On the Complexity of Inconsistency Measurement

Matthias Thimm[1] and Johannes P. Wallner[2]

[1]*Institute for Web Science and Technologies, Universität Koblenz-Landau, Germany,*
[2]*Institute of Logic and Computation, TU Wien, Austria*

Abstract

We survey a selection of inconsistency measures from the literature and investigate their computational complexity wrt. decision problems related to bounds on the inconsistency value and the functional problem of determining the actual value. Our findings show that those inconsistency measures can be partitioned into four classes related to their complexity. The first three classes contain measures whose complexities are located on the first three levels of the polynomial hierarchy, respectively. The final class is under standard complexity-theoretic assumptions located beyond the polynomial hierarchy. We provide membership results for all the investigated problems and completeness results for most of them. In addition, we undertake a preliminary study on the computational complexity of the measures on fragments of propositional logic.

Keywords: inconsistency measures, computational complexity

---

1. Introduction

The field of *inconsistency measurement* is about devising measures for quantitatively assessing the severity of inconsistencies in knowledge bases. Consider the following two knowledge bases $\mathcal{K}_1$ and $\mathcal{K}_2$ formalised in propositional logic:

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\} \qquad \mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Both knowledge bases are classically inconsistent, i.e., there is no classical interpretation satisfying any of the two knowledge bases. In particular, in $\mathcal{K}_1$ the formula a suggests that a has to hold while $\neg a \wedge \neg b$ suggests that $\neg a$ has to hold, while in $\mathcal{K}_2$ the contradiction is even more apparent by considering either the subset $\{a, \neg a\}$ or $\{b, \neg b\}$. These inconsistencies render the

knowledge bases useless for reasoning if one wants to use classical reasoning techniques. In order to make the knowledge bases useful again, one can either rely on non-monotonic/paraconsistent reasoning techniques (Makinson, 2005; Priest, 1979) or one revises the knowledge bases appropriately to make them consistent (Hansson, 2001). Looking at the knowledge bases $\mathcal{K}_1$ and $\mathcal{K}_2$ one can observe that the *severity* of their inconsistency is different. In $\mathcal{K}_1$, only two out of four formulas (a and $\neg a \wedge \neg b$) are "participating" in making $\mathcal{K}_1$ inconsistent while for $\mathcal{K}_2$ all formulas contribute to its inconsistency. Furthermore, for $\mathcal{K}_1$ only two propositions (a and b) are conflicting and using e. g. paraconsistent reasoning one could still infer meaningful statements about c and d. For $\mathcal{K}_2$ no such statement can be made. This leads to the assessment that $\mathcal{K}_2$ should be regarded *more* inconsistent than $\mathcal{K}_1$.

Inconsistency measures, firstly mentioned in (Grant, 1978), can be used to analyse inconsistencies and to provide insights on how to repair them. An inconsistency measure $\mathcal{I}$ is a function on knowledge bases, such that the larger the value $\mathcal{I}(\mathcal{K})$ the more severe the inconsistency in $\mathcal{K}$. A lot of different approaches of inconsistency measures have been proposed, mostly for classical propositional logic (Hunter and Konieczny, 2004, 2008, 2010; Ma et al., 2010; Mu et al., 2011; Xiao and Ma, 2012; Grant and Hunter, 2011; Ma et al., 2012; Grant and Hunter, 2013; McAreavey et al., 2014; Jabbour et al., 2015, 2014b; Raddaoui, 2015; Besnard, 2016; Thimm, 2016b; Jabbour and Sais, 2016; Jabbour, 2016; Ammoura et al., 2015, 2017), but also for classical first-order logic (Grant and Hunter, 2008), description logics (Ma et al., 2007; Zhou et al., 2009), default logics (Doder et al., 2010), answer set programming (Ulbricht et al., 2016) probabilistic and other weighted logics (Thimm, 2013; Potyka, 2014; De Bona and Finger, 2015), and relational databases (Decker, 2011), see also (Thimm, 2017b, 2018) for some recent surveys.

As the magnitude and diversity of these different measures show, the problem of conceptualizing a quantitative notion of inconsistency is not a trivial one. A common approach to categorize inconsistency measures is by differentiating whether they operate on the *formula level* or on the *language level*. The former category is also called the *syntactic approach* while the latter is called the *semantic approach* (Hunter and Konieczny, 2004). Measures belonging to the syntactic approach usually make use of *minimal inconsistent subsets*, i. e., subsets of the knowledge base that are inconsistent but removing any formula renders them consistent. A simple approach for measuring inconsistency is then to simply take the *number* of minimal inconsistent subsets

of a knowledge base as the value of inconsistency, cf. (Hunter and Konieczny, 2008). More recent measures also take the relationships between minimal inconsistent subsets into account (Jabbour and Sais, 2016). Other measures belonging to the syntactic approach may exploit other notions such as maximal consistent subsets (Ammoura et al., 2015, 2017) but the commonality of these approaches is that they focus on conflicts between formulas of the knowledge base. On the other hand, measures belonging to the semantic approach focus on conflicts between language components. More precisely, these measures aim at identifying those atoms of the underlying language that are conflicting and they usually employ non-classical and many-valued logics as a tool for that (Thimm, 2017a).

In this paper, we address the computational complexity of inconsistency measurement by investigating a selection of 17 inconsistency measures for propositional logic from the literature mentioned above. Inconsistency measurement is, by definition, a computationally intractable problem as it goes beyond merely detecting inconsistency (which is itself a coNP-complete problem for propositional logic). However, no systematic investigation of the complexity of inconsistency measures—and a comparison of measures wrt. it—has been conducted so far. The only complexity analyses on inconsistency measures we are aware of were presented in (Ma et al., 2010) and (Xiao and Ma, 2012) and each focused on a particular inconsistency measure. In (Ma et al., 2010) the complexity of a variant of the contension inconsistency measure $\mathcal{I}_c$ (Grant and Hunter, 2011) and in (Xiao and Ma, 2012) the complexity of the measure $\mathcal{I}_{mv}$ from (Xiao and Ma, 2012) themselves is investigated (we will recall the formal definitions of these measures in Section 3 and the corresponding results in Section 4, respectively). Recently, the algorithmic challenges in computing inconsistency measures have gained some attention (Ma et al., 2010; McAreavey et al., 2014; Jabbour et al., 2014a, 2016; Thimm, 2016b) and therefore calls for a theoretical investigation on the complexity of the involved computational problems. This paper continues our previous work (Thimm and Wallner, 2016) and provides a detailed analysis on the computational complexity of 17 measures wrt. three decision problems, namely deciding whether a given value is an upper, resp. lower bound, or is the exact value, as well as the functional problem of determining the inconsistency value. We mainly focus on the decision problems of deciding whether a given value is an upper, or resp. a lower bound, since, as we will see, the complexity classification of these decision problems gives crucial insights into the computational complexity of the inconsistency measure at hand. In ad-

3

dition, we also consider the Horn and Krom fragments of propositional logic and investigate whether measuring inconsistency becomes tractable in these fragments.

The main contributions of this paper are as follows.

- We show that the complexity of the decision problems of the inconsistency measures $\mathcal{I}_d$, $\mathcal{I}_\eta$, $\mathcal{I}_c$, $\mathcal{I}_{hs}$, $\mathcal{I}_{\mathrm{dalal}}^\Sigma$, $\mathcal{I}_{\mathrm{dalal}}^{\max}$, $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}$, $\mathcal{I}_{mcsc}$, and $\mathcal{I}_{\mathrm{forget}}$ is located on the first level of the polynomial hierarchy. These results imply that one can compute the exact value with logarithmically many calls to an NP-oracle. This in particular suggests the applicability of maximum satisfiability solvers (Ansótegui et al., 2013; Morgado et al., 2013) and similar systems for computing these measures.

- We establish completeness for a class in the second level of the polynomial hierarchy for measures $\mathcal{I}_p$, $\mathcal{I}_{mv}$, and $\mathcal{I}_{nc}$. Thus, these measures can be computed with logarithmically many calls to a Σ_2^p oracle. Systems capable of dealing with such high complexity are, e. g., answer-set programming solvers (Brewka et al., 2011).

- We show membership of the measure $\mathcal{I}_{CC}$ for the third level of the polynomial hierarchy. Although a completeness proof is still missing, this results suggests high intractability for $\mathcal{I}_{CC}$.

- We prove that the problems for measures $\mathcal{I}_{\mathsf{MI}}$, $\mathcal{I}_{\mathsf{MI}^c}$, $\mathcal{I}_{mc}$, and $\mathcal{I}_{\mathrm{is}}$ are hard for complexity classes related to counting. Under complexity theoretic assumptions, our results imply that (i) problems under these measures are computationally more challenging than propositional model counting, a problem itself seen as highly intractable and important (Gomes et al., 2009), and (ii) that decision problems associated with these measures presumably are not contained in a class of the polynomial hierarchy. Algorithms for computing several of these problems can be built upon systems for enumerating minimal unsatisfiable sets such as (Marques-Silva, 2012; McAreavey et al., 2014; Liffiton et al., 2016).

- We show that when considering the Horn or Krom fragment of propositional logic, measuring inconsistency wrt. the measures $\mathcal{I}_d$, $\mathcal{I}_{hs}$, and $\mathcal{I}_{\mathrm{dalal}}^{\max}$ becomes tractable. On the other hand, measuring inconsistency wrt. measures $\mathcal{I}_{\mathrm{dalal}}^\Sigma$, $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}$, and $\mathcal{I}_{\mathrm{forget}}$ remains intractable.

Before we give the details of our technical contributions in Section 4, we first provide some necessary preliminaries in Section 2 and introduce the inconsistency measures used in our analysis in Section 3. We conclude with a review of related work in Section 6 and a discussion in Section 7.

This paper extends the conference paper (Thimm and Wallner, 2016) by including all proofs of technical results, fixing some minor errors, providing many completeness results missing before, the analysis of four new measures $\mathcal{I}_{mcsc}$, $\mathcal{I}_{\text{forget}}$, $\mathcal{I}_{CC}$, and $\mathcal{I}_{\text{is}}$, and the preliminary investigation of tractable fragments. For details compare Table 1 in (Thimm and Wallner, 2016) with Table 2 of this paper. Proofs of technical results that are not in the main body of the paper can be found in Appendix A.

2. Preliminaries

In the following, we introduce some necessary preliminaries on propositional logic and computational complexity.

2.1. Propositional Logic

Let At be some fixed propositional signature, i. e., a (possibly infinite) set of propositions, and let $\mathcal{L}(\mathsf{At})$ be the corresponding propositional language constructed using the usual connectives $\wedge$ (*conjunction*), $\vee$ (*disjunction*), $\rightarrow$ (*implication*), and $\neg$ (*negation*). A literal is a proposition p or a negated proposition $\neg p$.

Definition 1. A knowledge base $\mathcal{K}$ is a finite set of formulas $\mathcal{K} \subseteq \mathcal{L}(\mathsf{At})$. Let $\mathbb{K}$ be the set of all knowledge bases.

A clause is a disjunction of literals. A formula is in conjunctive normal form (CNF) if the formula is a conjunction of clauses. If X is a formula or a set of formulas we write $\mathsf{At}(X)$ to denote the set of propositions appearing in X. The complement of a formula is denoted by overlining, i. e., $\overline{\phi} = \neg\phi$ if ϕ is not negated and $\overline{\neg\phi} = \phi$.

Semantics to a propositional language is given by *interpretations* and an *interpretation* ω on At is a function $\omega : \mathsf{At} \rightarrow \{\mathsf{true}, \mathsf{false}\}$. Let $\Omega(\mathsf{At})$ denote the set of all interpretations for At. An interpretation ω *satisfies* (or is a *model* of) an atom $a \in \mathsf{At}$, denoted by $\omega \models a$, if and only if $\omega(a) = \mathsf{true}$. The satisfaction relation $\models$ is extended to formulas in the usual way.

As an abbreviation we sometimes identify an interpretation ω with its *complete conjunction*, i. e., if $a_1, \ldots, a_n \in \mathsf{At}$ are those propositions that

5

are assigned true by ω and $a_{n+1}, \ldots, a_m \in \mathsf{At}$ are those propositions that are assigned false by ω we identify ω by $a_1 \ldots a_n \overline{a_{n+1}} \ldots \overline{a_m}$ (or any permutation of this). For example, the interpretation ω_1 on $\{a, b, c\}$ with $\omega(a) = \omega(c) = \mathsf{true}$ and $\omega(b) = \mathsf{false}$ is abbreviated by $a\bar{b}c$.

For $\Phi \subseteq \mathcal{L}(\mathsf{At})$ we also define $\omega \models \Phi$ if and only if $\omega \models \phi$ for every $\phi \in \Phi$. Define furthermore the set of models $\mathsf{Mod}(X) = \{\omega \in \Omega(\mathsf{At}) \mid \omega \models X\}$ for every formula or set of formulas X. If $\mathsf{Mod}(X) = \emptyset$ we also write $X \models \perp$ and say that X is *inconsistent*.

We also make use of the notation $\phi[\omega]$ for a formula ϕ and a (partial) interpretation ω, which denotes the uniform replacement of each proposition $x \in \mathsf{dom}\ \omega$ by $\top$ if $\omega(x) = \mathsf{true}$ and by $\perp$ if $\omega(x) = \mathsf{false}$.[1] If ω is partial, i. e. defined on a subset of variables in ϕ, this results in a formula with reduced vocabulary.

We recall some basic notions for inconsistency measurement that are used for many different measures. A set $M \subseteq \mathcal{K}$ is called a *minimal inconsistent subset* (MI) of $\mathcal{K}$ if $M \models \perp$ and there is no $M' \subset M$ with $M' \models \perp$. Let $\mathsf{MI}(\mathcal{K})$ be the set of all MIs of $\mathcal{K}$. Let $\mathsf{MC}(\mathcal{K})$ be the set of maximal consistent subsets of $\mathcal{K}$, i. e.

$$\mathsf{MC}(\mathcal{K}) = \{\mathcal{K}' \subseteq \mathcal{K} \mid \mathcal{K}' \not\models \perp \wedge \forall \mathcal{K}'' \supsetneq \mathcal{K}' : \mathcal{K}'' \models \perp\}$$

Furthermore, let $\mathsf{SC}(\mathcal{K})$ be the set of self-contradictory formulas of $\mathcal{K}$, i. e.

$$\mathsf{SC}(\mathcal{K}) = \{\phi \in \mathcal{K} \mid \phi \models \perp\}$$

2.2. Computational Complexity

We assume familiarity with the notions of *decision problems*, *functional problems*, *polynomial reductions*, *hardness*, *completeness*, and the complexity classes P, NP, and coNP, see also (Papadimitriou, 1994). In particular, recall that NP is the class of decision problems where the positive instances— i. e. those instances where the correct answer is TRUE—can be accepted in polynomial time by a non-deterministic Turing machine (or, equivalently, a non-deterministic algorithm running in polynomial time). The canonical NP-complete problem is the *satisfiability problem* SAT defined as

SAT **Input**:   a formula ϕ in CNF
             Output: TRUE iff $\mathsf{Mod}(\phi) \neq \emptyset$

---

[1] $\mathsf{dom}\ f$ denotes the domain of a function f.

The canonical coNP-complete problem is UNSAT defined as

UNSAT   **Input**:   a formula ϕ in CNF
                 Output: TRUE iff $\mathsf{Mod}(\phi) = \emptyset$

A decision problem can be identified with a language $\mathcal{L}$ that contains exactly the set of string representations of its positive instances.

For any two complexity classes $\mathcal{C}$ and $\mathcal{D}$, the class $\mathcal{C}^{\mathcal{D}}$ is the class of decision problems solvable in class $\mathcal{C}$ with access to an oracle for some problem complete in $\mathcal{D}$. For example, the class P^{NP} contains all problems that can be solved in polynomial time by a deterministic Turing machine that has access to an NP oracle (the oracle gives instantaneous answers to any query for a decision problem in NP and can therefore be queried at most polynomially many times). We make use of the polynomial hierarchy that can be defined as follows. Let $\Sigma_0^p = \Delta_0^p = \Pi_0^p = \mathsf{P}$ and define Σ_i^p, Δ_i^p, and Π_i^p for $i > 0$ recursively via

$$\Sigma_{i+1}^p = \mathsf{NP}^{\Sigma_i^p}$$
$$\Pi_{i+1}^p = \mathsf{coNP}^{\Sigma_i^p}$$
$$\Delta_{i+1}^p = \mathsf{P}^{\Sigma_i^p}$$

In particular, $\Sigma_1^p = \mathsf{NP}^{\Sigma_0^p} = \mathsf{NP}^{\mathsf{P}} = \mathsf{NP}$, $\Pi_1^p = \mathsf{coNP}$, and $\Delta_1^p = \mathsf{P}$.

A decision problem is in D_i^p iff it is the conjunction of a decision problem in Σ_i^p and a decision problem in Π_i^p. For one problem we also make use of the class coD_1^p, which is the complementary class of D_1^p. Furthermore, $\mathsf{P}^{\mathcal{C}[\log n]}$ contains all problems that can be solved in polynomial time with a deterministic Turing machine that may make logarithmically many calls to a $\mathcal{C}$ oracle (in contrast to $\mathsf{P}^{\mathcal{C}}$ which allows polynomial many calls, therefore $\mathsf{P}^{\mathcal{C}[\log n]} \subseteq \mathsf{P}^{\mathcal{C}}$). In general, $\mathsf{P}^{\mathcal{C}[f(n)]}$ contains all problems that can be solved in polynomial time with a deterministic Turing machine that may make $O(f(n))$ many calls to a $\mathcal{C}$ oracle. The class PSPACE contains all problems that can be solved in polynomial space.

The class FNP is the class of *functional problems* whose solutions can be determined in polynomial time by a non-deterministic Turing machine. We make use of the functional complexity class $\mathsf{FP}^{\mathcal{C}[f(n)]}$, i.e. the class containing *functional problems* whose solutions can be computed in polynomial time with a deterministic Turing machine that may make a $O(f(n))$ many calls to a $\mathcal{C}$ oracle. The class $\mathsf{FPSPACE}$ is the class of function problems whose

solutions can be computed in polynomial space. For functional complexity classes we utilize metric reductions to show hardness. A functional problem A reduces to a functional problem B if there exist polynomial-time computable functions f and g s.t. for input x for A it holds that $g(x, y)$ is a correct solution for problem A iff y is a correct solution for input $f(x)$ for problem B.

Some of the inconsistency measures inherently count certain structures, making counting complexity (Valiant, 1979b,a) a natural tool for our analysis. Intuitively, complexity classes, and complete problems, for counting complexity comprise of problems that ask whether there are at least (exactly) a given number of solutions to a problem, or, in analogy to function problems, what is the exact number of solutions.

For the classes containing decision problems regarding counting, we will make use of a class in the counting polynomial hierarchy defined by (Wagner, 1986). The classes in this hierarchy are defined in a similar fashion as the classes in the polynomial hierarchy, but we will introduce only what is needed for our purposes in this paper. Towards the definition, we define the counting quantifier C. This quantifier is defined for a predicate $H(x, y)$ with free variables x and y as

$$\mathsf{C}_y^k H(x, y) \leftrightarrow |\{y \mid H(x, y) \text{ is true }\}| \geq k.$$

That is, the counting quantifier is true for the predicate and bound k iff the number of values of y such that the predicate holds is at least k. The polynomially bounded version of the counting quantifiers is defined as follows. For a class of problems K a problem A is in $\mathsf{C}K$ iff there is a problem $B \in K$, a polynomial-time computable function f, and a polynomial p s.t.

$$x \in A \leftrightarrow \mathsf{C}_{|y| \leq p(|x|)}^{f(x)}(x, y) \in B.$$

In other words, instance x is in problem A iff there are at least $f(x)$ many y's, which are polynomially bounded by x, s.t. a predicate for (x, y) holds, with checking the predicate being in B.

