding clause in $F_{\min}$ is not Horn. Hence, h provides an upper bound for the number of non-Horn clauses of $F_{\min}$ and thus of $F_{\text{P'}}$ and F_{supset}.

Hence, the statement of the theorem holds. □

Corollary 23 *The problem k-BRAVE REASONING is fixed-parameter tractable when parameterized by at least one of the following combined parameters*

1. $k + \text{maxsize}^r_{H,B^-}$
2. $\#\text{non-Horn}^r + \text{maxsize}^r_{H,B^-}$

We denote by $\text{Mod}_k(P)$ the set of all models of P where $|M| \leq k$ for every $M \in \text{Mod}_k(P)$.

Observation 24 $\text{AS}_k(P) \subseteq Mod_k(P)$ *holds for every positive-body-free program P and some integer* $k \geq 0$.

Proof. Follows trivially from the definitions. □

In the following, we consider the parameterized complexity of the problem k-CONSISTENCY when the input is restricted to programs from **Pos+Cons** and establish new fpt results. In fact, Statement 2 in Theorem 6 already states W[2]-hardness for k-CONSISTENCY when restricted to programs from **Pos+Cons**. The subsequent proposition establishes, in addition, that the problem CONSISTENCY remains Σ_2^P-complete under the restrictions.

Proposition 25 (Truszczyński (2011)) *The problem* CONSISTENCY *is* Σ_2^P*-complete when the input is restricted to programs from* **Pos+Cons**.

Proof. The statement has already been established by Truszczyński (2011)[Theorem 2]. More precisely, by definition of programs in **Pos+Cons**, we have $\text{maxsize}^r_{B^-} = 0$ and maxsize_H and $\text{maxsize}^r_{B^+}$ are unbounded, $\text{maxsize}^c_{B^+}$ and $\text{maxsize}^c_{B^-}$ are unbounded. However, CONSISTENCY is already Σ_2^P-complete when the input is restricted to programs where $\text{maxsize}^r_{B^-} = 0$ and maxsize_H and $\text{maxsize}^r_{B^+}$ are unbounded, $\text{maxsize}^c_{B^+}$ is unbounded and $\text{maxsize}^c_{B^-} = 0$. □

In the next proposition, we summarize the fixed-parameter tractability results for WMMSAT by Lackner and Pfandler (2012b).

Proposition 26 (Lackner and Pfandler (2012b)) WMMSAT *is fixed-parameter tractable when parameterized by at least one of the following combined parameters*

1. $k + v^- + p + d^+_\pi$
2. $v^- + h + d^+_\pi$
3. $k + v^- + p + |\pi|$
4. $v^- + h + |\pi|$

5. $|\varphi| + |\pi|$

Lemma 27 *k*-CONSISTENCY *when restricted to programs from* **Pos+Cons** *is fixed-parameter tractable and parameterized by at least one of the following combined parameters*

1. $k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-}$
2. $\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-}$
3. $k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + |P_c|$
4. $\#\text{at}_{B^+} + \#\text{non-Horn}^r + |P_c|$
5. $|P_r| + |P_c|$

Proof. In order to decide k-CONSISTENCY when the input is restricted to programs from **Pos+Cons**, we give a reduction to WMMSAT, which preserves all parameters considered in the statement. Therefore, we use ideas from the construction in the proof of Theorem 6 for the opposite direction. Let (P,k) be an instance of k-CONSISTENCY where $P \in$ **Pos+Cons**. We now construct an instance (φ, π, k) of WMMSAT as follows. The variables of the CNF formulas φ and π will consist of a variable for each atom of P. Then for a rule $r \in P$ we let $C(r) := \{ x_a : a \in H(r) \} \cup \{ \neg x_a : a \in B^+(r) \}$. Further, we define $\varphi := \{ C(r) : r \in P, H(r) \neq \emptyset \}$ and $\pi := \{ C(r) : r \in P, H(r) = \emptyset \}$. Then, we show that φ has a minimal model M of size at most k such that M is also a model of π if and only if P has an answer set of size at most k. We can use the exact same construction as in the proof of Theorem 6 to establish the statement, since there program P_π consists only of constraint rules and program P_φ consists only of non-constraint rules.

