A Multiparametric View on Answer Set Programming*

Johannes K. Fichte[1], Martin Kronegger[2], and Stefan Woltran[1]
lastname@dbai.tuwien.ac.at

[1] TU Wien, Vienna, Austria
lastname@dbai.tuwien.ac.at
[2] Johannes Kepler University Linz, Austria
martin.kronegger@jku.at

Abstract. Disjunctive answer set programming (ASP) is an important framework for declarative modeling and problem solving, where the computational complexity of basic decision problems like consistency (deciding whether a program has an answer set) is located on the second level of the polynomial hierarchy. During the last decades different approaches have been applied to find tractable fragments of programs, in particular, also using parameterized complexity. However, the full potential of parameterized complexity has not been unlocked since only one or very few parameters have been considered at once. In this paper, we consider several natural parameters for the consistency problem of disjunctive ASP. In addition, we also take the size of the answer set into account; a restriction that is particularly interesting for applications requiring small solutions. Previous work on parameterizing the consistency problem by the size of answer sets yielded mostly negative results. In contrast, we start from recent findings for the problem WMMSAT and show several novel fixed-parameter tractability (fpt) results based on combinations of parameters. Moreover, we establish a variety of hardness results (paraNP, W[2], and W[1]-hardness) to assess tightness of our combined parameters.

1 Introduction

Answer set programming (ASP) is an important framework for declarative modelling and problem solving [15]. 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. Computational problems for disjunctive, propositional ASP such as the consistency problem (deciding whether a program has a solution) are complete for the second level of the Polynomial Hierarchy [4]. However, classical worst case complexity does not rule out efficient solutions if a certain (hidden)

* The work has been supported by the Austrian Science Fund (FWF), Grant Y698 and S11408-N23. The first author is also affiliated with the Institute of Computer Science and Computational Science at University of Potsdam, Germany. The second author is also affiliated with TU Wien, Austria.

structure is present in an input instance. On that score, several restrictions on input programs have been identified in the literature that make the consistency problem tractable or NP-complete, for a detailed trichotomy see [16]. A prominent approach to analyze and understand computational complexity incorporating the existence of certain hidden structure is to use the framework of parameterized complexity [3]. The main idea of parameterized complexity is to fix a certain structural property (the parameter) of a problem instance and to consider the computational complexity of the problem in dependency of the parameter. Various parameterized complexity analyzes have been carried out for ASP problems, see e.g., [6,7,14,8]. In particular, Lonc and Truszczyński [14] have considered the parameterized complexity of the consistency problem parameterized by a given integer k, when the input is restricted to normal (i.e., disjunction-free) programs and when then answer sets are allowed to be of size exactly k, or at most k, or at least k, and established various hardness results.

In AI more fine-grained complexity analysis, where hidden structure may consist of a combination of various structural properties, have also been established for problems such as weighted minimal model satisfiability (WMMSat) [13] and planning [12]. So far, there has been no rigorous study of disjunctive ASP when considering various combinations of structural properties.

Contribution. In this paper, we study the computational complexity of propositional disjunctive ASP using the framework of parameterized complexity theory [1,3]. We 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 [13] 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. Furthermore, we incorporate results by Lonc and Truszczyński [14] and Truszczyński [16]. 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.

2 Preliminaries

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 atoms 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). In addition to the traditional representation of a rule above, we sometimes also write $H(r) \leftarrow B^+(r), \neg B^-(r)$, and $H(r) \leftarrow B^+(r)$ instead of $H(r) \leftarrow B^+(r), \neg\emptyset$. We denote the sets of atoms occurring in a rule r or in a program P by $\text{at}(r) = H(r) \cup B^+(r) \cup B^-(r)$ and $\text{at}(P) = \bigcup_{r\in P} \text{at}(r)$, respectively. We write $\text{occ}_P(a) := \{\, r \in P : a \in \text{at}(r) \,\}$. 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 *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 *Horn* if it is negation-free and normal or a constraint, r is *definite Horn* if it is Horn and not a constraint, 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**. **NF+Cons** refers to the class of all programs where negation-free rules and arbitrary constraint rules (may also contain negative atoms) are allowed. 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 $P_c = P \setminus P_r$. We occasionally write $\bot$ as a head 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 [11]. M is an *answer set* (or *stable model*) of a program P if M is a minimal model of P^M. 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. 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 $LM(P)$ [10]. Dowling and Gallier [2] have established a linear-time algorithm for testing the satisfiability of propositional Horn formulas which easily extends to Horn programs.