The class CP is equal to the well-known class PP (see, e.g., (Papadimitriou, 1994)). A prototypical complete problem for CP is that of asking whether at least k many interpretations are models of a Boolean formula. In this paper we will use the class CNP, for which it holds that $\mathsf{CP} \subseteq \mathsf{CNP}$. From a computational point of view, counting is quite involving. One formal result that indicates this is that the whole polynomial hierarchy is contained in the

class P^{PP} (Toda, 1989), the class containing problems that can be solved via a deterministic polynomial time algorithm with access to a PP oracle.

For some problems relating to inconsistency measures it will be useful to consider also a variation of CNP, namely $\mathsf{C_{=}NP}$ (Wagner, 1986), which differs in the definition from CNP in that the quantifier is not defined on a lower bound (at least k) but asks whether the number is exactly the bound (equal to k).

We now turn to functional problems that ask for the number of solutions. The most well-known counting complexity class for this purpose is $\#\mathsf{P}$, the class containing problems asking for the number of accepting paths in a non-deterministic polynomial-time Turing machine. The prototypical $\#\mathsf{P}$-complete problem is $\#\mathsf{SAT}$, the problem of finding the number of models of a given formula, defined as

$\#\mathsf{SAT}$ **Input**:   a formula ϕ in CNF
                  Output: $|\mathsf{Mod}(\phi)|$

where the interpretations and models are restricted to the vocabulary of the formula. In this paper we use the class $\#\cdot\mathsf{coNP}$ from the counting complexity class hierarchy defined in (Hemaspaandra and Vollmer, 1995). Towards the definition of this class we first define counting problems which are in turn defined via witness functions w, which assign to a string from an input alphabet Σ a finite set of strings from an alphabet Γ. An instance for a counting problem consists of a given input string x from alphabet Σ and the task is to return $|w(x)|$, i.e. the cardinality of witnesses defined by witness function w associated with the counting problem. If $\mathcal{C}$ is a complexity class of decision problems, then $\#\cdot\mathcal{C}$ is the class of all counting problems for whose witness function w it holds that

- for every input string x, every $y \in w(x)$ is polynomially bounded by x; and

- the decision problem of deciding $y \in w(x)$ for given strings x and y is in $\mathcal{C}$.

For example, for $\#\mathsf{SAT}$ the witness function is $\mathsf{Mod}(\phi)$ for input strings ϕ corresponding to formulas. It holds that $\#\cdot\mathsf{P} = \#\mathsf{P}$ and $\#\cdot\mathsf{P} \subseteq \#\cdot\mathsf{coNP}$. An important type of reductions used for classes like $\#\cdot\mathsf{coNP}$ are subtractive reductions (Durand et al., 2005). Let $\#A$ and $\#B$ be counting problems. We identify A and B with their witness functions, i.e., we denote their witness

sets by $A(x)$ and $B(y)$ for input strings x and y. The counting problem $\#A$ reduces to $\#B$ via a strong subtractive reduction if there exist polynomial-time computable functions f and g s.t. for each input string x we have $B(f(x)) \subseteq B(g(x))$ and $|A(x)| = |B(g(x))| - |B(f(x))|$. Intuitively, we may overcount the solutions and carefully subtract what we overcounted. A strong subtractive reduction is called parsimonious if $B(f(x)) = \emptyset$ for all input strings x, i.e. $|A(x)| = |B(g(x))|$. Subtractive reductions are the transitive closure of strong subtractive reductions.

An illustrative example for subtractive reductions, which we took from (Durand et al., 2005), is that of showing $\#\mathsf{P}$ hardness for the problem of counting the number of models of a formula in *disjunctive normal form* (DNF), i.e., a Boolean formula consisting of a disjunction of conjunctions of literals. For a given Boolean formula in CNF, say ϕ, we reduce this formula to $\phi' \equiv \neg\phi$ in DNF, which is simply the negated formula ϕ where we propagated the negations to literals. It is immediate that ω is a model of ϕ iff ω does not satisfy ϕ'. This implies that $|\mathsf{Mod}(\phi)| = |\mathsf{Mod}(\top)| - |\mathsf{Mod}(\phi')|$, or, in other words, that the number of models of ϕ is equal to the number of any interpretation (on the vocabulary of ϕ, which we here for representation's sake write simply as $\top$) minus the models of ϕ' (the non-models of ϕ). This is a subtractive reduction from $\#\mathsf{SAT}$ to the problem of counting the models of a Boolean formula in DNF. This reduction is illustrative in two further senses. First, one instance created by the subtractive reduction ($\top$) is, in fact, not hard at all to solve. This holds for several subtractive reductions used to show hardness (and for all formal results relying on subtractive reductions in this paper). That is, many subtractive reductions create two instances, one easy (e.g. in P) and one hard to solve. Second, it should be noted that, under complexity theoretic assumptions, the problem of counting the number of models of a Boolean formula in DNF *cannot* be $\#\mathsf{P}$ hard under parsimonious reductions. To see this, assume that a parsimonious reduction f exists. Then, by definition, it holds that, for a given formula ϕ in CNF, we have $f(\phi) = \phi'$, and $\mathsf{Mod}(\phi) = \mathsf{Mod}(\phi')$. It follows that ϕ and ϕ' are equi-satisfiable (ϕ is satisfiable iff ϕ' is satisfiable). This implies that, for the decision problem SAT, to decide satisfiability of ϕ, we can reduce it to ϕ' and check satisfiability of ϕ'. However, satisfiability of Boolean formulas in DNF is polynomial time decidable (implying that P and NP coincide).

Another important property of subtractive reductions is that the counting complexity classes we utilize in this paper are all closed under subtractive reductions. Particularly for counting complexity classes the issue of closed-

ness of reductions is not a trivial one (Toda and Watanabe, 1992); however the classes we use in this paper are closed for the reductions we apply. For the decision problems regarding counting, closedness was proven in (Wagner, 1986, Theorem 4, item 6)).

The class #·coNP, which is the main counting class we utilize in this paper, contains several natural counting problems complete for this class (partially hard under subtractive reductions), including problems in the field of knowledge representation and reasoning, e. g. counting the number of explanations in the context of abduction (Hermann and Pichler, 2010).

3. Inconsistency Measures

Let $\mathbb{R}_{\geq 0}^{\infty}$ be the set of non-negative real values including ∞. Inconsistency measures are functions $\mathcal{I} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ that aim at assessing the severity of inconsistency in a knowledge base $\mathcal{K}$, cf. (Grant and Hunter, 2011). The basic idea is that the larger the inconsistency in $\mathcal{K}$ the larger the value $\mathcal{I}(\mathcal{K})$. However, inconsistency is a concept that is not easily quantified and there have been quite a number of proposals for inconsistency measures so far, in particular for classical propositional logic, see below. Formally, we define inconsistency measures as follows, cf. e. g. (Hunter and Konieczny, 2008).

Definition 2. An inconsistency measure $\mathcal{I}$ is a function $\mathcal{I} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ satisfying $\mathcal{I}(\mathcal{K}) = 0$ if and only if $\mathcal{K}$ is consistent, for all $\mathcal{K} \in \mathbb{K}$.

Here, we select a set of 17 inconsistency measures from the literature in order to conduct our analysis on computational complexity, taken from (Hunter and Konieczny, 2010; Grant and Hunter, 2011; Knight, 2002; Thimm, 2016b; Grant and Hunter, 2013; Xiao and Ma, 2012; Doder et al., 2010; Ammoura et al., 2017; Besnard, 2016; Jabbour et al., 2014a; Jabbour, 2016). All selected measures are indeed different as each satisfies a different set of rationality postulates from the literature (Thimm, 2017b). Moreover, our selection is representative in the sense that all major classes of inconsistency measures is present. In particular, we selected measures based on minimal inconsistent sets, e. g. (Hunter and Konieczny, 2008), measures based on maximal consistent sets (Grant and Hunter, 2011; Doder et al., 2010), measures based on paraconsistent logics (Grant and Hunter, 2011), measures based on probability theory (Knight, 2002), measures based on distances (Grant and Hunter, 2013), and several others. Some measures such as $\mathcal{I}_{LP_m}$ (Hunter and

11

Konieczny, 2010) are quite similar to some selected measure and are omitted. Measures for other formalisms than propositional logic such as (Zhou et al., 2009; Thimm, 2013; Mu et al., 2014; Condotta et al., 2016; Amgoud and Ben-Naim, 2017) and very recent developments such as (Jabbour et al., 2017; Thimm, 2017a; De Bona and Hunter, 2017; De Bona et al., 2018) are also not covered. We briefly introduce the selected measures in this section for the sake of completeness, but we refer for a detailed explanation to the corresponding original papers.

The measure $\mathcal{I}_d(\mathcal{K})$ is usually referred to as a baseline for inconsistency measures as it only distinguishes between consistent and inconsistent knowledge bases.

Definition 3 (Hunter and Konieczny (2008)). The *drastic inconsistency measure* $\mathcal{I}_d : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_d(\mathcal{K}) = \begin{cases} 1 & \text{if } \mathcal{K} \models \perp \\ 0 & \text{otherwise} \end{cases}$$

for $\mathcal{K} \in \mathbb{K}$.

In other words, $\mathcal{I}_d(\mathcal{K}) = 1$ if and only if $\mathcal{K}$ is inconsistent (and 0 otherwise).

Example 1. Consider again $\mathcal{K}_1$ and $\mathcal{K}_2$ from Section 1:

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

As both $\mathcal{K}_1$ and $\mathcal{K}_2$ are inconsistent we obtain $\mathcal{I}_d(\mathcal{K}_1) = \mathcal{I}_d(\mathcal{K}_2) = 1$.

Minimal inconsistent subsets play a crucial role in inconsistency measurement as they can be seen as the atomic conflicts within an inconsistent knowledge base. Accordingly, many inconsistency measure derive numerical values from the set $\mathsf{MI}(\mathcal{K})$. The most straightforward approach to do so, is by taking the cardinality of this set as a measure of inconsistency.

Definition 4 (Hunter and Konieczny (2008)). The MI-*inconsistency measure* $\mathcal{I}_{\mathsf{MI}} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{\mathsf{MI}}(\mathcal{K}) = |\mathsf{MI}(\mathcal{K})|$$

for $\mathcal{K} \in \mathbb{K}$.

The measure $\mathcal{I}_\mathsf{MI}$ can be regarded as too coarse as it treats minimal inconsistent subsets of different sizes in the same way. However, it has been noted—see e. g. (Hunter and Konieczny, 2008)—that the larger a minimal inconsistent subset, the less severe the inherent inconsistency should be regarded. The motivation for this perspective can given by the *lottery paradox* (Kyburg, 1961): Consider a lottery of n tickets and let a_i be the proposition that ticket i, $i = 1, \ldots, n$ will win. It is known that exactly one ticket will win $(a_1 \vee \ldots \vee a_n)$ but each ticket owner assumes that his ticket will not win $(\neg a_i, i = 1, \ldots, n)$. For $n = 1000$ it is reasonable for each ticket owner to believe that he will not win but for e. g., $n = 2$ it is not. Therefore larger minimal inconsistent subsets can be regarded less inconsistent than smaller ones. The following inconsistency measure takes this into account by weighing each minimal inconsistent subset with its reciprocal cardinality.

Definition 5 (Hunter and Konieczny (2008))**.** The MI^c-*inconsistency measure* $\mathcal{I}_\mathsf{MI^c} : \mathbb{K} \to \mathbb{R}_{\geq 0}^\infty$ is defined as

$$\mathcal{I}_\mathsf{MI^c}(\mathcal{K}) = \sum_{M \in \mathsf{MI}(\mathcal{K})} \frac{1}{|M|}$$

for $\mathcal{K} \in \mathbb{K}$.

Example 2. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Here we have

$$\mathsf{MI}(\mathcal{K}_1) = \{\{a, \neg a \wedge \neg b\}\}$$
$$\mathsf{MI}(\mathcal{K}_2) = \{\{a, \neg a\}, \{b, \neg b\}\}$$

Therefore we obtain $\mathcal{I}_\mathsf{MI}(\mathcal{K}_1) = 1$ and $\mathcal{I}_\mathsf{MI}(\mathcal{K}_2) = 2$ and

$$\mathcal{I}_\mathsf{MI^c}(\mathcal{K}_1) = \frac{1}{|\{a, \neg a \wedge \neg b\}|} = \frac{1}{2}$$
$$\mathcal{I}_\mathsf{MI^c}(\mathcal{K}_2) = \frac{1}{|\{a, \neg a\}|} + \frac{1}{|\{b, \neg b\}|} = 1$$

Another general approach for measuring inconsistency is by utilising many-valued logics (Thimm, 2017a). Here, we focus on the two measures $\mathcal{I}_\eta$ and $\mathcal{I}_c$, which utilise probabilistic logic and paraconsistent logic, respectively. A *probability function* P on $\mathcal{L}(\mathsf{At})$ is a function $P : \Omega(\mathsf{At}) \to [0,1]$ with $\sum_{\omega \in \Omega(\mathsf{At})} P(\omega) = 1$. We extend P to assign a probability to any formula $\phi \in \mathcal{L}(\mathsf{At})$ by defining

$$P(\phi) = \sum_{\omega \models \phi} P(\omega)$$

Let $\mathcal{P}(\mathsf{At})$ be the set of all those probability functions.

Definition 6 (Knight (2002)). The *η-inconsistency measure* $\mathcal{I}_\eta : \mathbb{K} \to \mathbb{R}^\infty_{\geq 0}$ is defined as

$$\mathcal{I}_\eta(\mathcal{K}) = 1 - \max\{\xi \mid \exists P \in \mathcal{P}(\mathsf{At}) : \forall \alpha \in \mathcal{K} : P(\alpha) \geq \xi\}$$

for $\mathcal{K} \in \mathbb{K}$.

In other words, $\mathcal{I}_\eta$ aims at maximising the probability that can be consistently assigned to each formula. The inconsistency value is then one minus this probability.

Example 3. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Consider the probability function $P_1 \in \mathcal{P}(\{a, b, c, d\})$ defined via

$$P_1(abcd) = P_1(\bar{a}\bar{b}cd) = 0.5$$
$$P_1(\omega) = 0 \quad \text{for } \omega \in \Omega(\{a, b, c, d\}) \setminus \{abcd, \bar{a}\bar{b}cd\}$$

Then we obtain

$$P_1(a) = P_1(\neg a \wedge \neg b) = 0.5$$
$$P_1(b \vee c) = P_1(d) = 1$$

and thus $P_1(\phi) \geq 0.5$ for all $\phi \in \mathcal{K}_1$. Furthermore, there can be no other P' that assigns larger probability to all $\phi \in \mathcal{K}_1$. Hence, we have $\mathcal{I}_\eta(\mathcal{K}_1) = 1 - 0.5 = 0.5$. The function P_1 can also be used to determine $\mathcal{I}_\eta(\mathcal{K}_2) = 0.5$.

α	β	$v(\alpha \wedge \beta)$	$v(\alpha \vee \beta)$	α	$v(\neg\alpha)$
T	T	T	T	T	F
T	B	B	T	B	B
T	F	F	T	F	T
B	T	B	T		
B	B	B	B		
B	F	F	B		
F	T	F	T		
F	B	F	B		
F	F	F	F		

Table 1: Truth tables for propositional three-valued logic.

A three-valued interpretation v (Priest, 1979) on At is a function $v : \mathsf{At} \to \{T, F, B\}$ where the values T and F correspond to the classical true and false, respectively. The additional truth value B stands for *both* and is meant to represent a conflicting truth value for a proposition. The function v is extended to arbitrary formulas as shown in Table 1. Then, an interpretation v satisfies a formula α, denoted by $v \models^3 \alpha$ if either $v(\alpha) = T$ or $v(\alpha) = B$. Then inconsistency can be measured by seeking an interpretation v that assigns B to a minimum number of propositions.

Definition 7 (Grant and Hunter (2011)). The *contension inconsistency measure* $\mathcal{I}_c : \mathbb{K} \to \mathbb{R}_{\geq 0}^\infty$ is defined as

$$\mathcal{I}_c(\mathcal{K}) = \min\{|v^{-1}(B)| \mid v \models^3 \mathcal{K}\}$$

for $\mathcal{K} \in \mathbb{K}$.

Example 4. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

and recall

$$\mathsf{MI}(\mathcal{K}_1) = \{\{a, \neg a \wedge \neg b\}\}$$
$$\mathsf{MI}(\mathcal{K}_2) = \{\{a, \neg a\}, \{b, \neg b\}\}$$

15

Consider $v_1 : \{a, b, c, d\} \to \{T, F, B\}$ defined via

$$v_1(a) = B \qquad v_1(b) = F \qquad v_1(c) = v_1(d) = T$$

Then $v_1 \models^3 \phi$ for all $\phi \in \mathcal{K}_1$ and there is no other v' that assigns B to fewer propositions, yielding $\mathcal{I}_c(\mathcal{K}_1) = 1$. For $v_2 : \{a, b\} \to \{T, F, B\}$ defined via

$$v_2(a) = v_2(b) = B$$

we have $v_2 \models^3 \phi$ for all $\phi \in \mathcal{K}_2$ and there is no other v' that assigns B to fewer propositions, yielding $\mathcal{I}_c(\mathcal{K}_2) = 2$.

The following measure takes both maximal consistent sets and self-contradictory formulas into account.[2]

Definition 8 (Grant and Hunter (2011))**.** The MC-*inconsistency measure* $\mathcal{I}_{mc} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{mc}(\mathcal{K}) = |\mathsf{MC}(\mathcal{K})| + |\mathsf{SC}(\mathcal{K})| - 1$$

for $\mathcal{K} \in \mathbb{K}$.

Example 5. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Here we have

$$\mathsf{MC}(\mathcal{K}_1) = \{\{a, b \vee c, d\}, \{b \vee c, \neg a \wedge \neg b, d\}\}$$
$$\mathsf{MC}(\mathcal{K}_2) = \{\{a, b\}, \{a, \neg b\}, \{\neg a, b\}, \{\neg a, \neg b\}\}$$

and $\mathsf{SC}(\mathcal{K}_1) = \mathsf{SC}(\mathcal{K}_2) = \emptyset$. Therefore we obtain

$$\mathcal{I}_{mc}(\mathcal{K}_1) = 1$$
$$\mathcal{I}_{mc}(\mathcal{K}_2) = 3$$

---

[2]Note that it is necessary to consider both, the set of maximal consistent subsets and the set of self-contradictory formulas, as knowledge base with a single maximal consistent subset is not necessarily consistent itself.

Another simple approach for measuring inconsistency is to take the number of formulas that are participating in *some* conflict.

Definition 9 (Grant and Hunter (2011))**.** The *problematic inconsistency measure* $\mathcal{I}_p : \mathbb{K} \to \mathbb{R}_{\geq 0}^\infty$ is defined as

$$\mathcal{I}_p(\mathcal{K}) = |\bigcup_{M \in \mathsf{MI}(\mathcal{K})} M|$$

for $\mathcal{K} \in \mathbb{K}$.

Example 6. We continue Example 1 and recall

$$\mathsf{MI}(\mathcal{K}_1) = \{\{a, \neg a \wedge \neg b\}\}$$
$$\mathsf{MI}(\mathcal{K}_2) = \{\{a, \neg a\}, \{b, \neg b\}\}$$

Then $\mathcal{I}_p(\mathcal{K}_1) = 2$ and $\mathcal{I}_p(\mathcal{K}_2) = 4$.

In general, consistency of a knowledge base is defined by the existence of at least one interpretation that satisfies all formulas. Another approach to measuring inconsistency is to relax this requirement and seek a minimum number of interpretations such that every formula is satisfied by one of them. More formally, a subset $H \subseteq \Omega(\mathsf{At})$ is called a *hitting set* of $\mathcal{K}$ if for every $\phi \in \mathcal{K}$ there is $\omega \in H$ with $\omega \models \phi$.[3]

Definition 10 (Thimm (2016b))**.** The *hitting-set inconsistency measure* $\mathcal{I}_{hs} : \mathbb{K} \to \mathbb{R}_{\geq 0}^\infty$ is defined as

$$\mathcal{I}_{hs}(\mathcal{K}) = \min\{|H| \mid H \text{ is a hitting set of } \mathcal{K}\} - 1$$

for $\mathcal{K} \in \mathbb{K}$ with $\min \emptyset = \infty$.