Next, we observe that our reduction preserves the parameters:

- k: directly corresponds to the maximum weight of a minimal model (k)
- $\text{maxsize}^r_{B^+}$: directly corresponds to the maximum negative clause size (d^+)
- $\#\text{non-Horn}^r$: directly corresponds to the number of non-horn clauses (h)
- maxocc^r_{H,B^-}: directly corresponds to the maximum number of positive occurrences of a variable in φ (p)
- $\#\text{at}_{B^+}$: directly corresponds to the number of variables that occur as negative literals in φ (v^-) or in π (v^-)
- $|P_r|$: directly corresponds to the number of clauses in φ ($|\varphi|$)
- $|P_c|$: directly corresponds to the number of clauses in π ($|\pi|$)
- $\text{maxsize}^c_{B^-}$: directly corresponds to the maximum positive clause size in π (d^+_π)

Finally, fixed-parameter tractability follows from Proposition 26. This concludes the proof. □

Remark 28 *We would like to mention that, using the reductions above, instances from* WMMSAT *and* k-CONSISTENCY *restricted to* **Pos+Cons** *coincide. More precisely, the proofs give a linear time reduction that transforms an instance from* WMMSAT *into an instance of* **Pos+Cons** *from* k-CONSISTENCY *and vice versa.*

Theorem 29 *The problem* k-CONSISTENCY *is fixed-parameter tractable and parameterized by at least one of the following combined parameters*

1. $k + \#\mathrm{at}_{B^+} + \mathrm{maxocc}^r_{H,B^-} + \mathrm{maxsize}^c_{B^-} + \#\mathrm{at}_{B^-}$
2. $\#\mathrm{at}_{B^+} + \#\text{non-Horn}^r + \mathrm{maxsize}^c_{B^-} + \#\mathrm{at}_{B^-}$
3. $k + \#\mathrm{at}_{B^+} + \mathrm{maxocc}^r_{H,B^-} + |P_c| + \#\mathrm{at}_{B^-}$
4. $\#\mathrm{at}_{B^+} + \#\text{non-Horn}^r + |P_c| + \#\mathrm{at}_{B^-}$
5. $|P_r| + |P_c| + \#\mathrm{at}_{B^-}$

Proof. In order to decide k-CONSISTENCY, we give an fpt-algorithm that uses the fpt-results established in Theorem 27 for k-CONSISTENCY when the input is restricted to programs from **Pos+Cons**. Therefore, let (P, ℓ) be an instance of k-CONSISTENCY, $N := \cup_{r \in P} \left(\mathrm{at}_{B^-,r}\right)$, $\tau \in 2^N$, $M_1 := \tau^{-1}(1)$, and $M_0 := \tau^{-1}(0)$. Further, we define $P^c_{M_1,M_0} := \{\, \bot \leftarrow \neg a : a \in M_1 \,\} \cup \{\, \leftarrow a : a \in M_0 \,\}$, use P_{M_1,M_0} as defined in Definition 10, and let $P[\tau] := P_{M,N} \cup P^c_{M,N}$.

Then, the program P has an answer set of size at most k if and only if at least one program $P[\tau]$ has an answer set of size at most k. Therefore, observe that $\mathrm{AS}(P) \subseteq \{\, \mathrm{AS}(P[\tau]) : \tau \in 2^N \,\}$, which is a special case of Lemma 3.6 in earlier work by Fichte and Szeider (2015b) where we simply set $X := M_0 \cup M_1$. It is easy to see from the definitions of an answer set and the construction of $P[\tau]$ that the opposite direction $\{\, \mathrm{AS}(P[\tau]) : \tau \in 2^N \,\} = \tau^{-1}(0) \,\} \subseteq \mathrm{AS}(P)$ also holds. In this way, we give a reduction to $2^{(\#\mathrm{at}_{B^-}+1)}$ many instances of k-CONSISTENCY that consists of $2^{(\#\mathrm{at}_{B^-}+1)}$ many subprograms by constructing "partial" GL reducts under a set M_1, which consists of atoms that we have set to true, and a set M_0, which consists of atoms that we have set to false, together with constraints that enforce that any minimal model M of the GL reduct satisfies that atoms in M_1 belong to M and atoms in M_0 do not belong to M. It remains to observe that our reduction preserves the parameters:

- k remains unaffected,
- $\mathrm{maxocc}^r_{H,B^-}(P[\tau]) \leq \mathrm{maxocc}^r_{H,B^-}(P)$,
- $\mathrm{maxsize}^c_{B^-}(P[\tau]) = \max\{\, \mathrm{maxsize}^c_{B^-}(P), 1 \,\}$,
- $\#\text{non-Horn}^r(P[\tau]) = \#\text{non-Horn}^r(P)$,

- $\#\text{at}_{B^+}(P[\tau]) = \#\text{at}_{B^+}(P) + \#\text{at}_{B^-}(P)$,
- $|(P[\tau])_r| \leq |P_r| + \#\text{at}_{B^-}(P)$, and
- $|(P[\tau])_c| \leq |P_c| + \#\text{at}_{B^-}(P)$.

Since our algorithm constructs $2^{(\#\text{at}_{B^-}+1)}$ many programs that can be solved in fpt-time according to Theorem 27, our algorithm runs in fpt-time. Hence, the theorem follows. □

Corollary 30 *The problem k-*BRAVE REASONING *is fixed-parameter tractable and parameterized by at least one of the following combined parameters*

1. $k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$
2. $\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$
3. $k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + |P_c| + \#\text{at}_{B^-}$
4. $\#\text{at}_{B^+} + \#\text{non-Horn}^r + |P_c| + \#\text{at}_{B^-}$
5. $|P_r| + |P_c| + \#\text{at}_{B^-}$

The reductions in the proofs of Theorems 2 and 27 already state that ASP and WMMSAT are very related with respect to the consistency and the considered reasoning problems. Hence, it seems reasonable to apply and extend concepts from WMMSAT by Lackner and Pfandler (2012b) to ASP. However, since answer sets additionally require minimality with respect to the GL reduct of the given program, we need to parameterize additionally in the number of negative atoms that occur in rules (both constraint and non-constraint rules) of the given program. Particularly, we do not have a direct counterpart of the concept of a compact representation for atoms in the head (see the concept of SSMs in the work by Lackner and Pfandler (2012b)) if the positive body is empty, but the negative body is not empty. Note that our reductions make certain concepts of parameters in the setting of answer set programming such as acyclicity-based backdoors by (Fichte and Szeider, 2015a) directly accessible to WMMSAT.

Proposition 31 (Lackner and Pfandler (2012b), The. 3) WMMSAT *is fixed-parameter tractable when parameterized by the maximum number of positive literals in a clause and the number of non-Horn clauses.*

Theorem 32 *k-*CONSISTENCY *when parameterized by* $\text{maxsize}^r_{H,B^-} + \#\text{non-Horn}^r$ *is fixed-parameter tractable.*

Proof. We use the reduction defined in the proof of Lemma 22 to reduce to WMMSAT. It remains to observe that the reduction preserves the parameters. Let P be a program and $d \geq 2$ be some integer. Moreover, assume that $\text{maxsize}^r_{H,B^-} \leq d$, by construction of F_P each clause in F_subset contains at most 1 positive literal and the maximum number of positive literals in a clause of F_min is at most d. Moreover, each clause in F_supset contains at most 1 positive literal. Hence, maximum number of positive literals in each clause of the resulting formulas is at most d. □

Corollary 33 k-BRAVE REASONING *when parameterized by* $\text{maxsize}^r_{H,B^-} + \#\text{non-Horn}^r$ *is fixed-parameter tractable.*

Observation 34 *For each problem* $L \in \{k$-CONSISTENCY, k-BRAVE REASONING, k-SKEPTICAL REASONING$\}$ *we have* L *when parameterized by* $|P_r|$ *is in* XP.

Proof. Let P be a program with $k = |P_r|$ rules and n be the number of different atoms occurring in the head of a rule. By Statement 2 of Observation 2 the size of an answer set is at most k. Thus, an algorithm can check all possible $\sum_{i=1}^{k} \binom{n}{k}$ answer sets of size at most k. For each of these answer set candidates, the minimality check can be done in time $\mathcal{O}(2^k)$. Since it holds that $\binom{n}{k} \leq \frac{n^k}{k!}$, the algorithm runs in time $\mathcal{O}(n^k)$. Consequently, the observation is established. □

Finally, we conclude with an observation that states that the results trivially extend to the main ASP problems where the answer sets can be arbitrarily large if the considered parameter does not depend on the maximum size of an answer set.