In this paper, we consider the following ASP problems. k-Consistency: Given a program P and an integer k decide whether P has an answer set of size at most k. k-Brave Reasoning: Given a program P, an atom $a \in \text{at}(P)$, and an integer k decide whether P has an answer set M of size at most k such that $a \in M$. We denote by *k-AspProblems* the family of the reasoning problems k-Consistency and k-Brave Reasoning. Further, we use the problem k-Enum:

Given a program P and an integer k list all answer sets of size at most k of P. We refer to the problems as Consistency, Brave Reasoning, and Enum, respectively, if the integer k can be arbitrarily large. We denote by $\mathcal{AspProblems}$ the family of the reasoning problems Consistency and Brave Reasoning.

We also need some notions from *propositional satisfiability*. A *clause* is a finite set of literals, a CNF formula is a finite set of clauses. The set of variables of a CNF formula F is denoted by $var(F)$. A *truth assignment* is a mapping $\tau : X \rightarrow \{0, 1\}$ defined for a set $X \subseteq U$ of atoms. By 2^X we denote the set of all truth assignments $\tau : X \rightarrow \{0, 1\}$. For $x \in X$ we define $\tau(\neg x) = 1 - \tau(x)$. 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 second removing from the remaining clauses all literals set to 0 by τ. τ *satisfies* F if $F_\tau = \emptyset$, and F is *satisfiable* if it is satisfied by some τ. Note that if a formula F contains some clause C of only positive literals and we have a truth assignment τ that sets all literals occurring in C to 0, then we obtain $\{\emptyset\} \subseteq F_\tau$, which is obviously not satisfiable. The problem Weighted Minimal Model Satisfiability (WMMSat) asks to decide given two propositional (CNF) formulas φ and π and an integer k whether there is a minimal model M of φ that sets at most k variables to true and also satisfies π. The problem $\text{WSat}_\leq$ is defined as follows. Weighted Satisfiability ($\text{WSat}_\leq$): Given A CNF formula F and some integer k decide whether F has a model $M \subseteq var(F)$ of cardinality $|M| \leq k$.

Parameterized Complexity. We give some basic background on parameterized complexity. For more detailed information we refer to other sources [1,3]. A *parameterized problem* L is a subset of $\Sigma^* \times \mathbb{N}$ for some finite alphabet Σ. For

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
$\text{maxsize}^r_{H,B^+}$	maximum size of the head and positive 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 negative body of a constraint
$\#\text{non-Horn}^r$	number of non-(definite Horn) rules
maxocc^r_{H,B^-}	maximum number of occurrences of a variable in P_r when only the head and negative-body occurrences are counted
$\text{maxocc}^r_{B^+}$	maximum number of occurrences of a variable in P_r when only the positive-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
$\|\|P_c\|\|$	the total number of variable occurrences in P_c

Table 1: List and informal description of the considered parameters.