Example 7. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

The set $H_1 = \{abcd, \bar{a}\bar{b}cd\}$ is a hitting set of both $\mathcal{K}_1$ and $\mathcal{K}_2$ and there is no smaller set that is a hitting set. Therefore we obtain $\mathcal{I}_{hs}(\mathcal{K}_1) = \mathcal{I}_{hs}(\mathcal{K}_2) = 1$.

---

[3]Note that, formally, H is a hitting set of $\{\mathsf{Mod}(\phi_1), \ldots, \mathsf{Mod}(\phi_n)\}$ for $\mathcal{K} = \{\phi_1, \ldots, \phi_n\}$ but we say that H is a hitting set of $\mathcal{K}$ for reasons of brevity.

In (Grant and Hunter, 2013) three families of inconsistency measures are presented that use *distances* between the models of the formulas in order to measure inconsistency. An *interpretation distance* d is a function $d : \Omega(\mathsf{At}) \times \Omega(\mathsf{At}) \to [0, \infty)$ that satisfies (let $\omega, \omega', \omega'' \in \Omega(\mathsf{At})$)

1. $d(\omega, \omega') = 0$ if and only if $\omega = \omega'$ (*reflexivity*),
2. $d(\omega, \omega') = d(\omega', \omega)$ (*symmetry*), and
3. $d(\omega, \omega'') \leq d(\omega, \omega') + d(\omega', \omega'')$ (*triangle inequality*).

One prominent example of such a distance is the *Dalal distance* d_{d} defined via

$$d_{\mathrm{d}}(\omega, \omega') = |\{a \in \mathsf{At} \mid \omega(a) \neq \omega'(a)\}|$$

for all $\omega, \omega' \in \Omega(\mathsf{At})$. If $X \subseteq \Omega(\mathsf{At})$ is a set of interpretations we define $d_{\mathrm{d}}(X, \omega) = \min_{\omega' \in X} d_{\mathrm{d}}(\omega', \omega)$ (if $X = \emptyset$ we define $d_{\mathrm{d}}(X, \omega) = \infty$). For definitions 11, 12, and 13 below and our treatment of these measures in this paper we assume d_{d} fixed but the measures could be defined using arbitrary distances (Grant and Hunter, 2013).[4]

Definition 11 (Grant and Hunter (2013)). *The Σ-distance inconsistency measure $\mathcal{I}_{\mathrm{dalal}}^{\Sigma} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as*

$$\mathcal{I}_{\mathrm{dalal}}^{\Sigma}(\mathcal{K}) = \min\left\{\sum_{\alpha \in \mathcal{K}} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) \mid \omega \in \Omega(\mathsf{At})\right\}$$

for $\mathcal{K} \in \mathbb{K}$.

Definition 12 (Grant and Hunter (2013)). *The max-distance inconsistency measure $\mathcal{I}_{\mathrm{dalal}}^{\max} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as*

$$\mathcal{I}_{\mathrm{dalal}}^{\max}(\mathcal{K}) = \min\left\{\max_{\alpha \in \mathcal{K}} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) \mid \omega \in \Omega(\mathsf{At})\right\}$$

for $\mathcal{K} \in \mathbb{K}$.

---

[4]Note that our complexity assessments regarding membership results on these measures remain to be true for other distance measures as long as they can be computed in polynomial time.

Definition 13 (Grant and Hunter (2013))**.** The *hit-distance inconsistency measure* $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}(\mathcal{K}) = \min\left\{|\{\alpha \in \mathcal{K} \mid d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) > 0\}| \mid \omega \in \Omega(\mathsf{At})\right\}$$

for $\mathcal{K} \in \mathbb{K}$.

In other words, $\mathcal{I}_{\mathrm{dalal}}^{\Sigma}$ seeks an interpretation such that the sum of its distances to the models of each individual formula is minimum. The measure $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{max}}$ seeks an interpretation such that the maximum of its distances to the models of each individual formula is minimum. Finally, the measure $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}$ seeks an interpretation that satisfies a maximum number of formulas.

Example 8. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Observe that for the interpretation $\omega_1 = a\bar{b}cd \in \Omega(\{a, b, c, d\})$ we have

$$d_{\mathrm{d}}(\mathsf{Mod}(a), \omega_1) = 0$$
$$d_{\mathrm{d}}(\mathsf{Mod}(b \vee c), \omega_1) = 0$$
$$d_{\mathrm{d}}(\mathsf{Mod}(\neg a \wedge \neg b), \omega_1) = 1$$
$$d_{\mathrm{d}}(\mathsf{Mod}(d), \omega_1) = 0$$

and therefore $\sum_{\alpha \in \mathcal{K}_1} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega_1) = 1$. There is no other interpretation ω' with a smaller total distance, so we have $\mathcal{I}_{\mathrm{dalal}}^{\Sigma}(\mathcal{K}_1) = 1$. Furthermore, we have $\max_{\alpha \in \mathcal{K}_1} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega_1) = 1$ and there is also no other interpretation ω' with a smaller maximum distance. Hence, we have $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{max}}(\mathcal{K}_1) = 1$ and similarly $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}(\mathcal{K}_1) = 1$. For $\mathcal{K}_2$ we obtain

$$\mathcal{I}_{\mathrm{dalal}}^{\Sigma}(\mathcal{K}_2) = 2$$
$$\mathcal{I}_{\mathrm{dalal}}^{\mathrm{max}}(\mathcal{K}_2) = 1$$

with a similar argumentation as above. For $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}(\mathcal{K}_2)$ observe that every interpretation ω must always falsify exactly one formula in $\{a, \neg a\}$ and exactly one formula in $\{b, \neg b\}$. Therefore we obtain $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}(\mathcal{K}_2) = 2$.

Recall, that $\mathcal{I}_p$ measures inconsistency by taking the number of formulas that participate in some conflict. Another inconsistency measure can be defined by not considering the formulas itself but the number of propositions within these formulas.

Definition 14 (Xiao and Ma (2012)). The *mv inconsistency measure* $\mathcal{I}_{mv}$: $\mathbb{K} \rightarrow \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{mv}(\mathcal{K}) = \frac{|\bigcup_{M \in \mathsf{MI}(\mathcal{K})} \mathsf{At}(M)|}{|\mathsf{At}(\mathcal{K})|}$$

for $\mathcal{K} \in \mathbb{K}$.

In other words, $\mathcal{I}_{mv}$ counts all propositions that appear in formulas participating in *some* conflict and takes the ratio wrt. the number of all propositions.

Example 9. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

and recall

$$\mathsf{MI}(\mathcal{K}_1) = \{\{a, \neg a \wedge \neg b\}\}$$
$$\mathsf{MI}(\mathcal{K}_2) = \{\{a, \neg a\}, \{b, \neg b\}\}$$

Then we have

$$\mathcal{I}_{mv}(\mathcal{K}_1) = \frac{|\{a, b\}|}{|\{a, b, c, d\}|} = 1/2$$
$$\mathcal{I}_{mv}(\mathcal{K}_2) = \frac{|\{a, b\}|}{|\{a, b\}|} = 1$$

With $\mathcal{I}_{mc}$ we already saw one measure that uses consistent sets as a way to measure inconsistency. Another approach with similar ideas is the following measure.

Definition 15 (Doder et al. (2010)). The *nc-inconsistency measure* $\mathcal{I}_{nc}$: $\mathbb{K} \rightarrow \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{nc}(\mathcal{K}) = |\mathcal{K}| - \max\{n \mid \forall \mathcal{K}' \subseteq \mathcal{K} : |\mathcal{K}'| = n \Rightarrow \mathcal{K}' \not\models \perp\}$$

for $\mathcal{K} \in \mathbb{K}$.

Informally speaking, $\mathcal{I}_{nc}$ seeks the maximal number n such that all subsets of cardinality n are consistent. The larger this value, the lower the inconsistency, so the inconsistency value is the cardinality of $\mathcal{K}$ minus n.

Example 10. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Note that for both $\mathcal{K}_1$ and $\mathcal{K}_2$ we can find subsets of size 2 that are inconsistent: $\{a, \neg a \wedge \neg b\}$ for $\mathcal{K}_1$ and $\{a, \neg a\}$ for $\mathcal{K}_2$. Furthermore, all one-element subsets of $\mathcal{K}_1$ and $\mathcal{K}_2$ are consistent, respectively. Therefore, we obtain

$$\mathcal{I}_{nc}(\mathcal{K}_1) = 3$$
$$\mathcal{I}_{nc}(\mathcal{K}_2) = 3$$

An even more elaborate approach to take consistent subsets into account is the following. A set of maximal consistent subsets $\mathcal{C} \subseteq \mathsf{MC}(\mathcal{K})$ is called an MC-*cover* if

$$\bigcup_{C \in \mathcal{C}} C = \mathcal{K}$$

An MC-cover $\mathcal{C}$ is *normal* if no proper subset of $\mathcal{C}$ is an MC-cover. A normal MC-cover is maximal if

$$\lambda(\mathcal{C}) = |\bigcap_{C \in \mathcal{C}} C|$$

is maximal for all normal MC-covers.

Definition 16 (Ammoura et al. (2017))**.** The *MCSC inconsistency measure* $\mathcal{I}_{mcsc} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{mcsc}(\mathcal{K}) = |\mathcal{K}| - \lambda(\mathcal{C})$$

for all $\mathcal{K} \in \mathbb{K}$ and any maximal MC-cover $\mathcal{C}$.

The idea behind the above measure is that a large intersection of many maximal consistent subsets indicates that many formulas are compatible with many parts of the knowledge base. Thus, the inconsistency value should be low.

Example 11. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

and recall

$$\mathsf{MC}(\mathcal{K}_1) = \{\{a, b \vee c, d\}, \{b \vee c, \neg a \wedge \neg b, d\}\}$$
$$\mathsf{MC}(\mathcal{K}_2) = \{\{a, b\}, \{a, \neg b\}, \{\neg a, b\}, \{\neg a, \neg b\}\}$$

Note that both sets in $\mathsf{MC}(\mathcal{K}_1)$ are required to cover $\mathcal{K}_1$ so there is only one normal MC cover $\mathcal{C}_1 = \mathsf{MC}(\mathcal{K}_1)$ and $\lambda(\mathcal{C}_1) = 2$, yielding $\mathcal{I}_{mcsc}(\mathcal{K}_1) = 4 - 2 = 2$. For $\mathcal{K}_2$ observe that $\mathcal{C} = \{\{a, b\}, \{\neg a, \neg b\}\}$ is a normal MC cover (but note that there are others) with $\lambda(\mathcal{C}_2) = 0$ yielding $\mathcal{I}_{mcsc}(\mathcal{K}_2) = 4 - 0 = 4$.

The measure of (Besnard, 2016) uses the approach of *variable forgetting* (Lang and Marquis, 2010) in order to measure inconsistency. For a formula ϕ let $\phi[a_1, i_1 \rightarrow \psi_1; \ldots, a_k, i_k \rightarrow \psi_k]$ denote the formula ϕ where the i_jth occurrence of the proposition a_j is replaced by the formula ψ_j, for all $j = 1, \ldots, k$. For example,

$$(a \wedge b \vee (\neg a \wedge b))[a, 2 \rightarrow \top; b, 1 \rightarrow \perp] = (a \wedge \perp \vee (\neg \top \wedge b))$$

Furthermore, for a set $S = \{\phi_1, \ldots, \phi_n\}$ let $\bigwedge S = \phi_1 \wedge \ldots \wedge \phi_n$.

Definition 17 (Besnard (2016)[5]). The *forgetting-based inconsistency measure* $\mathcal{I}_{\text{forget}} : \mathbb{K} \rightarrow \mathbb{R}_{\geq 0}^\infty$ is defined as

$$\mathcal{I}_{\text{forget}}(\mathcal{K}) = \min\{k \mid (\bigwedge \mathcal{K})[a_1, i_1 \rightarrow \phi_1; \ldots; a_k, i_k \rightarrow \phi_k] \not\models \perp, \phi_j \in \{\perp, \top\}\}$$

for all $\mathcal{K} \in \mathbb{K}$.

So $\mathcal{I}_{\text{forget}}$ seeks the minimal number of variable occurrences that need to be forgotten in order to render the knowledge base consistent.

Example 12. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

Observe that substituting the first occurrence of a in $\mathcal{K}_1$ with $\top$ yields a consistent knowledge base $\{\top, b \vee c, \neg a \wedge \neg b, d\}$. Therefore we have $\mathcal{I}_{\text{forget}}(\mathcal{K}_1) = 1$. For $\mathcal{K}_2$ we need to forget two occurrences of propositions, e. g., $\bigwedge \mathcal{K}_2[a, 1 \rightarrow \top b, 1 \rightarrow \top] = \top \wedge \neg a \wedge \top \wedge \neg b$ is consistent, yielding $\mathcal{I}_{\text{forget}}(\mathcal{K}_2) = 2$.

---

[5]Note that we give a slightly different but equivalent formalization.

The approach of (Jabbour et al., 2014a) aims at looking deeper into the structure of the set of minimal inconsistent subsets. The main rationale here is the assumption that having two disjoint minimal inconsistent subsets is a more severe inconsistency than having two minimal inconsistent subsets with a non-empty intersection, ceteris paribus. Formally, a set $\{K_1, \ldots, K_n\}$ of pairwise disjoint subsets of $\mathcal{K}$ is called a *conditional independent MUS partition* of $\mathcal{K}$, iff each K_i is inconsistent and $\mathsf{MI}(K_1 \cup \ldots \cup K_n)$ is the union of all $\mathsf{MI}(K_i)$.

Definition 18 (Jabbour et al. (2014a))**.** The *CC inconsistency measure* $\mathcal{I}_{CC} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{CC}(\mathcal{K}) = \max\{n \mid \{K_1, \ldots, K_n\} \text{ is a conditional independent MUS partition of } \mathcal{K}\}$$

for all $\mathcal{K} \in \mathbb{K}$.

Another approach with a similar motivation as $\mathcal{I}_{CC}$ is the following. A set S of sets is *independent* if for all $M, M' \in S$, if $M \neq M'$ then $M \cap M' = \emptyset$.

Definition 19 (Jabbour (2016))**.** The *independent set-based inconsistency measure* $\mathcal{I}_{\mathrm{is}} : \mathbb{K} \to \mathbb{R}_{\geq 0}^{\infty}$ is defined as

$$\mathcal{I}_{\mathrm{is}}(\mathcal{K}) = \log |\{M \subseteq \mathsf{MI}(\mathcal{K}) \mid M \text{ is independent}\}|$$

for all $\mathcal{K} \in \mathbb{K}$.

In other words, $\mathcal{I}_{\mathrm{is}}$ counts the number of independent sets that can be formed from $\mathsf{MI}(\mathcal{K})$ and takes its logarithm to appropriately normalise the measure.

Example 13. We continue Example 1 and consider

$$\mathcal{K}_1 = \{a, b \vee c, \neg a \wedge \neg b, d\}$$
$$\mathcal{K}_2 = \{a, \neg a, b, \neg b\}$$

and recall

$$\mathsf{MI}(\mathcal{K}_1) = \{\{a, \neg a \wedge \neg b\}\}$$
$$\mathsf{MI}(\mathcal{K}_2) = \{\{a, \neg a\}, \{b, \neg b\}\}$$

Note that $\mathsf{MI}(\mathcal{K}_1)$ itself is the only non-empty conditional independent MUS partition of $\mathcal{K}_1$. Similarly, $\mathsf{MI}(\mathcal{K}_2)$ is the maximal conditional independent MUS partition of $\mathcal{K}_2$ and we obtain

$$\mathcal{I}_{CC}(\mathcal{K}_1) = 1$$
$$\mathcal{I}_{CC}(\mathcal{K}_2) = 2$$

Furthermore, note that $\mathsf{MI}(\mathcal{K}_1)$ has two independent sets $\emptyset$ and $\{\{a, \neg a \wedge \neg b\}\}$. Finally, $\mathsf{MI}(\mathcal{K}_2)$ has four independent sets $\emptyset$, $\{\{a, \neg a\}\}$, $\{\{b, \neg b\}\}$, and $\{\{a, \neg a\}, \{b, \neg b\}\}$ yielding

$$\mathcal{I}_{\mathrm{is}}(\mathcal{K}_1) = \log 2$$
$$\mathcal{I}_{\mathrm{is}}(\mathcal{K}_2) = \log 4$$

This concludes the introduction of the measures we will consider in our discussion. Note that all mentioned functions are indeed inconsistency measures according to Definition 2. We recommend looking into the original papers mentioned above for more details on the individual measures, into (Thimm, 2016a, 2017b) for some further surveys, and into *Tweety@Web*[6] (Thimm, 2017c) for an online interface for computing the measures. For the reader's convenience, we summarise the formal definitions of these inconsistency measures in Figure 1.

4. Analysis of Computational Complexity

In this paper, we consider the following three decision problems for our investigation of the computational complexity of inconsistency measurement. Let $\mathcal{I}$ be some inconsistency measure.

EXACT$_{\mathcal{I}}$ **Input**: $\mathcal{K} \in \mathbb{K}$, $x \in \mathbb{R}_{\geq 0}^{\infty}$
                 Output: TRUE iff $\mathcal{I}(\mathcal{K}) = x$

UPPER$_{\mathcal{I}}$ **Input**: $\mathcal{K} \in \mathbb{K}$, $x \in \mathbb{R}_{\geq 0}^{\infty}$
                 Output: TRUE iff $\mathcal{I}(\mathcal{K}) \leq x$

LOWER$_{\mathcal{I}}$ **Input**: $\mathcal{K} \in \mathbb{K}$, $x \in \mathbb{R}_{\geq 0}^{\infty} \setminus \{0\}$
                 Output: TRUE iff $\mathcal{I}(\mathcal{K}) \geq x$

---

[6]`http://tweetyproject.org/w/incmes`

$$\mathcal{I}_d(\mathcal{K}) = \begin{cases} 1 & \text{if } \mathcal{K} \models \bot \\ 0 & \text{otherwise} \end{cases}$$

$$\mathcal{I}_{\mathsf{MI}}(\mathcal{K}) = |\mathsf{MI}(\mathcal{K})|$$

$$\mathcal{I}_{\mathsf{MI^c}}(\mathcal{K}) = \sum_{M \in \mathsf{MI}(\mathcal{K})} \frac{1}{|M|}$$

$$\mathcal{I}_\eta(\mathcal{K}) = 1 - \max\{\xi \mid \exists P \in \mathcal{P}(\mathsf{At}) : \forall \alpha \in \mathcal{K} : P(\alpha) \geq \xi\}$$

$$\mathcal{I}_c(\mathcal{K}) = \min\{|v^{-1}(B)| \mid v \models^3 \mathcal{K}\}$$

$$\mathcal{I}_{mc}(\mathcal{K}) = |\mathsf{MC}(\mathcal{K})| + |\mathsf{SC}(\mathcal{K})| - 1$$

$$\mathcal{I}_p(\mathcal{K}) = |\bigcup_{M \in \mathsf{MI}(\mathcal{K})} M|$$

$$\mathcal{I}_{hs}(\mathcal{K}) = \min\{|H| \mid H \text{ is a hitting set of } \mathcal{K}\} - 1$$

$$\mathcal{I}_{\mathrm{dalal}}^\Sigma(\mathcal{K}) = \min\{\sum_{\alpha \in \mathcal{K}} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) \mid \omega \in \Omega(\mathsf{At})\}$$

$$\mathcal{I}_{\mathrm{dalal}}^{\max}(\mathcal{K}) = \min\{\max_{\alpha \in \mathcal{K}} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) \mid \omega \in \Omega(\mathsf{At})\}$$

$$\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}(\mathcal{K}) = \min\{|\{\alpha \in \mathcal{K} \mid d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) > 0\}| \mid \omega \in \Omega(\mathsf{At})\}$$

$$\mathcal{I}_{mv}(\mathcal{K}) = \frac{|\bigcup_{M \in \mathsf{MI}(\mathcal{K})} \mathsf{At}(M)|}{|\mathsf{At}(\mathcal{K})|}$$

$$\mathcal{I}_{nc}(\mathcal{K}) = |\mathcal{K}| - \max\{n \mid \forall \mathcal{K}' \subseteq \mathcal{K} : |\mathcal{K}'| = n \Rightarrow \mathcal{K}' \not\models \bot\}$$

$$\mathcal{I}_{mcsc}(\mathcal{K}) = |\mathcal{K}| - \lambda(\mathcal{C})$$

$$\mathcal{I}_{\mathrm{forget}}(\mathcal{K}) = \min\{k \mid (\bigwedge \mathcal{K})[a_1, i_1 \to \phi_1; \ldots; a_k, i_k \to \phi_k] \not\models \bot, \phi_j \in \{\bot, \top\}\}$$

$$\mathcal{I}_{CC}(\mathcal{K}) = \max\{n \mid \{K_1, \ldots, K_n\} \text{ is a CI MUS partition of } \mathcal{K}\}$$

$$\mathcal{I}_{\mathrm{is}}(\mathcal{K}) = \log|\{M \subseteq \mathsf{MI}(\mathcal{K}) \mid M \text{ is independent}\}|$$

Figure 1: Definitions of the considered inconsistency measures

In other words, $\text{EXACT}_\mathcal{I}$ is the problem of deciding whether a given value x is the inconsistency value of a given knowledge base. The problems $\text{UPPER}_\mathcal{I}$ and $\text{LOWER}_\mathcal{I}$ are about deciding whether a given value x is an upper/lower bound of the inconsistency value of a given knowledge base, respectively. In this section we focus on the above problems for general propositional logic. We have a look at the special cases of the Horn and Krom fragment in Section 5.