Observation 35 *Let p be an ASP parameter. If the problem* $L \in \{k$-CONSISTENCY, k-BRAVE REASONING, k-SKEPTICAL REASONING$\}$ *is fixed-parameter tractable when parameterized by p and p does not depend on k, then its corresponding problem L', which decides the question of L for answer sets of arbitrary size, is fixed-parameter tractable when parameterized by p.*

Proof. The observations follows trivially by setting $k = \text{at}(P)$. □

6 Conclusion

We have identified several natural structural parameters of ASP instances (as summarized in Table 2) and carried out a fine-grained complexity analysis of the main reasoning problems in answer set programming when parameterized by various combinations of these parameters. Our study also considers the parameterized complexity of the main ASP reasoning problems while taking the size of the answer set into account. Such a restriction is particularly interesting for applications that require small solutions. We have presented various hardness (see Table 3) and membership results (see Table 4 and 5). Every hardness result of the reasoning problems when parameterized by a combined parameter also holds for any parameter that consists of a subset of the combination. Further, every fixed-parameter tractability result of the considered problems when parameterized by a combined parameter also holds for any extension of the parameter by additional structural properties (superset of the parameter combination). In that way, we have improved on the theoretical understanding by providing a novel multi parametric view on the parameterized complexity of ASP, which allows us to draw a detailed map for various combined ASP parameters.

Open Parameter Combinations Regarding k-CONSISTENCY we have identified slightly more than 80% of possible combined parameters when we consider FPT-membership and W[1]-hardness as final results that do not need further treatment. To be more precise, from $2^{17} = 131072$ possible combined parameters, only 25993 are left open for k-CONSISTENCY.

Future Work The results and concepts of this paper give rise to several research questions. For instance, it would be interesting to close the gap for the remaining parameter combinations. Therefore, we need to identify important corner cases. An interesting further research direction is to study how the parameters empirically distribute among ASP instances from the last ASP competitions, in particular, in random versus structured instances.

References

ABITEBOUL, S., HULL, R., AND VIANU, V. 1995. *Foundations of Databases: The Logical Level*, 1st ed. Addison-Wesley, Boston, MA, USA.

ALVIANO, M., DODARO, C., FABER, W., LEONE, N., AND RICCA, F. 2013. WASP: A native ASP solver based on constraint learning. In *Proceedings of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'13)*, P. Cabalar and T. Son, Eds. Lecture Notes in Computer Science, vol. 8148. Springer Verlag, Corunna, Spain, 54–66.

ANSÓTEGUI, C., BONET, M. L., LEVY, J., AND MANYA, F. 2008. Measuring the hardness of SAT instances. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI '08)*, R. C. Holte and A. E. Howe, Eds. The AAAI Press, Chicago, IL, USA, 222–228.

APT, K. R., BLAIR, H. A., AND WALKER, A. 1988. Towards a theory of declarative knowledge. *Foundations of deductive databases and logic programming*, 89–148.

ARORA, S. AND BARAK, B. 2009. *Computational Complexity: A Modern Approach*. Cambridge University Press, Cambridge.

ATSERIAS, A., FICHTE, J. K., AND THURLEY, M. 2011. Clause-learning algorithms with many restarts and bounded-width resolution. *J. Artif. Intell. Res. 40*, 353–373.

BEN-ELIYAHU, R. 1996. A hierarchy of tractable subsets for computing stable models. *J. Artif. Intell. Res. 5*, 27–52.

BEN-ELIYAHU, R. AND DECHTER, R. 1994. Propositional semantics for disjunctive logic programs. *Ann. Math. Artif. Intell. 12,* 1, 53–87.

BIERE, A., HEULE, M., VAN MAAREN, H., AND WALSH, T., Eds. 2009. *Handbook of Satisfiability*. Frontiers in Artificial Intelligence and Applications, vol. 185. IOS Press, Amsterdam, Netherlands.