		#at				maxsize					maxocc			
k	$\|\|P_c\|\|$	$\|\cdot\|^r_H$	$\|\cdot\|_{B+}$	$\|\cdot\|_{B-}$	nH	$\|\cdot\|^r_H$	$\|\cdot\|^r_{B+}$	$\|\cdot\|^r_{B-}$	$\|\cdot\|^c_{B+}$	$\|\cdot\|^c_{B-}$	$\|\cdot\|^r_H$	$\|\cdot\|^r_{B+}$	$\|\cdot\|^r_{B-}$	Result
		✗												∈ FPT
✗						✗		✗						∈ FPT
					✗	✗		✗						∈ FPT
✗			✗	✗						✗	✗		✗	∈ FPT
✗			✗							✗	✗		✗	open
			✗	✗	✗					✗				∈ FPT
			✗		✗					✗				open
✗	✗				✗		✗			✗	✗		✗	W[1]-h
✗						✗	✗		✗	✗		✗		W[2]-h
✗								✗	✗	✗				W[2]-h
✗	✗		✗				✗			✗				W[2]-h
✗			✗		✗		✗				✗		✗	W[2]-h
	✗		✗			✗	✗	✗		✗				paraNP-h
				✗		✗	✗	✗						paraNP-h
k	$\|\|P_c\|\|$	$\|\cdot\|^r_H$	$\|\cdot\|^r_{B+}$	$\|\cdot\|^r_{B-}$	nH	$\|\cdot\|^r_H$	$\|\cdot\|^r_{B+}$	$\|\cdot\|^r_{B-}$	$\|\cdot\|^c_{B+}$	$\|\cdot\|^c_{B-}$	$\|\cdot\|^r_H$	$\|\cdot\|^r_{B+}$	$\|\cdot\|^r_{B-}$	Result

Table 2: Summary of multi-parametric complexity results for k-Consistency. For each line the marked columns indicate according to the header a combined ASP parameter. Membership or hardness results are stated in the last column.

an instance $(I, k) \in \Sigma^* \times \mathbb{N}$ we call I the *main part* and k the *parameter*. L is *fixed-parameter tractable* if there exist a computable function f and a constant c such that we can decide whether $(I, k) \in L$ in time $O(f(k)\|I\|^c)$ where $\|I\|$ denotes the size of I. FPT is the class of all fixed-parameter tractable decision problems. The *Weft Hierarchy* consists of parameterized complexity classes $\mathrm{W}[1] \subseteq \mathrm{W}[2] \subseteq \cdots$ which are defined as the closure of certain parameterized problems under parameterized reductions. There is strong theoretical evidence that parameterized problems that are hard for classes $\mathrm{W}[i]$ are not fixed-parameter tractable. It is well-known that different variations of $\mathrm{WSat}_{\leq}$ can be used to define the W-hierarchy (see, e.g., the work of Flum and Grohe [9]).

Let $L \subseteq \Sigma^* \times \mathbb{N}$ and $L' \subseteq \Sigma'^* \times \mathbb{N}$ be two parameterized 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 $O(f(k)\|I\|^c)$ where $\|I\|$ denotes the size of I [9]. Thus, an fpt-reduction is, in particular, an fpt-algorithm. It is easy to see that the class FPT is closed under fpt-reductions. We would like to note that the theory of fixed-parameter intractability is based on fpt-reductions [9].

The parameterized complexity of the problems $\mathrm{WSat}_{\leq}$ and WMMSat has been studied in the work of Lackner and Pfandler [13]. Several hardness and tractability results for combined parameter turn out to be useful to show hardness and tractability results for the considered ASP problems.

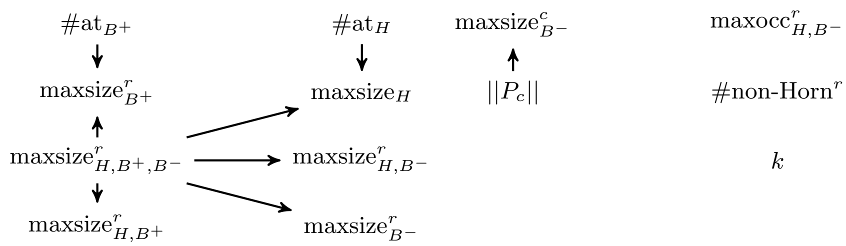

Fig. 1: Domination Graph (relationship between parameters). Let x and y be parameters. There is an arc $x \rightarrow y$ whenever $x \preceq y$ holds.

2.1 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. First, we first give a definition what we mean by an ASP parameter.