Note that for any inconsistency measure $\mathcal{I}$ according to Definition 2 the decision problems $\text{EXACT}_\mathcal{I}$ and $\text{UPPER}_\mathcal{I}$ are NP-hard. This follows from the fact that for the subclass of knowledge bases consisting of a single formula in CNF, deciding whether $\mathcal{I}(\mathcal{K}) = 0$ is equivalent to SAT. Similarly, the problem $\text{LOWER}_\mathcal{I}$ is coNP-hard as deciding whether $\mathcal{I}(\mathcal{K}) \geq x$ for any $x > 0$ entails that $\mathcal{K}$ is inconsistent, which itself is equivalent to UNSAT.

Furthermore, we consider the following natural function problem for our investigation:

$\text{VALUE}_\mathcal{I}$ **Input**: $\mathcal{K} \in \mathbb{K}$
                  Output: The value of $\mathcal{I}(\mathcal{K})$

Table 2 gives an overview on the technical results shown in the remainder of this paper. As can be seen, most measures fall into the first level of the polynomial hierarchy $(\mathcal{I}_d, \mathcal{I}_\eta, \mathcal{I}_c, \mathcal{I}_{hs}, \mathcal{I}^\Sigma_{\text{dalal}}, \mathcal{I}^{\max}_{\text{dalal}}, \mathcal{I}^{\text{hit}}_{\text{dalal}}, \mathcal{I}_{mcsc}, \mathcal{I}_{\text{forget}})$, where the decision problems $\text{UPPER}_\mathcal{I}$ and $\text{LOWER}_\mathcal{I}$ can be shown to be NP-complete and coNP-complete, respectively, and thus not computationally harder than SAT and UNSAT problems, respectively. The remaining measures are either on the second level of the polynomial hierarchy $(\mathcal{I}_p, \mathcal{I}_{mv}, \mathcal{I}_{nc})$, on the third level $(\mathcal{I}_{CC})$, or involve counting (sub)problems whose complexity goes beyond the third level of the polynomial hierarchy $(\mathcal{I}_{\mathsf{MI}}, \mathcal{I}_{\mathsf{MI}^\mathsf{c}}, \mathcal{I}_{mc}, \mathcal{I}_{\text{is}})$.

It is interesting to see that, judging from the results shown in Table 2, there is a certain kind of "uniformity" of the considered inconsistency measures wrt. their complexity. In fact, all of the considered measures that fall on the first level of the polynomial hierarchy have exactly the same complexity wrt. UPPER and LOWER, the same holds for the measures of the second level of the polynomial hierarchy, and all but one $(\mathcal{I}_{\mathsf{MI}^\mathsf{c}})$ for the measures that are outside the polynomial hierarchy. There is only one measure that falls into the third level of the polynomial hierarchy, leaving comparisons with similar complex measures difficult. While this "uniform" complexity is easier to explain for the measures on the first level of the polynomial hierarchy (as said before, $\text{UPPER}_\mathcal{I}$ is NP hard directly from definition), the uniformity is still

observable on the second level of the polynomial hierarchy, and for those that involve counting classes. A potential explanation for this is that all measures whose complexity lies on the second level of the polynomial hierarchy exhibit "similar" (yet different) subproblems that have relations to inconsistent subsets; analogously, for those measures involving counting complexity classes, counting refers often to structures related to inconsistent subsets. However, $\mathcal{I}_{mc}$, for instance, falls outside this schema, when compared to $\mathcal{I}_{\mathsf{MI}}$ (the former involving counting of maximal consistent subsets compared to counting minimal inconsistent subsets of the latter). Another observation is that there is a "shift" when going from UPPER (LOWER) from a measure on the first level of the polynomial hierarchy, which are $\mathsf{NP} = \Sigma_1^p$-complete ($\mathsf{coNP} = \Pi_1^p$-complete) to the corresponding problem for a measure on the second level of the polynomial hierarchy, which is Π_2^p-complete (Σ_2^p-complete). That is, there is a shift from Σ_1^p to Π_2^p for UPPER, which, we think, is mainly due to the fact that problems on the second level rely on subproblems connecting to inconsistent subsets. For instance, for $\text{LOWER}_{\mathcal{I}_p}$, the structures that contribute to the inconsistency value are coNP-hard to check, and, on top, a non-deterministic procedure seems to be required to find them. However, we think that the kind of uniformity exhibited by the considered inconsistency measures is not inherent to the field of inconsistency measurement as a whole. For instance, $\text{UPPER}_{\mathcal{I}_{CC}}$ is in Σ_3^p (not "shifting" again from Π_2^p for a problem on the second level).

Before we continue with the details of the technical results, we make some general observations first. In particular, in order to provide insights into the computational complexity of the problem $\text{VALUE}_{\mathcal{I}}$ it is useful to investigate the number of values an inconsistency measure can attain for knowledge bases of a given size, cf. (Thimm, 2016a) for a more detailed discussion of this topic.

Definition 20. Let ϕ be a formula. The *length* $\mathtt{len}(\phi)$ of ϕ is recursively defined as

$$\mathtt{len}(\phi) = \begin{cases} 1 & \text{if } \phi \in \mathsf{At} \\ 1 + \mathtt{len}(\phi') & \text{if } \phi = \neg\phi' \\ 1 + \mathtt{len}(\phi_1) + \mathtt{len}(\phi_2) & \text{if } \phi = \phi_1 \wedge \phi_2 \\ 1 + \mathtt{len}(\phi_1) + \mathtt{len}(\phi_2) & \text{if } \phi = \phi_1 \vee \phi_2 \end{cases}$$

Define the *length* $\mathtt{len}(\mathcal{K})$ of a knowledge base $\mathcal{K}$ via $\mathtt{len}(\mathcal{K}) = \sum_{\phi \in \mathcal{K}} \mathtt{len}(\phi)$.

	$\mathrm{EXACT}_{\mathcal{I}}$	$\mathrm{UPPER}_{\mathcal{I}}$	$\mathrm{LOWER}_{\mathcal{I}}$	$\mathrm{VALUE}_{\mathcal{I}}$
$\mathcal{I}_d$	$\mathsf{D}_1^p \cap \mathsf{coD}_1^p$	NP-c	coNP-c	FNP
$\mathcal{I}_{\mathsf{MI}}$	$\mathsf{C}_=\mathsf{NP}$-h	CNP-c	CNP-c	$\#{\cdot}\mathsf{coNP}$-c
$\mathcal{I}_{\mathsf{MI}^{\mathsf{c}}}$	$\mathsf{C}_=\mathsf{NP}$-h	CNP-h	CNP-h	$FP^{\#{\cdot}\mathsf{coNP}}$
$\mathcal{I}_\eta$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[n]}$
$\mathcal{I}_c$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$-c
$\mathcal{I}_{mc}$	$\mathsf{C}_=\mathsf{NP}$-h	CNP-c	CNP-c	$\#{\cdot}\mathsf{coNP}$-c†
$\mathcal{I}_p$	D_2^p-c	Π_2^p-c	Σ_2^p-c	$\mathsf{FP}^{\Sigma_2^p[\log n]}$
$\mathcal{I}_{hs}$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$
$\mathcal{I}_{\mathrm{dalal}}^{\Sigma}$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$-c
$\mathcal{I}_{\mathrm{dalal}}^{\max}$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$
$\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$-c
$\mathcal{I}_{mv}$	D_2^p-c	Π_2^p-c	Σ_2^p-c	$\mathsf{FP}^{\Sigma_2^p[\log n]}$
$\mathcal{I}_{nc}$	D_2^p	Π_2^p-c	Σ_2^p-c	$\mathsf{FP}^{\Sigma_2^p[\log n]}$
$\mathcal{I}_{mcsc}$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$
$\mathcal{I}_{\mathrm{forget}}$	D_1^p-c	NP-c	coNP-c	$\mathsf{FP}^{\mathsf{NP}[\log n]}$-c
$\mathcal{I}_{CC}$	D_3^p	Π_3^p	Σ_3^p	$\mathsf{FP}^{\Sigma_3^p[\log n]}$
$\mathcal{I}_{\mathrm{is}}$	$\mathsf{C}_=\mathsf{NP}$-h	CNP-c	CNP-c	$\#{\cdot}\mathsf{coNP}$-c‡

Table 2: Computational complexity of the considered inconsistency measures (all statements are membership statements, an additionally attached "-c" ("-h") also indicates completeness (hardness) for the class); we note that all hardness results for $\#{\cdot}\mathsf{coNP}$ are under subtractive reductions; †we show complexity of the (minor) variation that omits subtracting one from the result; ‡we consider here the problem variant that does not apply a logarithm on the result; the row regarding $\mathcal{I}_{mv}$ has been shown in (Xiao and Ma, 2012).

Definition 21. For an inconsistency measure $\mathcal{I}$ and $n \in \mathbb{N}$ define $\mathcal{C}_{\mathcal{I}}(n) = \{\mathcal{I}(\mathcal{K}) \mid \mathtt{len}(\mathcal{K}) \leq n\}$, i. e., $\mathcal{C}_{\mathcal{I}}(n)$ is the set of different inconsistency values that can be attained by $\mathcal{I}$ on knowledge bases of maximal length n.

For most measures, the number of attainable values is polynomial in the size of the knowledge base, see also the results in (Thimm, 2016a). Moreover, each attainable value itself can also be represented concisely.

Lemma 1. For $\mathcal{I} \in \{\mathcal{I}_d, \mathcal{I}_c, \mathcal{I}_p, \mathcal{I}_{hs}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{\max}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{mv}, \mathcal{I}_{nc}, \mathcal{I}_{mcsc}, \mathcal{I}_{forget}, \mathcal{I}_{CC}\}$ there is $k \in \mathbb{N}$ such that $|\mathcal{C}_{\mathcal{I}}(n)| \in O(n^k)$. Moreover, each $x \in \mathcal{C}_{\mathcal{I}}(n)$ can be represented in space $O(\log n)$.

The above lemma basically states that the number of different values most of the investigated inconsistency measures can attain on knowledge bases up to a certain size, is polynomially bounded by this size and the values are "small". Note that the statement is not true in general for the remaining measures $\mathcal{I}_{\mathsf{MI}}$, $\mathcal{I}_{\mathsf{MI^c}}$, $\mathcal{I}_{\eta}$, $\mathcal{I}_{mc}$, and $\mathcal{I}_{\mathsf{is}}$ as a knowledge base may have an exponential number of minimal (in-)consistent subsets.

Lemma 2. *For $\mathcal{I} \in \{\mathcal{I}_{\mathsf{MI}}, \mathcal{I}_{\eta}, \mathcal{I}_{mc}\}$ there is $k \in \mathbb{N}$ such that $|\mathcal{C}_{\mathcal{I}}(n)| \in O(2^{n^k})$. Moreover, each $x \in \mathcal{C}_{\mathcal{I}}(n)$ can be represented in space $O(n^k)$.*

Lemma 3. *For $\mathcal{I} \in \{\mathcal{I}_{\mathsf{MI^c}}, \mathcal{I}_{\mathsf{is}}\}$, $|\mathcal{C}_{\mathcal{I}}(n)| \in O(2^{2^n})$. Moreover, each $x \in \mathcal{C}_{\mathcal{I}}(n)$ can be represented in space $O(n^k)$.*

Lemmas 1–3 are in particular useful in combination with (exact) complexity bounds for problems $\mathrm{UPPER}_{\mathcal{I}}$ and $\mathrm{LOWER}_{\mathcal{I}}$. If, e. g., $\mathrm{UPPER}_{\mathcal{I}}$ is in complexity class C for a measure $\mathcal{I}$ for which it holds that e. g. $|\mathcal{C}_{\mathcal{I}}(n)| \in O(n^k)$, we can find the exact value of $\mathcal{I}(\mathcal{K})$ for a knowledge base $\mathcal{K}$ with binary search on the possible values requiring thus to solve just a logarithmic number of consecutive problems in C. These considerations are summarised in the following result.

Lemma 4. *Let $\mathcal{I}$ be some inconsistency measure and $i > 0$ an integer. If $\mathrm{UPPER}_{\mathcal{I}}$ is in Σ_i^p or in Π_i^p, the range $\mathcal{C}_{\mathcal{I}}(n)$ is representable in space $O(\log f(n))$ and $|\mathcal{C}_{\mathcal{I}}(n)| \in O(f(n))$ for some function $f : \mathbb{N} \to \mathbb{N}$, then $\mathrm{VALUE}_{\mathcal{I}}$ is in $\mathsf{FP}^{\Sigma_i^p[\log f(n)]}$.*

The decision problems $\mathrm{EXACT}_{\mathcal{I}}$, $\mathrm{UPPER}_{\mathcal{I}}$, and $\mathrm{LOWER}_{\mathcal{I}}$ are also related to each other. $\mathrm{UPPER}_{\mathcal{I}}$ and $\mathrm{LOWER}_{\mathcal{I}}$ are complementary to each other and $\mathrm{EXACT}_{\mathcal{I}}$ is the combination of both. However, we need another condition on inconsistency measures to see this.

Definition 22. An inconsistency measure $\mathcal{I}$ is called *well-serializable* if the following two problems can be solved in polynomial time:

1. Given $n \in \mathbb{N}$ and $x \in \mathcal{C}_{\mathcal{I}}(n)$, determine $y \in \mathcal{C}_{\mathcal{I}}(n)$ such that $y > x$ and there is no $y' \in \mathcal{C}_{\mathcal{I}}(n)$ with $y > y' > x$.
2. Given $n \in \mathbb{N}$ with $n > 0$ and $x \in \mathcal{C}_{\mathcal{I}}(n)$, determine $y \in \mathcal{C}_{\mathcal{I}}(n)$ such that $y < x$ and there is no $y' \in \mathcal{C}_{\mathcal{I}}(n)$ with $y < y' < x$.

In other words, a measure is called well-serializable if the immediate successor and predecessor of a value of $\mathcal{I}$ can be efficiently determined. Note that almost all considered measures satisfy this property.

Lemma 5. *The measures* $\mathcal{I}_d, \mathcal{I}_{\mathsf{MI}}, \mathcal{I}_\eta, \mathcal{I}_c, \mathcal{I}_{mc}, \mathcal{I}_p, \mathcal{I}_{hs}, \mathcal{I}_{dalal}^\Sigma, \mathcal{I}_{dalal}^{\max}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{mv},$ $\mathcal{I}_{nc}, \mathcal{I}_{mcsc}, \mathcal{I}_{forget}, \mathcal{I}_{CC}, \mathcal{I}_{is}$ *are well-serializable.*

The only measure absent from the above result is $\mathcal{I}_{\mathsf{MI^c}}$, for which we do not believe that determining the successor/predecessor of an inconsistency value can be done efficiently. As $\mathcal{I}_{is}$, the measure $\mathcal{I}_{\mathsf{MI^c}}$ exhibits double-exponential many inconsistency values, see Lemma 3. However, $\mathcal{I}_{is}$ is defined to be the logarithm of an integer-valued measure, so successors and predecessors can be efficiently computed by inverting the logarithm, adding or subtracting one and taking the logarithm again (see proof of Lemma 5 in Appendix A). For $\mathcal{I}_{\mathsf{MI^c}}$, no such simple inversion exists. Furthermore, applying a binary search on all attainable values to obtain successors and predecessors—as we did for $\mathcal{I}_\eta$ in the proof of Lemma 5—still needs exponential time.

However, in case of well-serializable inconsistency measures we can relate the complexity of $\text{EXACT}_\mathcal{I}$, $\text{UPPER}_\mathcal{I}$, and $\text{LOWER}_\mathcal{I}$ as follows.

Lemma 6. *Let $\mathcal{I}$ be some well-serializable inconsistency measure and $i > 0$ an integer. Let $C \in \{\Sigma_i^p, \Pi_i^p\}$.*

- $\text{UPPER}_\mathcal{I}$ *is C-complete iff $\text{LOWER}_\mathcal{I}$ is co-C-complete;*

- *if $\text{UPPER}_\mathcal{I}$ or $\text{LOWER}_\mathcal{I}$ is in C, then $\text{EXACT}_\mathcal{I}$ is in D_i^p.*

Lemma 4 and Lemma 6 taken together imply that showing the complexity of either $\text{UPPER}_\mathcal{I}$ or $\text{LOWER}_\mathcal{I}$ gives crucial insights into the computation of the measure $\mathcal{I}$.

In the following, we give details on the technical contributions summarized in Table 2. We structure our presentation by first discussing the problems on the first level of the polynomial hierarchy (Section 4.1), then those on the second level (Section 4.2), then those on the third level (Section 4.3), and finally those beyond the third level of the polynomial hierarchy (Section 4.4).

4.1. Problems on the first level of the polynomial hierarchy

In this section we discuss the measures $\mathcal{I}_d, \mathcal{I}_\eta, \mathcal{I}_c, \mathcal{I}_{hs}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{dalal}^\Sigma, \mathcal{I}_{dalal}^{\max},$ $\mathcal{I}_{forget}$, and $\mathcal{I}_{mcsc}$ and show that the corresponding decision and function problems reside on the first level of the polynomial hierarchy. For all these measures, we start by showing that $\text{UPPER}_\mathcal{I}$ is NP-complete and then utilize Lemmas 4 and 6 to gain insights on the remaining problems. If we can strengthen the bounds provided by Lemmas 4 and 6 we report on those results afterwards.

Before we proceed, we give an overview on the problems and reductions used in the proofs. We already noted that the problems $\text{EXACT}_{\mathcal{I}}$ and $\text{UPPER}_{\mathcal{I}}$ are NP-hard for inconsistency measures according to Definition 2 by reducing SAT to them. So to show NP-completeness for problems $\text{UPPER}_{\mathcal{I}}$, we only have to show membership in NP, which can usually be achieved by sketching a simple non-deterministic algorithm that runs in polynomial time. For problems $\text{EXACT}_{\mathcal{I}}$ we will usually show completeness in the class D_1^p. In order to do so we use the canonical D_1^p-complete problem SAT-UNSAT defined as

SAT-UNSAT   **Input**:   a pair (ϕ, ψ) of formulas in CNF
                        Output: TRUE iff $\mathsf{Mod}(\phi) \neq \emptyset$ and $\mathsf{Mod}(\psi) = \emptyset$

Note that we assume that ϕ and ψ do not share any atoms (otherwise we can rename the atoms in one of them without changing satisfiability).