BLIEM, B., ORDYNIAK, S., AND WOLTRAN, S. 2016. Clique-width and directed width measures for answer-set programming. In *Proceedings of the 22st Eureopean Conference on Artificial Intelligence (ECAI'16)*, M. Fox and G. Kaminka, Eds. Frontiers in Artificial Intelligence and Applications, vol. 285. IOS Press, The Hague, Netherlands, 1105–1113. Extended version `CoRR:1606.09449`.

BODLAENDER, H. L. 1993. A tourist guide through treewidth. *Acta Cybernetica 11,* 1-2, 1–22.

BOMANSON, J., GEBSER, M., AND JANHUNEN, T. 2014. Improving the normalization of weight rules in answer set programs. In *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA'14)*, E. Fermé and J. Leite, Eds. Lecture Notes in Computer Science, vol. 8761. Springer Verlag, Funchal, Madeira, Portugal, 166–180.

BRASS, S. AND DIX, J. 1998. Characterizations of the disjunctive well-founded semantics: Confluent calculi and iterated GCWA. *J. Automated Reasoning 20*, 143–165.

CHANDRA, A. K. AND HAREL, D. 1985. Horn clause queries and generalizations. *J. Logic Programming 2,* 1, 1–15.

COURCELLE, B. AND OLARIU, S. 2000. Upper bounds to the clique width of graphs. *Discr. Appl. Math. 101,* 1–3, 77–114.

CYGAN, M., FOMIN, F. V., KOWALIK, L., LOKSHTANOV, D., DÁNIEL MARX, M. P., PILIPCZUK, M., AND SAURABH, S. 2015. *Parameterized Algorithms*. Springer Verlag.

DOWLING, W. F. AND GALLIER, J. H. 1984. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. *J. Logic Programming 1,* 3, 267–284.

DOWNEY, R. G. AND FELLOWS, M. R. 1999. *Parameterized Complexity*. Monographs in Computer Science. Springer Verlag, New York, NY, USA.

DOWNEY, R. G. AND FELLOWS, M. R. 2013. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer Verlag, London, UK.

DOWNEY, R. G., FELLOWS, M. R., AND REGAN, K. W. 1998. Descriptive complexity and the W hierarchy. *DIMACS: Series in Discrete Mathematics and Theoretical Computer Science 39*, 119–134.

DOWNEY, R. G., FELLOWS, M. R., AND STEGE, U. 1999. Parameterized complexity: A framework for systematically confronting computational intractability. In *Contemporary Trends in Discrete Mathematics: From DIMACS and DIMATIA to the Future*, R. L. Graham, J. Kratochvíl, J. Nešetřil, and F. S. Roberts, Eds. DIMACS: Series in Discrete Mathematics and Theoretical Computer Science, vol. 49. American Mathematical Society, 49–99.

DRESCHER, C., GEBSER, M., GROTE, T., KAUFMANN, B., KÖNIG, A., OSTROWSKI, M., AND SCHAUB, T. 2008. Conflict-driven disjunctive answer set solving. In *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR'08)*, G. Brewka and J. Lang, Eds. The AAAI Press, Sydney, NSW, Australia, 422–432.

EITER, T., FINK, M., TOMPITS, H., AND WOLTRAN, S. 2004. Simplifying logic programs under uniform and strong equivalence. In *Proceedings 7th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'04)*, I. Niemelä and V. Lifschitz, Eds. Lecture Notes in Computer Science, vol. 2923. Springer Verlag, Tempe, AZ, USA, 87–99.

EITER, T. AND GOTTLOB, G. 1995. On the computational cost of disjunctive logic programming: Propositional case. *Ann. Math. Artif. Intell. 15,* 3–4, 289–323.

FELLOWS, M. R., PFANDLER, A., ROSAMOND, F. A., AND RÜMMELE, S. 2012. The parameterized complexity of abduction. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAA'12)*, J. Hoffmann and B. Selman, Eds. The AAAI Press, Toronto, ON, Canada.

FICHTE, J. K. 2015. Backdoors to tractability of disjunctive answer set programming. Ph.D. thesis, Fakultät für Informatik an der Technischen Universität Wien.