Definition 1 *An* ASP parameter *is a function* p *that assigns to every program* P *some non-negative integer* $p(P)$*. Let* p *and* q *be ASP parameters. We say that* p dominates q *(in symbols* $p \preceq q$*) if there is a computable function* f *such that* $p(P) \leq f(q(P))$ *holds for all programs* P*.*

Table 1 lists the considered parameters, which can be computed in polynomial time, and their intuitive description. For a more formal description, let P be a program and $X \subseteq \{H, B^+, B^-\}$ where H, B^+, and B^- are mappings defined as in Section 2. We omit P if the program is clear from the context. Further, let $\mathrm{at}_{X,r} := \cup_{f\in X} f(r)$, $\#\mathrm{at}_X := |\cup_{r\in P}\ \mathrm{at}_{X,r}|$, $\mathrm{maxsize}^r_X := \max\big\{ \sum_{f\in X, r'\in P} |f(r')| : |H(r')| > 0 \big\}$, $\mathrm{maxsize}^c_X := \max\big\{ \sum_{f\in X, r'\in P} |f(r')| : |H(r')| = 0 \big\}$, $\#\text{non-Horn}^r := |\{r' : r' \in P, r' \text{ not Horn}\}|$, and $\mathrm{maxocc}^r_X := \max\big\{ i : a \in at(P), i = \sum_{f\in X,\, r'\in P,\, |H(r')|>0} |\{ a : a \in f(r') \}| \big\}$. Figure 1 depicts the relationship in terms of domination of parameters that are useful for our results. Note, that this list is not complete.

2.2 Relationship of *AspProblems* and *k-AspProblems*

Recent research in parameterized complexity in the setting of answer set programming, has mainly focused on consistency or reasoning problems, which allow arbitrarily large answer sets. However, we focus on ASP problems that also take the size of the answer set into account. In the following, we explain and summarize connections between both versions. We observe that if the parameters do not depend on the maximum size of an answer set state conditions, we can trivially extend known membership and hardness results for problems in *AspProblems* to the respective problem in *k-AspProblems*. In other words, the problem k-Consistency is at least as hard as Consistency. Finally, we state how to extend known results for Consistency using standard counters that do not effect the other considered parameters.

Observation 2 *Let p be an ASP parameter, $\mathcal{C}$ be a parameterized complexity class, and $L \in \{$*Consistency, Brave Reasoning*$\}$, and k-L its corresponding decision problem k-*Consistency *or k-*Brave Reasoning*, respectively, in other words, k-L decides the question of L when restricted to answer sets of size at most k.*

1. *If the problem k-$L \in \mathcal{C}$ when parameterized by p and p does not depend on k, then the problem $L \in \mathcal{C}$ under fpt-reductions when parameterized by p.*
2. *Further, if problem L is $\mathcal{C}$-hard when parameterized by p and p does not depend on k, then the problem k-L is $\mathcal{C}$-hard under fpt-reductions when parameterized by p.*

Note that the restriction "p does not depend on k" is quite weak as in that case both problems coincide.

Next, we will see that if a decision problem in *AspProblems* is fixed-parameter tractable when parameterized by some fixed parameter p and p is not affected by restricting the solution size to at most k, then the corresponding problem for answer sets of size at most k is fixed-parameter tractable when parameterized by the combined parameter $p + k$ where k is the size of the answer set.

Definition 3 *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 4 ($\star$[3]) *Let p be a counter-preserving ASP parameter, $\mathcal{C}$ be a parameterized complexity class, $L \in \{$*Consistency, Brave Reasoning*$\}$, and k-L its corresponding decision problem k-*Consistency *or k-*Brave Reasoning*, respectively. If the problem L belongs to class $\mathcal{C}$ when parameterized by p, then the problem k-L belongs to class $\mathcal{C}$ under fpt-reductions when parameterized by p.*

3 Membership Results

In this section, we present for ASP reasoning problems several novel fixed-parameter tractability results, which are summarized in Table 2. We first observe that parameterizing in the number of head atoms already yields fixed-parameter tractability.