As a matter of fact, we will use the same polynomial reduction (and some minor variant) for transforming an instance of SAT-UNSAT to an instance of $\text{EXACT}_{\mathcal{I}}$, independently of the actual inconsistency measure $\mathcal{I}$. More precisely, define the function $R_{\text{SAT-UNSAT}}$ mapping pairs of formulas in CNF to a knowledge base via

$$R_{\text{SAT-UNSAT}}(\phi, \psi) = \{\phi, a, \neg a \vee \psi\}$$

for all formulas ϕ and ψ in CNF and a some fresh proposition not appearing in either ϕ or ψ. In the following results showing D_1^p-completeness for $\text{EXACT}_{\mathcal{I}}$, we will use this reduction by showing statements such as "(ϕ, ψ) is a positive instance for SAT-UNSAT iff $R_{\text{SAT-UNSAT}}(\phi, \psi)$ has a particular inconsistency value x wrt. $\mathcal{I}$". Note that $R_{\text{SAT-UNSAT}}$ is indeed a polynomial transformation. We can make some general observations on the reduction $R_{\text{SAT-UNSAT}}$ that will help understand the proofs:

1. If (ϕ, ψ) is a positive instance of SAT-UNSAT i.e., ϕ is satisfiable while ψ is not, then $R_{\text{SAT-UNSAT}}(\phi, \psi)$ is inconsistent. However, the inconsistency is somewhat "minor" as by considering a model of ϕ and extending this model by setting a to false, only the second formula a is violated.

2. If (ϕ, ψ) is a negative instance of SAT-UNSAT, we can differentiate the following cases
   (a) If ϕ is unsatisfiable then $R_{\text{SAT-UNSAT}}(\phi, \psi)$ is inconsistent, but—in difference to the case above—the inconsistency is "more severe" as ϕ is likely to be more complex than the simple formula a.

(b) If ϕ is satisfiable and ψ is satisfiable then $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ is consistent.

In the proofs the informal terms "minor" and "more severe" will be made more concrete by considering actual inconsistency measures. However, note that inconsistency measures who are able to differentiate these two cases can solve SAT-UNSAT.

We will also use a minor variant $R'_{\mathsf{SAT\text{-}UNSAT}}$ of the above reduction defined as follows. For a formula ϕ, let ϕ_{copy} be an exact copy of ϕ with different atoms, e.g. if $\phi = a \vee \neg b$ then $\phi_{\mathrm{copy}} = a_{\mathrm{copy}} \vee \neg b_{\mathrm{copy}}$ (note that ϕ is satisfiable if and only if ϕ_{copy} is satisfiable). Let a be a fresh atom not appearing in either ϕ, ϕ_{copy}, and ψ. Define $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ via

$$R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, \phi_{\mathrm{copy}}, a, \neg a \vee \psi\}$$

Observe that $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ as well can be constructed in polynomial time. Similar observations as those for $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ can be made for $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$.

For problems $\mathrm{VALUE}_{\mathcal{I}}$ we will usually show completeness in the class $\mathsf{FP}^{\mathsf{NP}[\log n]}$. In order to do so we use the canonical $\mathsf{FP}^{\mathsf{NP}[\log n]}$-complete problem MaxSATSize defined as

MaxSATSize   **Input**:   a set $C = \{c_1, \ldots, c_n\}$ of clauses
                        Output: $\max |\{C' \subseteq C \mid C' \not\models \bot\}|$

In other words, MaxSATSize seeks the maximum number of clauses in $C = \{c_1, \ldots, c_n\}$ that can be jointly satisfied. In contrast to the case of SAT-UNSAT we will use different transformations for MaxSATSize for each $\mathsf{FP}^{\mathsf{NP}[\log n]}$-completeness proof.

The first measure we investigate is the baseline inconsistency measure, $\mathcal{I}_d$, which is equal to 0 if the given knowledge base is consistent and 1 otherwise, making the problem $\mathrm{UPPER}_{\mathcal{I}_d}$ obviously NP-complete.

Proposition 1. $\mathrm{UPPER}_{\mathcal{I}_d}$ *is NP-complete.*

Applying Lemmas 4 and 6 we obtain the following.

Corollary 1. $\mathrm{LOWER}_{\mathcal{I}_d}$ *is coNP-complete,* $\mathrm{EXACT}_{\mathcal{I}_d}$ *is in* D_1^p, *and* $\mathrm{VALUE}_{\mathcal{I}_d}$ *is in* $\mathsf{FP}^{\mathsf{NP}[\log n]}$.

Regarding $\text{EXACT}_{\mathcal{I}_d}$, this problem is in fact also in the complementary class of D_1^p, namely in coD_1^p. The canonical complete problem for this class, SAT-OR-UNSAT is very similar to SAT-UNSAT, but a pair of formulas is a yes instance iff the first one is satisfiable *or* the second one is unsatisfiable.

| SAT-OR-UNSAT | **Input**: | a pair (ϕ, ψ) of formulas in CNF |
| | **Output**: | TRUE iff $\mathsf{Mod}(\phi) \neq \emptyset$ or $\mathsf{Mod}(\psi) = \emptyset$ |

A reduction from $\text{EXACT}_{\mathcal{I}_d}$ to SAT-OR-UNSAT is then straightforward to achieve: for an instance that asks whether the inconsistency value is 0, a pair $(\phi, \top)$ is constructed, with ϕ the conjunction of all elements in the given knowledge base, and, otherwise, if we ask whether the value is equal to 1, $(\bot, \phi)$. This shows membership in coD_1^p (and therefore $\text{EXACT}_{\mathcal{I}_d} \in \mathsf{D}_1^p \cap \mathsf{coD}_1^p$).

Corollary 2. $\text{EXACT}_{\mathcal{I}_d}$ *is in* coD_1^p.

Incidentally, $\mathcal{I}_d$ is an inconsistency measure that (directly) witnesses that NP-completeness of $\text{UPPER}_{\mathcal{I}}$ and coNP-completeness for $\text{LOWER}_{\mathcal{I}}$ does not imply D_1^p hardness for $\text{EXACT}_{\mathcal{I}}$, for some measure $\mathcal{I}$. In fact, under complexity theoretic assumptions, this seems unlikely for $\mathcal{I}_d$: suppose the contrary, then one could construct a knowledge base $\mathcal{K}$ with integer $k \in \{0,1\}$ from a pair of Boolean formulas (ϕ, ψ) s.t. $\mathcal{I}_d(\mathcal{K}) = k$ iff ϕ is satisfiable and ψ is unsatisfiable. However, if $k = 0$, then this is a problem in NP, and if $k = 1$, then this is a problem in coNP. That is, all instances of SAT-UNSAT could be solved either by an NP or a coNP oracle, yet one is sufficient. Furthermore, by presumption of existence of a polynomial time reduction from SAT-UNSAT to $\text{EXACT}_{\mathcal{I}_d}$, one can, in polynomial time, check whether a given instance of SAT-UNSAT is part of a subproblem that is in NP or in coNP (and all instances of SAT-UNSAT are in either one): simply construct the reduction and check whether $k = 0$ or $k = 1$. Thus, presuming such a reduction leads to a fragmentation of the instances of SAT-UNSAT into two categories.

As one can compute the value for $\mathcal{I}_d$ by one call to a SAT-solver we can strengthen the results pertaining $\text{VALUE}_{\mathcal{I}_d}$ as follows. In more details, one can directly transform a knowledge base to a conjunction of its members. Then the result of the functional problem of SAT, on this input, yields either a model or "no", in case the knowledge base is unsatisfiable. In both cases a direct translation back to $\text{VALUE}_{\mathcal{I}_d}$ is immediate: 1 if there is a model and 0 otherwise.

Proposition 2. $\text{VALUE}_{\mathcal{I}_d}$ *is in* FNP.

For the remaining measures, we mostly rely on the reductions and ideas discussed above. We show illustrative proofs for two measures, $\mathcal{I}_{hs}$ and $\mathcal{I}_c$, here, and for the others we refer the interested reader to Appendix A.

We begin with $\mathcal{I}_{hs}$. Recall that this measure is based on hitting sets, i.e., $\mathcal{I}_{hs}(\mathcal{K}) = x$ iff the minimum number of interpretations needed to satisfy all formulas in $\mathcal{K}$ is x.

Proposition 3. $\mathrm{UPPER}_{\mathcal{I}_{hs}}$ *is* NP-*complete,* $\mathrm{LOWER}_{\mathcal{I}_{hs}}$ *is* coNP-*complete,* $\mathrm{EXACT}_{\mathcal{I}_{hs}}$ *is* D_1^p-*complete, and* $\mathrm{VALUE}_{\mathcal{I}_{hs}}$ *is in* $\mathsf{FP}^{NP[\log n]}$.

Proof. For NP-membership of $\mathrm{UPPER}_{\mathcal{I}_{hs}}$, consider the following non-deterministic algorithm for a given $\mathcal{K}$, and value x:

1. Non-deterministically select $H \subseteq \Omega(\mathsf{At})$ with $|H| = x$.
2. For each $\phi \in \mathcal{K}$, if there is no $\omega \in H$ with $\omega \models \phi$ return FALSE.
3. Otherwise return TRUE.

Note that a minimal hitting set H is at most linear in $\mathcal{K}$ (selecting one model for each formula) and that any hitting set H can be extended by adding any interpretations and is still a hitting set. Furthermore, testing $\omega \models \phi$ is polynomial in the length of ϕ. The above guess-and-check algorithm therefore verifies that $\mathcal{K}$ has a hitting set of size x which is an upper bound for a minimal hitting set. NP-completeness follows from the fact that SAT can be reduced to $\mathrm{UPPER}_{\mathcal{I}_{hs}}$ with $x = 0$.

Except for D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_{hs}}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_{hs}}$ by reducing the problem SAT-UNSAT to $\mathrm{EXACT}_{\mathcal{I}_{hs}}$ via the reduction $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, a, \neg a \vee \psi\}$. In particular, we show that (ϕ, ψ) is a positive instance of SAT-UNSAT if and only if $(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi), 1)$ is a positive instance of $\mathrm{EXACT}_{\mathcal{I}_{hs}}$, i.e., if $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$.

1. Let (ϕ, ψ) be a positive instance of SAT-UNSAT. Then $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ is inconsistent, so $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) > 0$. As ϕ is satisfiable there is an interpretation ω with $\omega \models \phi$. As a is a fresh atom we can also safely assume $\omega \models a$. Let ω' be any interpretation with $\omega' \models \neg a$. Then $\omega' \models \neg a \vee \psi$ as well and $\{\omega, \omega'\}$ is a hitting set of $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$, showing $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) \leq 1$ and as $\mathcal{I}_{hs}$ only attains natural numbers we have $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$.

2. We show $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$ implies (ϕ, ψ) is a positive instance of $\mathsf{SAT\text{-}UNSAT}$ by contraposition. So assume (ϕ, ψ) is *not* a positive instance of $\mathsf{SAT\text{-}UNSAT}$. Then consider the following case differentiation:

   (a) ϕ is unsatisfiable: $\mathcal{I}_{hs}(\mathcal{K}) = \infty$ whenever $\mathcal{K}$ contains an unsatisfiable formula (Thimm, 2016b), so $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = \infty$.

   (b) ϕ and ψ are both satisfiable: in this case $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ is consistent and we have $\mathcal{I}_{hs}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 0$.

Thus we polynomially reduced $\mathsf{SAT\text{-}UNSAT}$ to $\mathrm{EXACT}_{\mathcal{I}_{hs}}$, showing that the latter is D_1^p-complete. $\qquad\square$

Next, we illustrate usage of the second main reduction we defined earlier, $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$, on the measure $\mathcal{I}_c$. This measure is based on three-valued interpretations and is defined for a knowledge base $\mathcal{K}$ as the minimum number of atoms assigned to B ("both" or inconsistent) for three-valued interpretations that satisfy $\mathcal{K}$. In the following proof we also show completeness of $\mathrm{VALUE}_{\mathcal{I}_c}$ for complexity class $\mathsf{FP}^{\mathsf{NP}[\log n]}$.

Proposition 4. $\mathrm{UPPER}_{\mathcal{I}_c}$ *is* NP*-complete,* $\mathrm{LOWER}_{\mathcal{I}_c}$ *is* coNP*-complete,* $\mathrm{EXACT}_{\mathcal{I}_c}$ *is* D_1^p*-complete, and* $\mathrm{VALUE}_{\mathcal{I}_c}$ *is* $\mathsf{FP}^{\mathsf{NP}[\log n]}$*-complete.*

Proof. For NP-membership of $\mathrm{UPPER}_{\mathcal{I}_c}$, consider the following non-deterministic algorithm for a given knowledge base $\mathcal{K}$ and value x:

1. Non-deterministically select $v : \mathsf{At} \to \{T, F, B\}$ with $|v^{-1}(B)| = x$.
2. For each $\phi \in \mathcal{K}$, if $v \not\models^3 \phi$ then return FALSE.
3. Otherwise return TRUE.

Note that the size of any 3-valued model is linear in the size of the input and that testing $v \models^3 \phi$ is polynomial in the length of ϕ. The above guess-and-check algorithm therefore verifies that $\mathcal{K}$ has a 3-valued model with x propositions set to B which is an upper bound for a minimal model. NP-completeness follows from the fact that SAT can be reduced to $\mathrm{UPPER}_{\mathcal{I}_c}$ with $x = 0$.

Except D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_c}$ and $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness of $\mathrm{VALUE}_{\mathcal{I}_c}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_c}$ by reducing the problem $\mathsf{SAT\text{-}UNSAT}$ to $\mathrm{EXACT}_{\mathcal{I}_c}$ via the reduction $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, \phi_{\mathrm{copy}}, a, \neg a \vee \psi\}$. We show that (ϕ, ψ) is a positive instance of $\mathsf{SAT\text{-}UNSAT}$ if and only if $(R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi), 1)$ is a positive instance of $\mathrm{EXACT}_{\mathcal{I}_c}$, i.e., if $\mathcal{I}_c(R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$.

1. Let (ϕ, ψ) be a positive instance of SAT-UNSAT. Then there is a three-valued interpretation v_1 over the atoms in ϕ and ϕ_{copy} with $v_1 \models^3$ $\phi \wedge \phi_{\text{copy}}$ and $v_1^{-1}(B) = \emptyset$ as both ϕ and ϕ_{copy} are satisfiable. As ψ is unsatisfiable, there is no v_2 with $v_2 \models^3 \psi$ and $v_2^{-1}(B) = \emptyset$. This already shows that $\mathcal{I}_c(R'_{\text{SAT-UNSAT}}(\phi, \psi)) \geq 1$. But v_3 with $v_3(a) = B$ and $v_3(p) \in \{T, F\}$ arbitrarily for all atoms p appearing in ψ yields $v_3 \models^3 a \wedge (\neg a \vee \psi)$ and $|v_3^{-1}(B)| = 1$. Let v be the combination of v_1 and v_3 (note that the latter two have disjoint domains). Then $v \models^3$ $R'_{\text{SAT-UNSAT}}(\phi, \psi)$ and $|v^{-1}(B)| = 1$, showing $\mathcal{I}_c(R'_{\text{SAT-UNSAT}}(\phi, \psi)) = 1$.

2. We show by contraposition that $\mathcal{I}_c(R'_{\text{SAT-UNSAT}}(\phi, \psi)) = 1$ implies (ϕ, ψ) is a positive instance of SAT-UNSAT. So assume (ϕ, ψ) is *not* a positive instance of SAT-UNSAT. Then consider the following case distinction:

   (a) ϕ is unsatisfiable: then for any v with $v \models^3 R'_{\text{SAT-UNSAT}}(\phi, \psi)$ we have $v(p) = B$ for at least one proposition p appearing in ϕ. As ϕ_{copy} is unsatisfiable as well, we have $v(p') = B$ for another proposition p'. It follows $\mathcal{I}_c(R'_{\text{SAT-UNSAT}}(\phi, \psi)) > 1$.

   (b) ϕ and ψ are both satisfiable: Then there is v with $v \models^3 R'_{\text{SAT-UNSAT}}(\phi, \psi)$ and $v^{-1}(B) = \emptyset$, showing $\mathcal{I}_c(R'_{\text{SAT-UNSAT}}(\phi, \psi)) = 0$ (with $v(a) = T$ and such that its two-valued restriction satisfies ϕ, ϕ_{copy}, and ψ classically).

Thus we polynomially reduced SAT-UNSAT to $\text{EXACT}_{\mathcal{I}_c}$, showing that the latter is D_1^p-complete.

Regarding $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness of $\text{VALUE}_{\mathcal{I}_c}$, let $\phi_s = \{c_1, \ldots, c_n\}$ be an instance over the signature $\mathsf{At} = \{a_1, \ldots, a_m\}$ for MaxSATSize. We now provide a reduction of MaxSATSize to $\text{VALUE}_{\mathcal{I}_c}$. Let $\phi_1, \ldots, \phi_n$ with $\phi_j = \{c_1^j, \ldots, c_n^j\}$ (for $j = 1, \ldots, n$) be copies of ϕ with pairwise different signatures $\mathsf{At}_j = \{a_1^j, \ldots, a_m^j\}$. Let $\alpha_1, \ldots, \alpha_n$ be fresh propositions. Consider the following knowledge base $\mathcal{K}_\phi$.

$$\mathcal{K}_\phi = \{\alpha_1 \wedge \ldots \wedge \alpha_n\} \cup \bigcup_{i=1}^{n} \{\alpha_i \to c_i^1, \ldots, \alpha_i \to c_i^n\}$$

Note that $\mathcal{K}_\phi$ can be constructed in polynomial time. We now claim that the maximum number of jointly satisfiable clauses of ϕ is K if and only if $\mathcal{I}_c(\mathcal{K}_\phi) = n - K$.

Without loss of generality assume that $\{c_1, \ldots, c_K\}$ is jointly satisfiable. Define a three-valued interpretation v as follows. Let $v(\alpha_1) = \ldots = v(\alpha_K) =$

T and $v(\alpha_{K+1}) = \ldots = v(\alpha_n) = B$. Let ω be any (two-valued) model of $\{c_1, \ldots, c_K\}$. For all $i = 1, \ldots, m$ define $v(a_i^1) = \ldots = v(a_i^n) = \omega(a_i)$. Then $v \models^3 \mathcal{K}_\phi$:

1. $v \models^3 \alpha_1 \wedge \ldots \wedge \alpha_n$ as $v(\alpha_i) \in \{B, T\}$ for all $i = 1, \ldots, n$
2. for $i \in \{1, \ldots, K\}$, $v \models^3 \{\alpha_i \to c_i^1, \ldots, \alpha_i \to c_i^n\}$. By definition, $v(\alpha_i) = T$, so in order to have $v \models^3 \{\alpha_i \to c_i^1, \ldots, \alpha_i \to c_i^n\}$ it must hold that $v \models^3 \{c_i^1, \ldots, c_i^n\}$. As $\omega \models c_i$ and each c_i^j is a copy of c_i we have that $v \models^3 c_i^j$ for each $j = 1, \ldots, n$.
3. for $i \in \{K+1, \ldots, n\}$ note that $v(\alpha_i) = B$ so, $v \models^3 \{\alpha_i \to c_i^1, \ldots, \alpha_i \to c_i^n\}$.

We also have $|v^{-1}(B)| = n - K$ which gives $\mathcal{I}_c(\mathcal{K}_\phi) \leq n - K$.