FICHTE, J. K. AND SZEIDER, S. 2015a. Backdoors to normality for disjunctive logic programs. *ACM Trans. Comput. Log. 17,* 1 (Dec.), 7.

FICHTE, J. K. AND SZEIDER, S. 2015b. Backdoors to tractable answer-set programming. *Artificial Intelligence 220,* 0, 64–103.

FLUM, J. AND GROHE, M. 2003. Describing parameterized complexity classes. *Information and Computation 187,* 2, 291–319.

FLUM, J. AND GROHE, M. 2005. Model-checking problems as a basis for parameterized intractability. *Logical Methods in Computer Science 1,* 1, 1–36.

FLUM, J. AND GROHE, M. 2006. *Parameterized Complexity Theory*. Theoretical Computer Science, vol. XIV. Springer Verlag, Berlin.

GASPERS, S. AND SZEIDER, S. 2012. Backdoors to satisfaction. In *The Multivariate Algorithmic Revolution and Beyond*, H. Bodlaender, R. Downey, F. Fomin, and D. Marx, Eds. Lecture Notes in Computer Science, vol. 7370. Springer Verlag, Heidelberg, Germany, 287–317.

GEBSER, M., GLASE, T., SABUNCU, O., AND SCHAUB, T. 2013. Matchmaking with answer set programming. In *Proceedings of 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'13)*, P. Cabalar and T. C. Son, Eds. Lecture Notes in Computer Science, vol. 8148. Springer Verlag, Corunna, Spain, 342–347.

GEBSER, M., KAMINSKI, R., KAUFMANN, B., AND SCHAUB, T. 2011. Multi-Criteria Optimization in Answer Set Programming. In *Technical Communications of the 27th International Conference on Logic Programming (ICLP'11)*, J. Gallagher and M. Gelfond, Eds. Leibniz International Proceedings in Informatics (LIPIcs), vol. 11. Dagstuhl Publishing, Lexington, KY, USA, 1–10.

GEBSER, M., KAMINSKI, R., KAUFMANN, B., AND SCHAUB, T. 2012. *Answer Set Solving in Practice*. Morgan & Claypool.

GEBSER, M., KAUFMANN, B., AND SCHAUB, T. 2012a. Conflict-driven answer set solving: From theory to practice. *Artificial Intelligence 187-188*, 52–89.

GEBSER, M., KAUFMANN, B., AND SCHAUB, T. 2013. Advanced conflict-driven disjunctive answer set solving. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI'13)*, F. Rossi and S. Thrun, Eds. The AAAI Press, Beijing, China, 912–918.

GEBSER, M., KAUFMANN, R., AND SCHAUB, T. 2012b. Gearing up for effective ASP planning. In *Correct Reasoning – Essays on Logic-Based AI in Honour of Vladimir Lifschitz*, E. Erdem, J. Lee, Y. Lierler, and D. Pearce, Eds. Lecture Notes in Computer Science, vol. 7265. Springer Verlag, 296–310.

GEBSER, M., SCHAUB, T., AND THIELE, S. 2007. Gringo: A new grounder for answer set programming. In *Proceedings of the 9th International Conference Logic Programming and Nonmonotonic Reasoning (LPNMR'07)*, C. Baral, G. Brewka, and J. Schlipf, Eds. Lecture Notes in Computer Science, vol. 4483. Springer Verlag, Tempe, AZ, USA, 266–271.

GELDER, A. V. 1989. Negation as failure using tight derivations for general logic programs. *J. Logic Programming 6,* 1–2, 109–133.

GELFOND, M. AND LIFSCHITZ, V. 1988. The stable model semantics for logic programming. In *Proceedings of the 5th International Conference and Symposium on Logic Programming (ICLP/SLP'88)*, R. A. Kowalski and K. A. Bowen, Eds. Vol. 2. MIT Press, Seattle, WA, USA, 1070–1080.

GELFOND, M. AND LIFSCHITZ, V. 1991. Classical negation in logic programs and disjunctive databases. *New Generation Comput. 9,* 3/4, 365–386.

GOLDREICH, O. 2008. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, Cambridge.