Observation 5 ($\star$) *For each problem $L \in \{k$-*Consistency*, k-*Brave Reasoning*$\}$, we have L is fixed-parameter tractable when parameterized by at least one of the following parameters (i) $\#\mathrm{at}_H$ or (ii) $\mathrm{maxsize}_H + |P_r|$.*

[3] Statements whose proofs are omitted due to space limitations are marked with "($\star$)".

We now proceed to two tractability results that can be obtained by a reduction to WMMSat.

Theorem 6 (⋆) *Let $L \in \{k$-Consistency, k-Brave Reasoning$\}$. Then, L is fixed-parameter tractable when parameterized by at least one of the following combined parameters*

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

Proof (Idea). The main idea of the proof is a reduction to WMMSat. WMMSat is fixed-parameter tractable when parameterized by at least one of the following combined parameters (i) $k+$ maximum positive clause size, or (ii) maximum positive clause size + number of non-Horn clauses [13]. Our reduction runs in linear time and preserves all necessary parameters.

The reduction consists of two reductions: (i) from P we construct in linear time programs $P^{\text{mmod}} \cup P^{\text{subset}}$ and P^{supset}, and (ii) from $P^{\text{mmod}} \cup P^{\text{subset}}$ and P^{supset} we construct in linear time an instance of WMMSat. For a set X, we let $(X)' := \{a' \mid a \in X\}$. In Step (i) we set: $P^{\text{mmod}} := \big\{\, H(r), \big(B^-(r)\big)' \leftarrow B^+(r) : r \in P \,\big\}$, $P^{\text{subset}} := \big\{\, a' \leftarrow a : a \in \text{at}(P) \,\big\}$, $P^{\text{supset}} := \big\{\, a \leftarrow a' : a \in \text{at}(P) \,\big\}$, and $P^* := P' \cup P^{\text{supset}}$. Then in Step (ii) we construct a WMMSat instance by $\varphi := F_{\text{mmod}} \wedge F_{\text{subset}}$ and $\pi := F_{\text{supset}}$. Where every $* \in \{\text{mmod}, \text{subset}, \text{supset}\}$ we construct formula F_* from P^* as follows:

$$F_* := \bigwedge_{r \in P^*} \left(\bigvee_{a \in B^+(r)} \neg v[a] \vee \bigvee_{a \in H(r)} v[a] \right)$$

Then, the formula F_{mmod} encodes P^{mmod} and expresses that an atom a belongs to the minimal model if and only if the atom occurs either in the head or the negative body of a rule. F_{subset} encodes P^{subset} and expresses that whenever an atom a belongs to the minimal model then also its copy a' belongs to the minimal model. Finally, F_{supset} encodes P^{supset} and "simulates the GL reduct" and expresses that whenever the copy a' of atom a belongs to the model, also its originating atom a has to be in the model.

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, the 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 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}.

We obtain membership for k-Brave Reasoning by the same arguments. ∎

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

Theorem 7 **($\star$)** *Let $L \in \{k$-Consistency, k-Brave Reasoning$\}$. Then L is fixed-parameter tractable when parameterized by at least one of the following combined parameters*

1. $k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$, *and*
2. $\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-} + \#\text{at}_{B^-}$.

In order to prove the theorem, we first establish the statement for a restricted version, namely, when the input is restricted to programs from **NF+Cons**. In fact, the problem Consistency is already Σ^P_2-complete when the input is restricted to programs from **NF+Cons** [16]. By definition, such programs have an empty negative body for non-constraint rules and hence the parameter $\#\text{at}_{B^-}$ is of value 0.

Lemma 8 *Let $L \in \{k$-Consistency, k-Brave Reasoning$\}$. If the input of L is restricted to programs from* **NF+Cons**, *then L is fixed-parameter tractable when parameterized by at least one of the following combined parameters (i) $k + \#\text{at}_{B^+} + \text{maxocc}^r_{H,B^-} + \text{maxsize}^c_{B^-}$, and (ii) $\#\text{at}_{B^+} + \#\text{non-Horn}^r + \text{maxsize}^c_{B^-}$.*

To proof the lemma we need the following results.