Assume there is $\hat{v}$ with $\hat{v} \models^3 \mathcal{K}_\phi$ and $|\hat{v}^{-1}(B)| < n - K$. In order to have $\hat{v} \models^3 \alpha_1 \wedge \ldots \wedge \alpha_n$ it must hold that $\hat{v}(\alpha_i) \in \{T, B\}$ for $i = 1, \ldots, n$. As $|\hat{v}^{-1}(B)| < n - K$ assume w.l.o.g. $\hat{v}(\alpha_1) = \hat{v}(\alpha_{K+1}) = T$. It follows

$$\hat{v} \models^3 \{c_1^1, \ldots, c_1^n, \ldots, c_{K+1}^1, \ldots, c_{K+1}^n\}$$

As each set $\{c_1^j, \ldots, c_{K+1}^j\}$ for $j = 1, \ldots, n$ is not classically satisfiable (recall that K was the maximal number of jointly satisfiable clauses) there is a $a_{k_j}^j \in \mathsf{At}_j$ with $\hat{v}(a_{k_j}^j) = B$ for all $j = 1, \ldots, n$. It follows $|\hat{v}^{-1}(B)| \geq n$ contradicting $|\hat{v}^{-1}(B)| < n - K$. It follows $\mathcal{I}_c(\mathcal{K}_\phi) = n - K$ and therefore the claim. $\qquad \square$

The proofs of the remaining measures we study in this section ($\mathcal{I}_\eta$, $\mathcal{I}_{dalal}^{hit}$, $\mathcal{I}_{dalal}^{\Sigma}$, $\mathcal{I}_{dalal}^{max}$, $\mathcal{I}_{forget}$, and $\mathcal{I}_{mcsc}$) share certain similarities to the proofs of Propositions 3 and 4, however each requires their own proof, which we show in the Appendix A.

Proposition 5. *It holds that*

- UPPER$_\mathcal{I}$ *is* NP-*complete if* $\mathcal{I} \in \{\mathcal{I}_\eta, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{max}, \mathcal{I}_{forget}, \mathcal{I}_{mcsc}\}$;

- LOWER$_\mathcal{I}$ *is* coNP-*complete if* $\mathcal{I} \in \{\mathcal{I}_\eta, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{max}, \mathcal{I}_{forget}, \mathcal{I}_{mcsc}\}$;

- EXACT$_\mathcal{I}$ *is* D_1^p-*complete if* $\mathcal{I} \in \{\mathcal{I}_\eta, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{max}, \mathcal{I}_{forget}, \mathcal{I}_{mcsc}\}$;

- VALUE$_\mathcal{I}$ *is* $\mathsf{FP}^{NP[\log n]}$-*complete if* $\mathcal{I} \in \{\mathcal{I}_{dalal}^{hit}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{forget}\}$; *and*

- VALUE$_\mathcal{I}$ *is in* $\mathsf{FP}^{NP[\log n]}$ *if* $\mathcal{I} \in \{\mathcal{I}_\eta, \mathcal{I}_{dalal}^{max}, \mathcal{I}_{mcsc}\}$.

We conclude our investigation of problems on the first level of the polynomial hierarchy by pointing out an interesting relationship. As for $\text{VALUE}_{\mathcal{I}_{hs}}$, we are currently unable to provide a $\mathsf{FP}^{\mathsf{NP}[\log n]}$-completeness proof. However, the problem $\text{VALUE}_{\mathcal{I}_{hs}}$ has a close relationship with the problem Chromatic-Number defined via

ChromaticNumber   **Input**:   an undirected graph $G = (V, E)$
                                       Output: The chromatic number $\gamma(G)$ of G

Recall that the chromatic number $\gamma(G)$ of a graph G is the minimum number of colours needed to colour all vertices of G so that no two adjacent vertices have the same colour. We have the following relationship between the two problems.

Proposition 6. *ChromaticNumber can be polynomially reduced to* $\text{VALUE}_{\mathcal{I}_{hs}}$.

However, it seems not to be known whether ChromaticNumber is $\mathsf{FP}^{\mathsf{NP}[\log n]}$-complete (which would also show $\mathsf{FP}^{\mathsf{NP}[\log n]}$-completeness for $\text{VALUE}_{\mathcal{I}_{hs}}$).

4.2. Problems on the second level of the polynomial hierarchy

We now turn to inconsistency measures, which involve problems on the second level of the polynomial hierarchy. We first recall a result regarding $\mathcal{I}_{mv}$ from (Xiao and Ma, 2012) which is given without proof.

Proposition 7. $\text{EXACT}_{\mathcal{I}_{mv}}$ *is* D_2^p-*complete,* $\text{UPPER}_{\mathcal{I}_{mv}}$ *is* Π_2^p-*complete,* $\text{LOWER}_{\mathcal{I}_{mv}}$ *is* Σ_2^p-*complete, and* $\text{VALUE}_{\mathcal{I}_{mv}}$ *is in* $\mathsf{FP}^{\Sigma_2^p[\log n]}$.

We now continue with two novel results regarding the inconsistency measures $\mathcal{I}_{nc}$ and $\mathcal{I}_p$. For both we provide direct proofs of Σ_2^p-completeness for the problem $\text{LOWER}_{\mathcal{I}}$ and utilize again Lemmas 4 and 6 to gain insights on the remaining problems. Intuitively, the increase in terms of complexity of measures in this section, compared to measures discussed in previous Section 4.1, is due to the fact that to verify a lower bound we non-deterministically guess a witness that guarantees the bound, but checking the witness itself is a coNP-hard problem.

We give an overview on the problems used in the proofs. In order to show hardness results we will make use of several problems involving *Quantified Boolean Formulas* (QBFs). In general, a quantified Boolean formula ϕ (in prenex normal form) is an expression of the form

$$\phi = Q_1 X_1 \ldots Q_n X_n \psi$$

where $Q_1, \ldots, Q_n \in \{\forall, \exists\}$, $X_1, \ldots, X_n$ are pairwise disjoint sets of propositions and ψ is a propositional formula over the signature $X_1 \cup \ldots \cup X_n$. Truth of quantified Boolean formulas is inductively defined as follows:

- $\phi = \forall X_1 Q_2 X_2 \ldots Q_n X_n \psi$ is *true* if for all $\omega \in \Omega(X_1)$, $Q_2 X_2 \ldots Q_n X_n \psi[\omega]$ is true.

- $\phi = \exists X_1 Q_2 X_2 \ldots Q_n X_n \psi$ is *true* if there is $\omega \in \Omega(X_1)$ such that $Q_2 X_2 \ldots Q_n X_n \psi[\omega]$ is true.

- $\phi = \psi$ is true if $\psi \equiv \top$

A QBF ϕ as above is *alternating* if universal and existential quantifiers alternate. The following problems $\exists \mathsf{QBF}_i$ and $\forall \mathsf{QBF}_i$ are the canonical Σ_i^p- and Π_i^p-complete problems ($i \in \mathbb{N}$), respectively.

$\exists \mathsf{QBF}_i$ **Input**:   an alternating QBF $\phi = \exists X_1 \forall X_2 \ldots Q_i X_i \psi$
                      Output: TRUE iff ϕ is true

$\forall \mathsf{QBF}_i$ **Input**:   an alternating QBF $\phi = \forall X_1 \exists X_2 \ldots Q_i X_i \psi$
                      Output: TRUE iff ϕ is true

The problem families $\exists \mathsf{QBF}_i$ and $\forall \mathsf{QBF}_i$ generalise the problems SAT and UNSAT to higher levels of the polynomial hierarchy. In particular, note that $\exists \mathsf{QBF}_1$ is equivalent to SAT as it asks whether a QBF $\phi = \exists X \psi$ is true, i.e., whether ψ is satisfiable. Similarly, $\forall \mathsf{QBF}_1$ is equivalent to UNSAT as it asks whether a QBF $\phi = \forall X \psi$ is true, i.e., whether $\neg \psi$ is unsatisfiable.

In order to show D_2^p-completeness, we can also generalise the problem SAT-UNSAT in a similar way to higher levels of the polynomial hierarchy. Here, the problem $\forall \exists \mathsf{QBF}_i$ defined as follows is the canonical D_i^p-complete problem.

$\forall \exists \mathsf{QBF}_i$ **Input**:   a pair (ϕ, ϕ') of QBFs with
                               $\phi = \exists X_1 \forall X_2 \ldots Q_i X_i \psi$ and $\phi' = \forall X_1' \exists X_2' \ldots Q_i' X_i' \psi'$
                     Output: TRUE iff ϕ and ϕ' are true

Note as well that $\forall \exists \mathsf{QBF}_1$ is equivalent to SAT-UNSAT.

Our results are as follows.

Proposition 8. $\mathrm{LOWER}_{\mathcal{I}_{nc}}$ *is* Σ_2^p-*complete,* $\mathrm{UPPER}_{\mathcal{I}_{nc}}$ *is* Π_2^p-*complete,* $\mathrm{EXACT}_{\mathcal{I}_{nc}}$ *is in* D_2^p, *and* $\mathrm{VALUE}_{\mathcal{I}_{nc}}$ *is in* $\mathsf{FP}^{\Sigma_2^p[\log n]}$.

Proof. We first consider $\text{LOWER}_{\mathcal{I}_{nc}}$. Observe that, if $\mathcal{K} \models \bot$, we have

$$
\begin{aligned}
\mathcal{I}_{nc}(\mathcal{K}) =&|\mathcal{K}| - \max\{n \mid \forall \mathcal{K}' \subseteq \mathcal{K} : |\mathcal{K}'| = n \Rightarrow \mathcal{K}' \not\models \bot\} \\
=&|\mathcal{K}| - \min\{m \mid \exists \mathcal{K}' \subseteq \mathcal{K} : |\mathcal{K}'| = m \wedge \mathcal{K}' \models \bot\} + 1
\end{aligned}
$$

For the membership result, first check whether $\mathcal{K} \models \bot$ (via one NP oracle call). Then, we non-deterministically guess a set $\mathcal{K}' \subseteq \mathcal{K}$ with $|\mathcal{K}'| = k$ and ask an NP-oracle whether $\mathcal{K}'$ is inconsistent. If $\mathcal{K}'$ is inconsistent then k is an upper bound for

$$
\min\{m \mid \exists \mathcal{K}' \subseteq \mathcal{K} : |\mathcal{K}'| = m \wedge \mathcal{K}' \models \bot\}
$$

and thus $|\mathcal{K}| - k + 1$ is a lower bound for $\mathcal{I}_{nc}(\mathcal{K})$.

Regarding hardness, we provide a reduction from $\exists\mathsf{QBF}_2$. Let $\phi = \exists X \forall Y \psi$ be an instance of $\exists\mathsf{QBF}_2$. Let $X = \{x_1, \ldots, x_n\}$. Construct an instance of $\text{LOWER}_{\mathcal{I}_{nc}}$ as follows.

$$
\begin{aligned}
\chi_i = p_i \wedge (d \to x_i) \quad &\overline{\chi_i} = p_i \wedge (d \to \neg x_i) \\
\chi = (\bigwedge_{1 \le i \le n} p_i) &\to (d \wedge \neg\psi)
\end{aligned}
$$

The atoms d and p_i are fresh atoms not occurring in ψ. Then let the knowledge base $\mathcal{K} = \bigcup_{1 \le i \le n}\{\chi_i, \overline{\chi_i}\} \cup \{\chi\}$. Note that $\mathcal{K} \models \bot$. Further set bound $x = |\mathcal{K}| - (n+1) + 1 = 2 \cdot n + 1 - n = n + 1$. Knowledge base $\mathcal{K}$ can be constructed in polynomial time. We claim that ϕ is true iff $\mathcal{I}_{nc}(\mathcal{K}) \ge n + 1$. We start with the following observation: it holds that any $\mathcal{K}' \subseteq \mathcal{K}$ is satisfiable if (i) $\chi \notin \mathcal{K}'$, or (ii) $|\mathcal{K}'| < n + 1$. If $\chi \notin \mathcal{K}'$ then there is a model of $\mathcal{K}'$ assigning d to false. If $|\mathcal{K}'| < n + 1$ then $\mathcal{K}'$ is satisfiable, since either $\chi \notin \mathcal{K}'$ or for one i with $1 \le i \le n$ neither $\chi_i \in \mathcal{K}'$ nor $\overline{\chi_i} \in \mathcal{K}'$ (p_i can be assigned to false). It holds that (note that $|\mathcal{K}| = 2 \cdot n + 1$)

$$
\begin{aligned}
&\mathcal{I}_{nc}(\mathcal{K}) \ge n + 1 = 2 \cdot n + 1 - (n+1) + 1 = |\mathcal{K}| - (n+1) + 1 \\
&\text{iff } \exists \mathcal{K}' \subseteq \mathcal{K} \text{ s.t. } \mathcal{K}' \models \bot \text{ and } |\mathcal{K}'| = n + 1 \\
&\text{iff } \exists \mathcal{K}' \subseteq \mathcal{K} \text{ s.t.} \\
&\quad \mathcal{K}' \models \bot, \chi \in \mathcal{K}', \text{ and } |\mathcal{K}' \cap \{\chi_i, \overline{\chi_i}\}| = 1 \; \forall i \; 1 \le i \le n \\
(*) \; &\text{iff } \exists \omega \text{ defined on } X \text{ s.t. } \neg\psi[\omega] \models \bot \\
&\text{iff } \exists \omega \text{ defined on } X \text{ s.t. } \top \models \psi[\omega] \\
&\text{iff } \phi \text{ is true.}
\end{aligned}
$$

For ($*$), intuitively, a "simulation" of a truth value assignment of an interpretation is achieved by "choosing" either χ_i or $\overline{\chi_i}$.

All other results follow from Lemmas 4 and 6. $\qquad\square$

The proof of the following proposition regarding $\mathcal{I}_p$ is similar as the one above and can be found in Appendix A.

Proposition 9. UPPER$_{\mathcal{I}_p}$ *is* Π_2^p-*complete,* LOWER$_{\mathcal{I}_p}$ *is* Σ_2^p-*complete,* EXACT$_{\mathcal{I}_p}$ *is* D_2^p-*complete., and* VALUE$_{\mathcal{I}_p}$ *is in* $\mathsf{FP}^{\Sigma_2^p[\log n]}$.

4.3. Problems on the third level of the polynomial hierarchy

We now look into the computational complexity of problems involving the measure $\mathcal{I}_{CC}$. Recall that computing $\mathcal{I}_{CC}(\mathcal{K})$ for a knowledge base $\mathcal{K}$ amounts to considering sets $\{K_1, \ldots, K_n\}$ such that all K_i are pairwise disjoint inconsistent subsets of $\mathcal{K}$ and $\mathsf{MI}(K_1 \cup \ldots \cup K_n)$ is the union of all $\mathsf{MI}(K_i)$, $i = 1, \ldots, n$. Such a set is called conditional independent MUS partition of $\mathcal{K}$ and $\mathcal{I}_{CC}(\mathcal{K})$ is defined to be the cardinality of a maximal one.

Let us first consider the following decision problem DECMUSPART about conditional independent MUS partitions:

DECMUSPART   **Input**: $\mathcal{K} \in \mathbb{K}$, $\{K_1, \ldots, K_n\}$
                                 Output: TRUE iff $\{K_1, \ldots, K_n\}$ is a conditional independent MUS partition of $\mathcal{K}$

We show now that DECMUSPART is Π_2^p-complete.

Proposition 10. DECMUSPART *is* Π_2^p-*complete.*

Proof. For membership, we show that the complementary problem CODEC-MUSPART defined as

CODECMUSPART   **Input**: $\mathcal{K} \in \mathbb{K}$, $\{K_1, \ldots, K_n\}$
                                     Output: TRUE iff $\{K_1, \ldots, K_n\}$ is not a conditional independent MUS partition of $\mathcal{K}$

is in $\Sigma_2^p = \mathsf{NP}^{\mathsf{NP}}$. Then it automatically follows that DECMUSPART is in Π_2^p. To show that CODECMUSPART is in $\mathsf{NP}^{\mathsf{NP}}$ consider the following algorithm

1. Non-deterministically guess an interpretation ω, a set $M = \{\alpha_1, \ldots, \alpha_m\} \subseteq \mathcal{K}$, and interpretations $\omega_1, \ldots, \omega_m$
2. If $K_1, \ldots, K_n$ are not pairwise disjoint return TRUE

3. If $K_1 \cup \ldots \cup K_n \nsubseteq \mathcal{K}$ return TRUE
4. If $\omega \models K_i$ for some $i \in \{1, \ldots, n\}$ return TRUE
5. If there are no $i, j \in \{1, \ldots, n\}$, $i \neq j$, with $M \cap K_i \neq \emptyset$ and $M \cap K_j \neq \emptyset$ return FALSE
6. Use NP-oracle to decide $M \models \bot$; if $M \nvDash \bot$ return FALSE
7. for $i = 1, \ldots, m$, if $\omega_i \nvDash M \setminus \{\alpha_i\}$ return FALSE
8. Return TRUE

Step 2 and step 3 check whether $K_1, \ldots, K_n$ are pairwise disjoint subsets of $\mathcal{K}$. If this is not the case they cannot be a conditional independent MUS partition. Step 4 checks whether any K_i is consistent, which also shows that $\{K_1, \ldots, K_n\}$ cannot be a conditional independent MUS partition. Steps 5–7 check whether there is a minimal inconsistent subset M that intersects two different K_i, K_j. In particular, in step 5 we check whether the guessed M intersects with two different K_i, K_j. In step 6 we check inconsistency of M and in step 7 minimality (all subsets with one fewer element are consistent). Note that the algorithm has polynomial runtime.

In order to show Π_2^p-hardness, we consider a reduction from $\forall\mathsf{QBF}_2$ (see previous section). Let $\phi = \forall X \exists Y \psi$ be an instance of $\forall\mathsf{QBF}_2$. Let $X = \{x_1, \ldots, x_m\}$ and c a fresh propositional variable. Then define

$$K_1 = \{x_1, \neg x_1\}$$
$$\vdots$$
$$K_m = \{x_m, \neg x_m\}$$
$$K_{m+1} = \{\psi \wedge c, \neg c\}$$

We claim that $\phi = \forall X \exists Y \psi$ evaluates to true if and only if $\{K_1, \ldots, K_{m+1}\}$ is a conditional independent MUS partition of $K_1 \cup \ldots \cup K_{m+1}$. First, observe that each set $K_1, \ldots, K_{m+1}$ is a minimal inconsistent set. So, $\{K_1, \ldots, K_{m+1}\}$ can only be a conditional independent MUS partition if there is no $M \in \mathsf{MI}(K_1 \cup \ldots \cup K_{m+1})$ with $M \neq K_1, \ldots, M \neq K_{m+1}$. Assume there is such an M. Then $\psi \wedge c \in M$ as every minimal inconsistent subset of $K_1 \cup \ldots \cup K_m$ is one of the $K_1, \ldots, K_m$ itself. So

$$M = \{(\neg)x_{i_1}, \ldots, (\neg)x_{i_k}, \psi \wedge c\}$$

for some $i_1, \ldots, i_k \subseteq \{1, \ldots, m\}$ with $i_j \neq i_{j'}$ for $j \neq j'$. As c cannot be responsible for the inconsistency in M, it follows that there is a configuration of

X (prescribed by $(\neg)x_{i_1}, \ldots, (\neg)x_{i_k}$) such that ψ is false (for all configurations of Y). So if $\{K_1, \ldots, K_{m+1}\}$ is not a conditional independent MUS partition then $\phi = \forall X \exists Y \psi$ evaluates to false. On the other hand, if $\{K_1, \ldots, K_{m+1}\}$ is a conditional independent MUS partition then $\phi = \forall X \exists Y \psi$ evaluates to true. $\square$

Having established the exact complexity class of DECMUSPART on the second level of the polynomial hierarchy, we can easily establish that $\text{LOWER}_{\mathcal{I}_{CC}}$ resides not higher than on the third level of the polynomial hierarchy.

Proposition 11. $\text{LOWER}_{\mathcal{I}_{CC}}$ *is in* Σ_3^p.

The Π_2^p-completeness of DECMUSPART is a strong evidence that $\text{LOWER}_{\mathcal{I}_{CC}}$ cannot reside on the second level of the polynomial hierarchy itself. However, we did not find any Σ_3^p-hardness proof for $\text{LOWER}_{\mathcal{I}_{CC}}$, yet. Nonetheless, using the membership result above we can also provide upper bounds for the other computational problems, utilising similar reasoning as in Lemmas 4 and 6.

Corollary 3. $\text{UPPER}_{\mathcal{I}_{CC}}$ *is in* Π_3^p, $\text{EXACT}_{\mathcal{I}_{CC}}$ *is in* D_3^p, *and* $\text{VALUE}_{\mathcal{I}_{CC}}$ *is in* $\mathsf{FP}^{\Sigma_3^p[\log n]}$.