GOMES, C. P., KAUTZ, H., SABHARWAL, A., AND SELMAN, B. 2008. Chapter 2: Satisfiability solvers. In *Handbook of Knowledge Representation*, V. L. Frank van Harmelen and B. Porter, Eds. Foundations of Artificial Intelligence, vol. 3. Elsevier Science Publishers, North-Holland, Amsterdam, Netherlands, Chapter 2, 89–134.

GOTTLOB, G., PICHLER, R., AND WEI, F. 2010. Bounded treewidth as a key to tractability of knowledge representation and reasoning. *Artificial Intelligence 174,* 1, 105–132.

GOTTLOB, G., SCARCELLO, F., AND SIDERI, M. 2002. Fixed-parameter complexity in AI and nonmonotonic reasoning. *Artificial Intelligence 138,* 1-2, 55–86.

GOTTLOB, G. AND SZEIDER, S. 2008. Fixed-parameter algorithms for artificial intelligence, constraint satisfaction and database problems. *The Computer Journal 51,* 3, 303–325.

JAKL, M., PICHLER, R., AND WOLTRAN, S. 2009. Answer-set programming with bounded treewidth. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI'09)*, C. Boutilier, Ed. Vol. 2. The AAAI Press, Pasadena, CA, USA, 816–822.

JANHUNEN, T. AND NIEMELÄ, I. 2011. Compact translations of non-disjunctive answer set programs to propositional clauses. In *Logic Programming, Knowledge Representation, and Nonmonotonic Reasoning – Essays Dedicated to Michael Gelfond on the Occasion of His 65th Birthday*, M. Balduccini and T. Son, Eds. Lecture Notes in Artificial Intelligence, vol. 6565. Springer Verlag, 111–130.

KAUFMANN, B., GEBSER, M., KAMINSKI, R., AND SCHAUB, T. 2015. clasp – a conflict-driven nogood learning answer set solver. `http://www.cs.uni-potsdam.de/clasp/`.

KLEINE BÜNING, H. AND LETTMAN, T. 1999. *Propositional logic: deduction and algorithms*. Cambridge University Press, Cambridge, New York, NY, USA.

KRONEGGER, M., PFANDLER, A., AND PICHLER, R. 2013. Parameterized complexity of optimal planning: A detailed map. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, F. Rossi, Ed. The AAAI Press, Beijing, China, 954–961.

LACKNER, M. AND PFANDLER, A. 2012a. Fixed-parameter algorithms for closed world reasoning. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI'14)*, T. Schaub, G. Friedrich, and B. O'Sullivan, Eds. Frontiers in Artificial Intelligence and Applications, vol. 242. IOS Press, Prague, Czech Republic, 492–497.

LACKNER, M. AND PFANDLER, A. 2012b. Fixed-parameter algorithms for finding minimal models. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR'12)*, T. Eiter and S. McIlraith, Eds. The AAAI Press, Rome, Italy, 85–95.

LEONE, N., PFEIFER, G., FABER, W., EITER, T., GOTTLOB, G., PERRI, S., AND SCARCELLO, F. 2006. The DLV system for knowledge representation and reasoning. *ACM Trans. Comput. Log. 7,* 3, 499–562.

LIN, F. AND ZHAO, X. 2004. On odd and even cycles in normal logic programs. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI'04)*, D. L. McGuinness and G. Ferguson, Eds. The AAAI Press, San Jose, CA, USA, 80–85.

MAREK, V. W. AND TRUSZCZYŃSKI, M. 1999. Stable models and an alternative logic programming paradigm. In *The Logic Programming Paradigm: a 25-Year Perspective*, K. R. Apt, V. W. Marek, M. Truszczyński, and D. S. Warren, Eds. Artificial Intelligence. Springer Verlag, 375–398.

MARX, D. AND PILIPCZUK, M. 2014. Everything you always wanted to know about the parameterized complexity of Subgraph Isomorphism (but were afraid to ask). In *Proceedings of the 31st International Symposium on Theoretical Aspects of Computer Science (STACS'14)*, E. W. Mayr and N. Portier, Eds. Leibniz International Proceedings in Informatics (LIPIcs), vol. 25. Dagstuhl Publishing, Lyon, France, 542–553.