Proposition 9 ([13]) WMMSat *is fixed-parameter tractable when parameterized by at least one of the following combined parameters (i) $k + v^- + p + d^+_\pi$, and (ii) $v^- + h + d^+_\pi$, where k is the maximum weight of the minimal model, h is the number of non-Horn clauses, p is the maximum number of positive occurrences of a variable in π, v^- is the number of variables that occur as negative literals in π or in π, and d^+_π maximum positive clause size in π.*

Proof of Lemma 8 (Idea). We give a reduction to WMMSat, which preserves all parameters considered in the statement. Let (P, k) be an instance of k-Consistency where $P \in$ **NF+Cons**. We 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 $\pi := \{\, C(r) : r \in P, H(r) = \emptyset\,\}$. 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 observe that our reduction preserves the parameters:

- k: directly corresponds to the maximum weight of a minimal model (k)
- $\text{maxsize}^r_{H,B^-}$: directly corresponds to the maximum positive clause size in φ (d^+)
- $\text{maxsize}^r_{B^+}$: directly corresponds to the maximum negative clause size in φ (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 or in π (v^-)
- $\text{maxsize}^c_{B^-}$: directly corresponds to the maximum positive clause size in π (d^+_π). ∎

Before we are able to prove the theorem, we extend concepts from earlier work [7] and define the concept of a truth assignment reduct under sets M and N of atoms.

Definition 10 *Let P be a program, $M \subseteq \text{at}(P)$, and $N \subseteq \text{at}(P) \setminus M$. 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 $a \in M$.*

Proof of Theorem 7 (Sketch). We give an fpt-reduct that constructs $2^{(\#\text{at}_{B^-}+1)}$ many programs that can be solved in fpt-time using results established in Lemma 8. Let (P, ℓ) be an instance of k-Consistency, $N := \cup_{r \in P, H(r) \neq \emptyset} \left(\text{at}_{B^-,r}\right)$, $\tau \in 2^N$, $M_1 := \tau^{-1}(1)$, 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. In this way, we give a reduction to $2^{(\#\text{at}_{B^-}+1)}$ many instances of k-Consistency that consists of $2^{(\#\text{at}_{B^-}+1)}$ 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. ∎

The reduction in the proof of Theorem 8 states that ASP and WMMSat are very related with respect to the considered reasoning problems. However, answer sets additionally require minimality with respect to the GL reduct of the given program. In consequence, we need to parameterize additionally in the number of negative atoms that occur in 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 [13]) if the positive body is empty and the negative body is not empty.

The next result states that a fixed-parameter tractability result for the Enum problem directly extends to a fixed-parameter tractability result with the same parameter for our considered ASP reasoning problems, where we are interested only in answer sets of size at most k.

Proposition 11 (⋆) *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*$\}$, L is fixed-parameter tractable when parameterized by p.*

Consequently, known results for backdoors [7] immediately apply to our problems in k-$\mathcal{AspProblems}$.

Corollary 12 *Let $\mathcal{C}$ be an enumerable class of normal programs. Every problem $L \in \{k$-Consistency, k-Brave Reasoning$\}$ is fixed-parameter tractable when parameterized by the size of a strong $\mathcal{C}$-backdoor.*

Remark 13 *We would like to mention that the parameter incidence or primal treewidth [5] immediately provides fixed-parameter tractability for our problems in k-$\mathcal{AspProblems}$, since the dynamic programming algorithms as presented in previous work can be trivially modified such that the computation stops at size at most k.*

4 Hardness Results

In this section, we present as already stated in Table 2 for ASP reasoning problems several hardness results. Observe that hardness for a combination of parameters trivially implies hardness for any subset of these parameters. Whereas the fpt results in the previous section imply fpt results for any superset of these parameters.