4.4. Problems beyond the third level of the polynomial hierarchy

In this section we study the complexity of the remaining measures $\mathcal{I}_{\mathsf{MI}}$, $\mathcal{I}_{\mathsf{MI}^c}$, $\mathcal{I}_{mc}$, and $\mathcal{I}_{\mathsf{is}}$. The main results in this section are that these measures contain (sub)problems whose counting complexity is higher than for propositional model counting. In particular, for $\mathcal{I}_{\mathsf{MI}}$, $\mathcal{I}_{mc}$, and, $\mathcal{I}_{\mathsf{is}}$, we show completeness for CNP for the decision problems with lower and upper bounds and hardness for $\mathsf{C_=NP}$ checking the exact bound. Further, we show $\#\cdot\mathsf{coNP}$ completeness, under subtractive reductions (see Section 2 for the definition) for these measures for the problem of computing the exact value.

Since $\#\cdot\mathsf{coNP}$ is a complexity class containing counting problems, we consider the problems of counting the number of (a) minimal inconsistent subsets, (b) maximal consistent subsets and self-contradictory formulas, and (c) independent sets from the set of minimal inconsistent subsets. Problem (a) is equal to the problem $\text{VALUE}_{\mathcal{I}_{\mathsf{MI}}}$. Problem (b) differs from $\text{VALUE}_{\mathcal{I}_{mc}}$ by being off by one ($\text{VALUE}_{\mathcal{I}_{mc}}$ is equal to the number of maximal consistent subsets plus the number of self-contradictory formulas minus one). The value sought in problem $\text{VALUE}_{\mathcal{I}_{\mathsf{is}}}$ is based on a logarithm of the number of

43

independent sets; we omit in our investigation of $\textsc{Value}_{\mathcal{I}_{\mathsf{is}}}$ this logarithm. The reason for omitting the logarithm is that we can then directly apply counting complexity classes (which rely on a non-negative integer as output). We think this is not an obstacle in complexity classification: one can directly obtain the result with negligible computational cost. The problems are formalized in our usual notation as follows.

#MCAndContradictory	**Input**:	$\mathcal{K} \in \mathbb{K}$
	Output:	$|\mathsf{MC}(\mathcal{K})| + |\{\phi \in \mathcal{K} \mid \phi \models \bot\}|$
#is	**Input**:	$\mathcal{K} \in \mathbb{K}$
	Output:	$|\{M \subseteq \mathsf{MI}(\mathcal{K}) \mid M \text{ is independent}\}|$

For the measure $\mathcal{I}_{\mathsf{MI}^{\mathsf{c}}}$, we show hardness for all three decision problems (for CNP and $\mathsf{C_=NP}$), and show membership for $\textsc{Value}_{\mathcal{I}_{\mathsf{MI}^{\mathsf{c}}}}$ for class $FP^{\#\cdot\mathsf{coNP}}$. That is, the value of this measure can be computed via a deterministic polynomial time algorithm that can access a functional $\#\cdot\mathsf{coNP}$ oracle.

Briefly put, these complexity results indicate that problems under $\mathcal{I}_{\mathsf{MI}}$, $\mathcal{I}_{\mathsf{MI}^{\mathsf{c}}}$, $\mathcal{I}_{mc}$, and $\mathcal{I}_{\mathsf{is}}$ exhibit considerably high complexity, in particular, if compared to all the other measures studied in this paper. An intuitive explanation is that all these measures are based on counting structures that are themselves hard (e.g. D_1^p-complete) to check whether a structure is to be counted or not, and, in addition, that the number of attainable values may be exponential in the size of the knowledge base (see also (Thimm, 2016a)).

The main problems we use in this section to reduce from are those related to counting models of an open quantified Boolean formula. More concretely, we use problems with a given QBF of the form $\chi = \forall Y \phi(X, Y)$ with X and Y sets of Boolean variables. That is, we have a universally quantified part (Y variables) and an open part (X variables). A model of χ is an assignment to X s.t. $\forall Y \phi'(Y)$ evaluates to true, with $\phi'(Y)$ being the formula where all occurrences of variables in X have been replaced by the assignment to X (i.e. if x is true replaced by $\top$ and if x is false replaced by $\bot$). We denote, like in the unquantified case, the set of all models of χ by $\mathsf{Mod}(\chi)$.

The three main problems we make use of are then defined, for a given $\chi = \forall Y \phi(X, Y)$, as (i) the counting problem of computing $|\mathsf{Mod}(\chi)|$, (ii) determining whether $|\mathsf{Mod}(\chi)| \geq k$ holds for a given integer k, and (iii) determining whether $|\mathsf{Mod}(\chi)| = k$ for a given integer k. Problem (i) is

#·coNP-complete, problem (ii) is CNP-complete, and problem (iii) is $C_=NP$-complete.

# Π_1SAT	**Input**:	a QBF $\chi = \forall Y \phi(X, Y)$
	Output:	$|\mathsf{Mod}(\chi)|$
$\#_{\geq}\Pi_1$SAT	**Input**:	a QBF $\chi = \forall Y \phi(X, Y)$, and an integer $k \geq 0$
	Output:	TRUE iff $|\mathsf{Mod}(\chi)| \geq k$
$\#_=\Pi_1$SAT	**Input**:	a QBF $\chi = \forall Y \phi(X, Y)$, and an integer $k \geq 0$
	Output:	TRUE iff $|\mathsf{Mod}(\chi)| = k$

In the following subsections, we introduce reductions s.t. the models of an open QBF χ are in a certain relation to the constructed instance for an inconsistency measure, which we then use for showing (several) results.

For membership results, we make use of relations between (counting) complexity classes that have been established. It holds that $\#\cdot\mathsf{coNP} = \#\cdot\Delta_2^p$ (Hemaspaandra and Vollmer, 1995, Theorem 1.5), which eases proofs of membership, in particular since $\mathsf{D}^p \subseteq \Delta_2^p$, and, in turn, $\#\cdot\mathsf{coNP}$ contains counting problems where the complexity of verifying whether to count or not can be in D^p. A similar useful result for us is that $\mathsf{CNP} = \mathsf{C}\mathcal{B}(\mathsf{NP})$ (Wagner, 1986, Theorem 4), with $\mathcal{B}(\mathsf{NP})$ the Boolean closure of NP, which, most importantly for our work here, implies that $\mathsf{D}^p \subseteq \mathcal{B}(\mathsf{NP})$ and $\mathsf{CNP} = \mathsf{CD}^p$.

For problems EXACT$_\mathcal{I}$, for $\mathcal{I} \in \{\mathcal{I}_{\mathsf{MI}}, \mathcal{I}_{mc}, \mathcal{I}_{\mathsf{is}}, \mathcal{I}_{\mathsf{MI}}c\}$, we show hardness for $\mathsf{C}_=\mathsf{NP}$. The reason for (only) showing hardness for these cases is that similar results, as for $\#\cdot\mathsf{coNP}$ and CNP from the preceding paragraph, have not been established, to our knowledge, for $\mathsf{C}_=\mathsf{NP}$. That is, it is not clear whether $\mathsf{C}_=\mathsf{NP} = \mathsf{C}_=\mathcal{B}(\mathsf{NP})$, for instance. Nevertheless, it holds that EXACT$_\mathcal{I}$ is in a class that is defined via the conjunction of two problems in CNP (one asking whether the value is an upper, the other whether the value is a lower bound) for the measures $\mathcal{I}_{\mathsf{MI}}$, $\mathcal{I}_{mc}$, and $\mathcal{I}_{\mathsf{is}}$, where we have established membership for CNP for the corresponding problems asking for upper, resp. lower, bounds. Such classes, to our knowledge, have not been thoroughly investigated, however.

To prove our results, we make use of auxiliary lemmas for the measures that (directly) connect them to models of quantified Boolean formulas. We illustrate the proof for one measure in detail ($\mathcal{I}_{\mathsf{MI}}$) and show the proofs for the remaining measures in the appendix.

4.4.1. The measure $\mathcal{I}_{\mathsf{MI}}$

Recall that $\mathcal{I}_{\mathsf{MI}}(\mathcal{K}) = |\mathsf{MI}(\mathcal{K})|$, i.e., $\mathcal{I}_{\mathsf{MI}}(\mathcal{K})$ is the number of minimal inconsistent subsets of $\mathcal{K}$. Although, conceptually, this measure is one of the simpler ones, computationally it is quite demanding as we will see below. We start with defining a translation from a quantified Boolean formula to a knowledge base such that the models of the formula and the minimal inconsistent subsets correspond.

$$R_{\#\mathsf{MI}}(\forall Y \phi(X,Y)) = \{\psi\} \cup \bigcup_{1 \le i \le |X|} \{\phi_i, \overline{\phi_i}\}$$

$$\phi_i = p_i \wedge (\bigwedge_{j \ne i} p_j \to x_i)$$

$$\overline{\phi_i} = p_i \wedge (\bigwedge_{j \ne i} p_j \to \neg x_i)$$

$$\psi = \bigwedge_{1 \le i \le |X|} p_i \to \neg\phi(X,Y)$$

Formally, we show the correspondence we require for subsequent complexity proofs in the following lemma.

Lemma 7. *Let $\chi = \forall Y \phi(X,Y)$ be an open QBF with X and Y sets of variables. For $\mathcal{K} = R_{\#\mathsf{MI}}(\forall Y \phi(X,Y))$ it holds that $|\mathsf{Mod}(\chi)| = |\mathsf{MI}(\mathcal{K})| - |X|$.*

Proof. Let $P_1 = \bigcup_{1 \le i \le n} \{\phi_i\}$ and $\overline{P_1} = \bigcup_{1 \le i \le n} \{\overline{\phi_i}\}$. Further, let $P_2 = \{\psi\} \cup P_1 \cup \overline{P_1}$. We now claim that the number of truth assignments over X that satisfy χ is $|\mathsf{MI}(P_2)| - |\mathsf{MI}(P_1 \cup \overline{P_1})|$. Note that, due to monotonicity of MI, it holds that, if $\mathcal{K}_1 \subseteq \mathcal{K}_2$ then $\mathsf{MI}(\mathcal{K}_1) \subseteq \mathsf{MI}(\mathcal{K}_2)$.

Let $M \in \mathsf{MI}(P_2)$. It holds that for $1 \le i \le n$ that $\phi_i \in M$ or $\overline{\phi_i} \in M$. Suppose the contrary, i.e. there exists an i s.t. neither ϕ_i nor $\overline{\phi_i}$ is in M. Then a truth assignment assigning all p_j with $j \ne i$ to true and p_i to false satisfies all formulas in M. This is a contradiction to $M \in \mathsf{MI}(P_2)$.

Now assume that $M \in \mathsf{MI}(P_2)$ s.t. $\exists i$ with both $\phi_i \in M$ and $\overline{\phi_i} \in M$. Then $\psi \notin M$, since by observations above we have for each $1 \le j \le n$ with $j \ne i$ that $\phi_j \in M$ or $\overline{\phi_j} \in M$ and both for formulas for i. These formulas together are inconsistent, and thus adding a further formula (such as ψ) would not be minimal anymore. This means if $M \in \mathsf{MI}(P_2)$ s.t. $\exists i$ with both $\phi_i \in M$ and $\overline{\phi_i} \in M$, then $M \in \mathsf{MI}(P_1 \cup \overline{P_1})$. Further, if $M \in \mathsf{MI}(P_2)$ and $\nexists i$

with both $\phi_i \in M$ and $\overline{\phi_i} \in M$, then $\psi \in M$ (if one of ϕ_i or $\overline{\phi_i}$ is missing and also ψ is not present, then the set of formulas is satisfiable).

Let $\mathcal{K}^* \subseteq 2^{\mathcal{K}}$ be the set of subsets of $\mathcal{K}$ s.t. each $\mathcal{K}' \in \mathcal{K}^*$ contains for each i with $1 \leq i \leq n$ exactly one of ϕ_i or $\overline{\phi_i}$ but not both, and in addition $\mathcal{K}'$ contains ψ. We define a bijection f from $\mathcal{K}' \in \mathcal{K}^*$ to an interpretation over X by

$$f(\mathcal{K}')(x_i) = \begin{cases} \text{true if } \phi_i \in \mathcal{K}' \\ \text{false if } \overline{\phi_i} \in \mathcal{K}'. \end{cases}$$

For $f(M) = \omega_M$ and by the observations above it holds that

$$M \in (\mathsf{MI}(P_2) \setminus \mathsf{MI}(P_1 \cup \overline{P_1}))$$
$$\text{iff } M \in \mathsf{MI}(P_2), \psi \in M, \text{ and}$$
$$|M \cap \{\phi_i, \overline{\phi_i}\}| = 1 \; \forall i \text{ with } 1 \leq i \leq n$$
$$\text{iff } \top \models \phi(X, Y)[\omega_M]$$
$$\text{iff } w_M \text{ satisfies } \forall Y \phi(X, Y).$$

Thus, there is a bijection between $\mathsf{MI}(P_2 \setminus (P_1 \cup \overline{P_1}))$ and the set of satisfying assignments defined on X of $\chi(X)$. Finally, $|\mathsf{MI}(P_1 \cup \overline{P_1})| = |X|$ (for each $1 \leq i \leq |X|$ there is one member in this set with both ϕ_i and $\overline{\phi_i}$ part of the member). $\square$

Equipped with the preceding lemma, we can straightforwardly use the reduction to show completeness results for $\mathrm{LOWER}_{\mathcal{I}_{\mathsf{MI}}}$ and $\mathrm{UPPER}_{\mathcal{I}_{\mathsf{MI}}}$.

Proposition 12. $\mathrm{LOWER}_{\mathcal{I}_{\mathsf{MI}}}$ and $\mathrm{UPPER}_{\mathcal{I}_{\mathsf{MI}}}$ are $\mathsf{C}\mathsf{NP}$-complete, $\mathrm{EXACT}_{\mathcal{I}_{\mathsf{MI}}}$ is $\mathsf{C}_=\mathsf{NP}$-hard.

For the functional problem $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}}}$, we show completeness for the counting class $\#\cdot\mathsf{coNP}$. By utilizing Lemma 7, we can show hardness under subtractive reductions.

Proposition 13. $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}}}$ is $\#\cdot\mathsf{coNP}$-complete under subtractive reductions.

4.4.2. The measure $\mathcal{I}_{\mathsf{MI}^c}$

Determining $\mathcal{I}_{\mathsf{MI}^c}(\mathcal{K})$ involves summing up the reciprocals of the cardinalities of all minimal inconsistent subset of $\mathcal{K}$. We show that all three associated decision problems are CNP ($\mathsf{C}_=\mathsf{NP}$) hard, and that the actual value can be

computed in $FP^{\#\cdot\text{coNP}}$, i.e., via a deterministic polynomial time algorithm that can access a functional $\#\cdot\text{coNP}$ oracle. The last result implies the same membership for the decision problems.

Proposition 14. $\text{LOWER}_{\mathcal{I}_{\text{MIC}}}$ and $\text{UPPER}_{\mathcal{I}_{\text{MIC}}}$ are both $\textsf{C}\textsf{NP}$-hard problems, and $\text{EXACT}_{\mathcal{I}_{\text{MIC}}}$ is a $\textsf{C}_{=}\textsf{NP}$ hard problem.

For problem $\text{VALUE}_{\mathcal{I}_{\text{MIC}}}$, we show that an instance of this problem can be solved via an algorithm that can access a $\#\cdot\text{coNP}$ oracle, which can be used to compute for each possible size of an MI the number of such minimal inconsistent subsets of a knowledge base.

Proposition 15. $\text{VALUE}_{\mathcal{I}_{\text{MIC}}}$ is in $FP^{\#\cdot\text{coNP}}$.

4.4.3. The measure $\mathcal{I}_{mc}$

We move on to the complexity of $\mathcal{I}_{mc}$ which is defined as the sum of the number of maximal consistent subsets and the number of inconsistent formulas of a knowledge base (minus one). So this measure has two components, which we analyse both separately and combined.

Similarly as for $\mathcal{I}_{\text{MI}}$, we begin with defining a translation of an open quantified Boolean formula to a knowledge base, so that the models of that formula are in a correspondence to the maximal consistent subsets of the constructed knowledge base.

$$R_{\#\text{MC}}(\forall Y \phi(X,Y)) = \{\psi\} \cup \bigcup_{1 \le i \le |X|} \{\phi_i, \overline{\phi_i}\}$$
$$\phi_i = p \to x_i$$
$$\overline{\phi_i} = p \to \neg x_i$$
$$\psi = p \land \neg \phi(X,Y)$$

In the next formal result we show that the interpretations that are not models of χ correspond, with an offset of one, to maximal consistent subsets of $\mathcal{K}$.

Lemma 8. Let $\chi = \forall Y \phi(X,Y)$ be an open QBF with X and Y sets of variables. For $\mathcal{K} = R_{\#\text{MC}}(\forall Y \phi(X,Y))$ it holds that $|\text{Mod}(\chi)| = 2^{|X|} - |\text{MC}(\mathcal{K})| + 1$.

Based on Lemma 8, we show the following completeness and hardness results for the decision problems relating to $\mathcal{I}_{mc}$.

Proposition 16. LOWER$_{\mathcal{I}_{mc}}$ *and* UPPER$_{\mathcal{I}_{mc}}$ *are* C*NP-complete,* EXACT$_{\mathcal{I}_{mc}}$ *is* C$_=$*NP-hard.*

For VALUE$_{\mathcal{I}_{mc}}$, recall that this measure is equal to the sum of the number of maximal consistent subsets and the number of inconsistent formulas of a knowledge base minus one. We analyze the (counting) complexity of the (sub)problems that ask, for a given knowledge base, (i) the number of maximal consistent subsets, (ii) the number of inconsistent formulas, and (iii) the sum of both values. We begin with (i), and introduce an auxiliary problem which counts the number of subset-maximal models of a propositional formula wrt. the propositions assigned to true. For that, we define the ordering $<_t$ over interpretations by $\omega <_t \omega'$ iff $\{p \mid \omega(p) = \text{true}\} \subset \{p \mid \omega'(p) = \text{true}\}$.

#MaxModels	**Input**:	formula ϕ in CNF
	Output:	$|\{\omega \models \phi \mid \nexists \omega' \text{ s.t. } \omega <_t \omega'\}|$

Durand et al. (2005) have shown (Theorem 5.1) that the problem #CIRCUMSCRIPTION—which is basically the dual of the problem #MaxModels as it counts the subset-minimal models—is #·coNP-complete (via subtractive reductions). We provide a corollary showing that also counting the number of subset-maximal models is #·coNP-complete.

Corollary 4. *#MaxModels is #·coNP-complete via subtractive reductions.*

We are now prepared to show that counting the number of maximal consistent subsets has the same complexity as counting the number of minimal inconsistent subsets.

Proposition 17. *The problem of counting all maximal consistent subsets of a given knowledge base is #·coNP-complete under subtractive reductions.*

The other component of $\mathcal{I}_{mc}$, the number of self-conflicting formulas, is arguably easier to compute, the functional problem is FP$^{\text{NP}[\log n]}$-complete.

Proposition 18. *The problem of counting unsatisfiable formulas in a given knowledge base is* FP$^{\text{NP}[\log n]}$*-complete.*

Finally, in this section, we show that counting both maximal consistent subsets and self-contradictory formulas, with the shorthand #MCAndContradictory, is a #·coNP-complete problem as well. In particular hardness holds if no self-contradictory formulas occur in the knowledge base. This problem corresponds to $\text{VALUE}_{\mathcal{I}_{mc}}$, except for being off by a constant one.