MORAK, M., PICHLER, R., RÜMMELE, S., AND WOLTRAN, S. 2010. A dynamic-programming based ASP-solver. In *Proceedings of 12th European Conference on Logics in Artificial Intelligence (JELIA'10)*, T. Janhunen and I. Niemelä, Eds. Lecture Notes in Computer Science, vol. 6341. Springer Verlag, Helsinki, Finland, 369–372.

NIEDERMEIER, R. 2006. *Invitation to Fixed-Parameter Algorithms*. Oxford Lecture Series in Mathematics and its Applications, vol. 31. Oxford University Press, New York, NY, USA.

NIEMELÄ, I. 1999. Logic programs with stable model semantics as a constraint programming paradigm. *Ann. Math. Artif. Intell. 25,* 3, 241–273.

NIEMELÄ, I. AND RINTANEN, J. 1994. On the impact of stratification on the complexity of nonmonotonic reasoning. *J. Applied Non-Classical Logics 4,* 2, 141–179.

NIEMELÄ, I., SIMONS, P., AND SOININEN, T. 1999. Stable model semantics of weight constraint rules. In *Proceedings of the 5th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'99)*, M. Gelfond, N. Leone, and G. Pfeifer, Eds. Lecture Notes in Computer Science, vol. 1730. Springer Verlag, El Paso, TX, USA, 317–331.

PAPADIMITRIOU, C. H. 1994. *Computational Complexity*. Addison-Wesley.

PFANDLER, A., RÜMMELE, S., AND SZEIDER, S. 2013. Backdoors to abduction. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, F. Rossi, Ed. The AAAI Press, Beijing, China, 1046–1052.

PICHLER, R., RÜMMELE, S., SZEIDER, S., AND WOLTRAN, S. 2014. Tractable answer-set programming with weight constraints: bounded treewidth is not enough. *Theory Pract. Log. Program. 14,* 2, 141–164.

PIPATSRISAWAT, K. AND DARWICHE, A. 2011. On the power of clause-learning SAT solvers as resolution engines. *Artificial Intelligence 175,* 2, 512–525.

PONTELLI, E., SON, T., BARAL, C., AND GELFOND, G. 2012. Answer set programming and planning with knowledge and world-altering actions in multiple agent domains. In *Correct Reasoning – Essays on Logic-Based AI in Honour of Vladimir Lifschitz*, E. Erdem, J. Lee, Y. Lierler, and D. Pearce, Eds. Lecture Notes in Computer Science, vol. 7265. Springer Verlag, 509–526.

RICCA, F., GRASSO, G., ALVIANO, M., MANNA, M., LIO, V., IIRITANO, S., AND LEONE, N. 2012. Team-building with answer set programming in the Gioia-Tauro seaport. *Theory Pract. Log. Program. 12*, 361–381.

ROBERTSON, N. AND SEYMOUR, P. 1984. Graph Minors. III. Planar Tree-Width. *J. Combin. Theory Ser. B 36,* 1, 49–64.

SAMER, M. AND SZEIDER, S. 2010. Constraint satisfaction with bounded treewidth revisited. *J. Comput. Syst. Sci. 76,* 2, 103–114.

SYRJÄNEN, T. 2009. Logic programs and cardinality constraints: Theory and practice. Ph.D. thesis, Helsinki University of Technology.

TRUSZCZYŃSKI, M. 2011. Trichotomy and dichotomy results on the complexity of reasoning with disjunctive logic programs. *Theory Pract. Log. Program. 11*, 881–904.

VARDI, M. Y. 2014. Boolean satisfiability: Theory and engineering. *Communications of the ACM 57,* 3 (Mar.), 5–5.

WILLIAMS, R., GOMES, C., AND SELMAN, B. 2003a. Backdoors to typical case complexity. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, G. Gottlob and T. Walsh, Eds. Morgan Kaufmann, Acapulco, Mexico, 1173–1178.

WILLIAMS, R., GOMES, C., AND SELMAN, B. 2003b. On the connections between backdoors, restarts, and heavy-tailedness in combinatorial search. In *Informal Proceedings of the 6th International Conference on Theory and Applications of Satisfiability Testing (SAT'03)*. Portofino, Italy, 222–230.