Theorem 14 (⋆) *Let $L \in \{k$-Consistency, k-Brave Reasoning$\}$. Then L is* paraNP*-hard when parameterized by the following parameters*

1. $\mathrm{maxsize}^r_{H,B^+,B^-} + \mathrm{maxsize}^r_{H,B^-} + \mathrm{maxsize}^r_{B^+} + \#\mathrm{at}_{B^+} + \mathrm{maxsize}^c_{B^-} + ||P_c||$, *and*
2. $\mathrm{maxsize}^r_{H,B^+,B^-} + \mathrm{maxsize}^r_{H,B^-} + \mathrm{maxsize}^r_{B^+} + \#\mathrm{at}_{B^-}$.

Let $L \in \{k$-Consistency, k-Brave Reasoning$\}$. Then L is W[2]*-hard when parameterized by the following parameters*

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

Let $L \in \{k$-Consistency, k-Brave Reasoning$\}$. Then L is W[1]*-hard when parameterized by the following parameters*

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

Proof (Sketch). We proceed by a reduction from the problem WMMSat for Statements (1) and (5)–(7) and WSat for Statement (4). Statement (2) has already been established by Truszczyński [16]. Statement (3) is an immediate consequence from a reduction established by Lonc and Truszczyński [14][The. 4.4]. Note that according to results by Lackner and Pfandler [13] WMMSat is paraNP-hard when parameterized by the following combined parameter: (i) $d + d^+ + d^- + p + v^- + d^+_\pi + ||\pi||$; WMMSat is W[2]-hard when parameterized by the following combined parameters (ii) $k + d^- + v^- + d^+_\pi + ||\pi||$ and (iii) $k + d^- + h + p + v^-$; WMMSat is W[1]-hard when parameterized by the following combined parameters (iv) $k + d^- + h + p + d^+_\pi + ||\pi||$ where k is the maximum weight of the minimal model, d is the maximum clause size, d^+, d^- is the maximum positive or negative clause size, respectively in φ, h is the number of non-horn clauses in φ, b is the minimum size of strong Horn backdoor set in φ, p is the maximum number of positive occurrences of a variable in φ, v^+, v^- is the number of variables that occur as positive or negative literals in φ or in π, respectively, d^+_π maximum positive clause size in π, $||\pi||$ is the length of π, i.e., the total number of variable occurrences in π.

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 affecting the size of the models and hence the minimality.[4] 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 $P^{\min} := \{\, C^1 \leftarrow C^0 : C \in \varphi \,\}$ and $P^{\text{cons}} := \{\, \leftarrow \neg C^1, C^0 : C \in \pi \,\}$ and we define a program $P := P^{\min} \cup P^{\text{cons}}$. 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.

Next, we can employ the construction and proofs from above to establish a reduction from an instance (φ, k) of WSat for Statement 4. Note that WSat is well known to be W[2]-hard, e.g., [3]. Therefore, observe that φ has a model M of size at most k if and only if $P^{\min}$ 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 in φ (d)
- $\text{maxsize}^r_{H,B^-}$: directly corresponds to the maximum positive clause size in φ (d^+)
- $\text{maxsize}^r_{B^+}$: directly corresponds to the maximum negative clause size in φ (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)

[4] Note that this has also no effect to the results we use for WMMSat, since the parameters used in the proofs for WMMSat remain unaffected (it only effects d and d^-, however, there d^- is already bounded by v^-; see the proofs of Theorems 16 and 17 in [13]).

- $\#\mathrm{at}_{B^+}$: directly corresponds to the number of variables that occur as negative literals in φ or in π (v^-)
- $\mathrm{maxsize}^c_{B^-}$: directly corresponds to the maximum positive clause size in π (d^+_π)
- $||P_c||$: directly corresponds to the length of π, i.e., the total number of variable occurrences in π ($\|\pi\|$)

The runtime follows from the results by Lackner and Pfandler [13] for WMMSat as stated above. ∎

Remark 15 *Note that our reductions make certain concepts of parameters in the setting of answer set programming such as fpt-results for acyclicity-based backdoors [6] or bounded treewidth [5] directly accessible to* WMMSat*.*

5 Conclusion

We have identified several natural structural parameters of ASP instances (as summarized in Table 1) 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 (see Table 2 for an overview). 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 and membership results (see Table 1). 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.

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. Additionally, it would be interesting to conduct a parameterized analysis as well as considering multiple parameters in the non-ground setting.

References

1. Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Dániel Marx, M.P., Pilipczuk, M., Saurabh, S.: Parameterized Algorithms. Springer Verlag (2015)

2. Dowling, W.F., Gallier, J.H.: Linear-time algorithms for testing the satisfiability of propositional Horn formulae. J. Logic Programming 1(3), 267–284 (1984)
3. Downey, R.G., Fellows, M.R.: Fundamentals of Parameterized Complexity. Texts in Computer Science, Springer Verlag, London, UK (2013)
4. Eiter, T., Gottlob, G.: On the computational cost of disjunctive logic programming: Propositional case. Ann. Math. Artif. Intell. 15(3–4), 289–323 (1995)
5. Fichte, J.K., Hecher, M., Morak, M., Woltran, S.: Answer set solving with bounded treewidth revisited. In: Balduccini, M., Janhunen, T. (eds.) Proceedings of the 14th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'17) (2017), to appear. Extended version https://arxiv.org/pdf/1702.02890.pdf.
6. Fichte, J.K., Szeider, S.: Backdoors to normality for disjunctive logic programs. ACM Trans. Comput. Log. 17(1), 7 (Dec 2015)
7. Fichte, J.K., Szeider, S.: Backdoors to tractable answer-set programming. Artificial Intelligence 220(0), 64–103 (2015)
8. Fichte, J.K., Szeider, S.: Backdoor trees for answer set programming. In: Bogaerts, B., Harrison, A. (eds.) Informal Proceedings of the 10th Workshop on Answer Set Programming and Other Computing Paradigms (ASPOCP'17) (2017), to appear.
9. Flum, J., Grohe, M.: Parameterized Complexity Theory, Theoretical Computer Science, vol. XIV. Springer Verlag, Berlin (2006)
10. Gelfond, M., Lifschitz, V.: The stable model semantics for logic programming. In: Kowalski, R.A., Bowen, K.A. (eds.) Proceedings of the 5th International Conference and Symposium on Logic Programming (ICLP/SLP'88). vol. 2, pp. 1070–1080. MIT Press, Seattle, WA, USA (August 1988)
11. Gelfond, M., Lifschitz, V.: Classical negation in logic programs and disjunctive databases. New Generation Comput. 9(3/4), 365–386 (1991)
12. Kronegger, M., Pfandler, A., Pichler, R.: Parameterized complexity of optimal planning: A detailed map. In: Rossi, F. (ed.) Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13). pp. 954–961. The AAAI Press, Beijing, China (Aug 2013)
13. Lackner, M., Pfandler, A.: Fixed-parameter algorithms for finding minimal models. In: Eiter, T., McIlraith, S. (eds.) Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR'12). pp. 85–95. The AAAI Press, Rome, Italy (Jun 2012)
14. Lonc, Z., Truszczyński, M.: Fixed-parameter complexity of semantics for logic programs. ACM Trans. Comput. Log. 4(1), 91–119 (Jan 2003), http://doi.acm.org/10.1145/601775.601779
15. Marek, V.W., Truszczyński, M.: Stable models and an alternative logic programming paradigm. In: Apt, K.R., Marek, V.W., Truszczyński, M., Warren, D.S. (eds.) The Logic Programming Paradigm: a 25-Year Perspective, pp. 375–398. Artificial Intelligence, Springer Verlag (1999)
16. Truszczyński, M.: Trichotomy and dichotomy results on the complexity of reasoning with disjunctive logic programs. Theory Pract. Log. Program. 11, 881–904 (11 2011)