Proposition 19. *#MCAndContradictory is #·coNP-complete under subtractive reductions. Hardness holds even for knowledge bases without self-contradictory formulas.*

4.4.4. The measure $\mathcal{I}_{is}$

The final measure we consider in this paper is $\mathcal{I}_{is}$ where $\mathcal{I}_{is}(\mathcal{K})$ is defined to be the logarithm of the number of independent sets of $\text{MI}(\mathcal{K})$. We start this subsection with defining a translation of a knowledge base to another, so that the minimal inconsistent subsets of the first are in a correspondence to the independent sets of the second.

$$R_{\#\text{IS}}(\mathcal{K}) = \{\neg p\} \cup \bigcup_{\alpha \in \mathcal{K}} (\alpha \vee p)$$

Lemma 9. *Let $\mathcal{K}$ be a knowledge base and $R_{\#\text{IS}}(\mathcal{K}) = \mathcal{K}'$. It holds that $|\text{MI}(\mathcal{K})| = |\text{IS}(\mathcal{K}')| + 1$.*

Similarly as in previous sections, we show completeness (hardness) for the decision problems now that we are equipped with the preceding lemma.

Proposition 20. $\text{LOWER}_{\mathcal{I}_{is}}$ *and* $\text{UPPER}_{\mathcal{I}_{is}}$ *are* CNP-*complete and* $\text{EXACT}_{\mathcal{I}_{is}}$ *is* $C_{=}NP$ *hard.*

Next, we show complexity of the functional problem $\text{VALUE}_{\mathcal{I}_{is}}$. We omit the logarithm in our investigation. We dubbed the corresponding problem by #is. The logarithm is omitted to make classification of the problem amenable to counting complexity classes (which are defined in non-negative integers as output). We see this not as an obfuscation: the logarithm can be obtained from the number of independent sets with negligible computational cost.

Proposition 21. *#is is #·coNP-complete via subtractive reductions.*

5. Some Notes on Tractable Cases

Due to the intractability of general propositional logic, many fragments of it have been investigated in order to make it usable for actual applications. As a matter of fact, there exist quite expressive fragments of propositional logic such as the *Horn fragment* and the *Krom fragment* where the satisfiability problem can be solved in polynomial time. In the following, we investigate whether our problems related to inconsistency measurement become tractable (or at least easier) when moving to such fragments of propositional logic. For this initial study we only consider measures, which are located on the first level of the polynomial hierarchy in the general case (see Section 4.1)—namely $\mathcal{I}_d, \mathcal{I}_\eta, \mathcal{I}_c, \mathcal{I}_{hs}, \mathcal{I}_{\text{dalal}}^{\text{hit}}, \mathcal{I}_{\text{dalal}}^{\Sigma}, \mathcal{I}_{\text{dalal}}^{\max}, \mathcal{I}_{\text{forget}}$, and $\mathcal{I}_{mcsc}$. We do make some general comments on the other measures in Section 5.3 as well, but leave a deeper analysis of those for future work.

Let $\mathcal{L}^X(\mathsf{At}) \subseteq \mathcal{L}(\mathsf{At})$ be some subset of propositional logic and define the following problems

$\text{EXACT}_{\mathcal{I}}^X$ **Input**: $\mathcal{K} \subseteq \mathcal{L}^X(\mathsf{At})$, $x \in \mathbb{R}_{\geq 0}^\infty$
                     Output: TRUE iff $\mathcal{I}(\mathcal{K}) = x$

$\text{UPPER}_{\mathcal{I}}^X$ **Input**: $\mathcal{K} \subseteq \mathcal{L}^X(\mathsf{At})$, $x \in \mathbb{R}_{\geq 0}^\infty$
                     Output: TRUE iff $\mathcal{I}(\mathcal{K}) \leq x$

$\text{LOWER}_{\mathcal{I}}^X$ **Input**: $\mathcal{K} \subseteq \mathcal{L}^X(\mathsf{At})$, $x \in \mathbb{R}_{\geq 0}^\infty \setminus \{0\}$
                     Output: TRUE iff $\mathcal{I}(\mathcal{K}) \geq x$

$\text{VALUE}_{\mathcal{I}}^X$ **Input**: $\mathcal{K} \subseteq \mathcal{L}^X(\mathsf{At})$
                     Output: The value of $\mathcal{I}(\mathcal{K})$

The above four problems represent special cases of our general problems.

Before we continue with our investigation on the computational complexity of the above problems, we make some general observations for knowledge bases in *conjunctive normal form.*

Definition 23. A *clause* $c \in \mathcal{L}(\mathsf{At})$ is a disjunction of literals, i. e.

$$c = l_1 \vee \ldots \vee l_k$$

and each l_i is of the form $(\neg)a$ for some $a \in \mathsf{At}$. A knowledge base $\mathcal{K} \subseteq \mathcal{L}(\mathsf{At})$ is in *conjunctive normal form* (CNF) if for all $\alpha \in \mathcal{K}$, α is a clause. Let $\mathcal{L}^{CNF}(\mathsf{At}) \subseteq \mathcal{L}(\mathsf{At})$ be the set of clauses on At.

	$\text{EXACT}_{\mathcal{I}}^{H}$	$\text{UPPER}_{\mathcal{I}}^{H}$	$\text{LOWER}_{\mathcal{I}}^{H}$	$\text{VALUE}_{\mathcal{I}}^{H}$
$\mathcal{I}_d$	P	P	P	FP
$\mathcal{I}_\eta$	(D_1^p)	(NP)	(coNP)	$(\mathsf{FP}^{\mathsf{NP}[n]})$
$\mathcal{I}_c$	(D_1^p)	(NP)	(coNP)	$(\mathsf{FP}^{\mathsf{NP}[\log n]})$
$\mathcal{I}_{hs}$	P	P	P	FP
$\mathcal{I}_{\mathrm{dalal}}^{\Sigma}$	(D_1^p)	NP-c	coNP-c	$(\mathsf{FP}^{\mathsf{NP}[\log n]})$
$\mathcal{I}_{\mathrm{dalal}}^{\mathrm{max}}$	P	P	P	FP
$\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}$	(D_1^p)	NP-c	coNP-c	$(\mathsf{FP}^{\mathsf{NP}[\log n]})$
$\mathcal{I}_{mcsc}$	(D_1^p)	(NP)	(coNP)	$(\mathsf{FP}^{\mathsf{NP}[\log n]})$
$\mathcal{I}_{\mathrm{forget}}$	(D_1^p)	NP-c	coNP-c	$(\mathsf{FP}^{\mathsf{NP}[\log n]})$

Table 3: Computational complexity in the Horn/Krom fragment of the considered inconsistency measures (all statements are membership statements, an additionally attached "-c" also indicates completeness for the class); cells in grey indicate cases with the same complexity as in the general case; for cells where the class is given in parentheses we have no stronger results than those provided in Section 4.1

Restricting the syntactical form of knowledge bases to CNF already makes several inconsistency measures coincide with each other and simplifies their definition.

Proposition 22. *For all $\mathcal{K} \subseteq \mathcal{L}^{CNF}(\mathsf{At})$, $\mathcal{I}_d(\mathcal{K}) = \mathcal{I}_{dalal}^{\max}(\mathcal{K}) = \mathcal{I}_{hs}(\mathcal{K})$.*

Proposition 23. *For all $\mathcal{K} \subseteq \mathcal{L}^{CNF}(\mathsf{At})$, $\mathcal{I}_{dalal}^{\Sigma}(\mathcal{K}) = \mathcal{I}_{dalal}^{hit}(\mathcal{K}) = \mathcal{I}_{forget}(\mathcal{K})$.*

We will now investigate the computational complexity of the problems above wrt. our inconsistency measures for the *Horn fragment* (Section 5.1) and the *Krom fragment* (Section 5.2). Table 3 summarises the results of these two sections, which are identical for both fragments. For the measures $\mathcal{I}_\eta$, $\mathcal{I}_c$, and $\mathcal{I}_{mcsc}$ we could not establish stronger results than the ones shown for the general case. The membership statements for the rows of these measures in Table 3 follow directly from the general results before. Showing completeness for these classes or showing tractability is an open problem.

5.1. The Horn fragment

The first fragment of propositional logic that we will consider is the *Horn fragment* (Horn, 1951).

Definition 24. A *Horn clause* is a clause $l_1 \vee \ldots \vee l_n$ where at most one literal from $l_1, \ldots, l_n$ is unnegated. For a propositional signature At let $\mathcal{L}^H(\mathsf{At}) \subseteq \mathcal{L}(\mathsf{At})$ be the set of Horn clauses on At.

Obviously, $\mathcal{L}^H(\mathsf{At}) \subseteq \mathcal{L}^{CNF}(\mathsf{At})$. The Horn fragment is computationally attractive as deciding whether a set of Horn clauses $\mathcal{K} \subseteq \mathcal{L}^H(\mathsf{At})$ is satisfiable is in P (Papadimitriou, 1994). This problem is also called HornSAT.

First, a direct corollary of the fact that HornSAT is in P is the following.

Proposition 24. $\text{EXACT}_{\mathcal{I}_d}^H$, $\text{UPPER}_{\mathcal{I}_d}^H$, $\text{LOWER}_{\mathcal{I}_d}^H$, and $\text{VALUE}_{\mathcal{I}_d}^H$ can be computed in polynomial time.

Due to Proposition 22 the above result implies the next observation.

Corollary 5. $\text{EXACT}_{\mathcal{I}}^H$, $\text{UPPER}_{\mathcal{I}}^H$, $\text{LOWER}_{\mathcal{I}}^H$, and $\text{VALUE}_{\mathcal{I}}^H$ can be computed in polynomial time for $\mathcal{I} \in \{\mathcal{I}_{dalal}^{\max}, \mathcal{I}_{hs}\}$.

Interestingly, although $\mathcal{I}_{dalal}^{\max}$ becomes tractable in $\mathcal{L}^H(\mathsf{At})$, this is not the case for $\mathcal{I}_{dalal}^{\Sigma}(\mathcal{K})$ (and its equivalent measures $\mathcal{I}_{dalal}^{\text{hit}}$ and $\mathcal{I}_{\text{forget}}$).

Proposition 25. For $\mathcal{I} \in \{\mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{forget}\}$, $\text{UPPER}_{\mathcal{I}}^H$ is NP-complete, $\text{LOWER}_{\mathcal{I}}^H$ is coNP-complete, $\text{EXACT}_{\mathcal{I}}^H$ is in D_1^p, and $\text{VALUE}_{\mathcal{I}}^H$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.

5.2. The Krom fragment

The second fragment of propositional logic that we will consider is the *Krom fragment*.

Definition 25. A *Krom clause* is a clause $l_1 \vee l_2$. For a propositional signature At let $\mathcal{L}^K(\mathsf{At}) \subseteq \mathcal{L}(\mathsf{At})$ be the set of Krom clauses on At.

As it is the case with the Horn fragment, the Krom fragment is computationally attractive as deciding whether a set of Krom clauses $\mathcal{K} \subseteq \mathcal{L}^K(\mathsf{At})$ is satisfiable is in P (Papadimitriou, 1994). This problem is also called KromSAT. A central tool for showing this fact and other statements in this section is the translation of a Krom knowledge base into a directed graph as follows, cf. (Papadimitriou, 1994).

Definition 26. Let $\mathcal{K} \subseteq \mathcal{L}^K(\mathsf{At})$. The *induced graph* $G_{\mathcal{K}}$ of $\mathcal{K}$ is a directed graph $G_{\mathcal{K}} = (V, E)$ defined via

$$V = \{a, \neg a \mid a \in \mathsf{At}\}$$
$$E = \{(l, l') \mid \bar{l} \vee l' \in \mathcal{K}\}$$

One can show that $\mathcal{K} \subseteq \mathcal{L}^{\mathsf{K}}(\mathsf{At})$ is inconsistent if and only if there is an atom $a \in \mathsf{At}$ such that there are directed paths in $G_{\mathcal{K}}$ from a to $\neg a$ and from $\neg a$ to a, i.e., there is a cycle involving a and $\neg a$, see e.g. (Papadimitriou, 1994).

As in the previous section, a direct corollary of the fact that KromSAT is in P is the following.

Proposition 26. $\mathrm{EXACT}_{\mathcal{I}_d}^K$, $\mathrm{UPPER}_{\mathcal{I}_d}^K$, $\mathrm{LOWER}_{\mathcal{I}_d}^K$, and $\mathrm{VALUE}_{\mathcal{I}_d}^K$ can be computed in polynomial time.

Due to Proposition 22 the above result implies the next observation.

Corollary 6. $\mathrm{EXACT}_{\mathcal{I}}^K$, $\mathrm{UPPER}_{\mathcal{I}}^K$, $\mathrm{LOWER}_{\mathcal{I}}^K$, and $\mathrm{VALUE}_{\mathcal{I}}^K$ can be computed in polynomial time for $\mathcal{I} \in \{\mathcal{I}_{dalal}^{\max}, \mathcal{I}_{hs}\}$.

Analogously to the Horn case, $\mathcal{I}_{\mathrm{dalal}}^{\max}$ becomes tractable in $\mathcal{L}^{\mathsf{K}}(\mathsf{At})$, and, also analogously to the Horn case, this is not the case for $\mathcal{I}_{\mathrm{dalal}}^{\Sigma}(\mathcal{K})$ (and its equivalent measures $\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}$ and $\mathcal{I}_{\mathrm{forget}}$).

Proposition 27. For $\mathcal{I} \in \{\mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{forget}\}$, $\mathrm{UPPER}_{\mathcal{I}}^K$ is NP-complete, $\mathrm{LOWER}_{\mathcal{I}}^K$ is coNP-complete, $\mathrm{EXACT}_{\mathcal{I}}^K$ is in D_1^p, and $\mathrm{VALUE}_{\mathcal{I}}^K$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.

5.3. Higher-level measures

For measures located beyond the first level of the polynomial hierarchy it is less likely to obtain tractability when moving to fragments such as the Horn or Krom fragment, as those typically possess some additional complexity. For example, many measures beyond the first level of the polynomial hierarchy make use of minimal inconsistent sets, which may still be exponential in number in knowledge bases in CNF and its fragments. However, some problems of these measures may become *easier* and drop a level in the polynomial hierarchy. In this section, we look at some of these measures in an exemplary fashion and provide some membership results. We leave an in-depth analysis and, in particular, completeness results for future work.

In (Davydov et al., 1998) it has been shown that checking whether a set $M \subseteq \mathcal{L}^H(\mathsf{At})$ is a minimal inconsistent set is solvable in polynomial time. This problem is also tractable in $\mathcal{L}^{\mathsf{K}}(\mathsf{At})$.

Lemma 10. *Deciding whether a set $M \subseteq \mathcal{L}^K(\mathsf{At})$ is a minimal inconsistent set is in P.*

These two observations do not imply tractability of the problems related to $\mathcal{I}_p$ and $\mathcal{I}_{mv}$, but they lead to a decrease of complexity of one level in the polynomial hierarchy.

Proposition 28. *For $X \in \{H, K\}$, $\text{LOWER}_{\mathcal{I}_p}^X$ is in NP, $\text{UPPER}_{\mathcal{I}_p}^X$ is in coNP, $\text{EXACT}_{\mathcal{I}_p}^X$ is in D_1^p, and $\text{VALUE}_{\mathcal{I}_p}^X$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.*

Proposition 29. *For $X \in \{H, K\}$, $\text{LOWER}_{\mathcal{I}_{mv}}^X$ is in NP, $\text{UPPER}_{\mathcal{I}_{mv}}^X$ is in coNP, $\text{EXACT}_{\mathcal{I}_{mv}}^X$ is in D_1^p, and $\text{VALUE}_{\mathcal{I}_{mv}}^X$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.*

As NP oracle calls that ask for satisfiability can be solved in polynomial time in the Horn and Krom fragment, the complexity of problems related to $\mathcal{I}_{nc}$ also drop one level.

Proposition 30. *For $X \in \{H, K\}$, $\text{LOWER}_{\mathcal{I}_{nc}}^X$ is in NP, $\text{UPPER}_{\mathcal{I}_{nc}}^X$ is in coNP, $\text{EXACT}_{\mathcal{I}_{nc}}^X$ is in D_1^p, and $\text{VALUE}_{\mathcal{I}_{nc}}^X$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.*

Also the complexity of $\mathcal{I}_{CC}$ drops from the third level to the second level of the polynomial hierarchy.

Proposition 31. *For $X \in \{H, K\}$, $\text{LOWER}_{\mathcal{I}_{CC}}^X$ is in Σ_2^p, $\text{UPPER}_{\mathcal{I}_{CC}}^X$ is in Π_2^p, $\text{EXACT}_{\mathcal{I}_{CC}}^X$ is in D_2^p, and $\text{VALUE}_{\mathcal{I}_{CC}}^X$ is in $\mathsf{FP}^{\Sigma_2^p[\log n]}$.*

A further corollary from the fact that checking whether a set is minimally unsatisfiable is polynomially solvable in the Horn and Krom fragment, is that counting MIs in this fragment is a problem in #P. Hardness can be proven, as well, by a reduction from the problem of counting all simple paths in a given directed graph (i.e. paths without repeated vertices). The latter problem was proven to be #P-complete (Valiant, 1979b), however using another kind of reduction, namely Turing reductions. In this case, this means that a problem A can be Turing-reduced to problem B if there is a polynomial-time algorithm that solves A and can query B as a subproblem (in contrast, e.g., subtractive reductions do not allow a computation "afterwards", i.e., after querying B, which induces different formal results for some complexity classes). It is currently open whether this problem is hard also for parsimonious or subtractive reductions. The hardness results for counting MIs, in fact, have been proven already for a related concept in description logics (Peñaloza and Sertkaya, 2017, Corollary 42).

Proposition 32. *For $X \in \{H, K\}$, $\text{VALUE}_{\mathcal{I}_{\mathsf{MI}}}^X$ is #P-complete under Turing reductions.*

Similarly, the measure $\mathcal{I}_{mc}$ exhibits milder complexity. First, there are no self-contradictory elements in a Horn or Krom knowledge base (a single clause is always satisfiable). Finally, since checking whether a set of Horn (Krom) clauses is satisfiable is decidable in polynomial time, also checking maximality is decidable in polynomial time: it suffices to consider whether each superset with exactly one additional element from the given knowledge base is inconsistent. Thus, the following result also follows straightforwardly.

Proposition 33. *For $X \in \{H, K\}$, $\mathrm{VALUE}_{\mathcal{I}_{mc}}^{X}$ is in $\#\mathsf{P}$.*

This closes our outlook on the behaviour of the considered measures in fragments of propositional logic.

6. Related Works

Our analysis of the computational complexity of inconsistency measures in the setting of propositional logic continues and updates our previous work (Thimm and Wallner, 2016). Another work that explicitly discusses computational complexity of inconsistency measures is (Xiao and Ma, 2012) whose results we already mentioned in Section 4.2. In this section, we will now discuss further works that are related to our current work. More precisely, we will review works discussing general computational issues in inconsistency measurement in Section 6.1 and analyses of related computational complexity questions in Section 6.2.

6.1. Computational issues in inconsistency measurement

Other works on inconsistency measurement that address computational issues are as follows. These works either develop and evaluate algorithmic solutions for certain inconsistency measures or discuss computational complexity of inconsistency measurement in other formalisms than propositional logic.

In (Konieczny and Roussel, 2013) the authors present a software platform for reasoning, among other, with the Shapley inconsistency value (Hunter and Konieczny, 2010). While there are not given many details, the platform is written in Java and makes uses of external tools such as SAT solvers. McAreavey et al. (2014) present algorithmic approaches to count minimal inconsistent sets in arbitrary knowledge bases (not necessarily in CNF) with the motivation of inconsistency measures such as $\mathcal{I}_{MI}$. They empirically evaluate their approach and show its feasibility. Ammoura et al. (2017) develop two

formalisations of the computation of $\mathcal{I}_{mcsc}$: first, as an integer linear program and, second, as a partial MaxSAT problem. These formalisations are shown to be able to faithfully compute the value of $\mathcal{I}_{mcsc}$. Similarly, (Jabbour et al., 2016) present a formalisation of the computation of $\mathcal{I}_{CC}$ in terms of a similar SAT optimisation problem. An experimental evaluation shows that its implementation can compute inconsistency values in reasonable time. The work (Thimm, 2016b) develops a general scheme for approximation algorithms for inconsistency measurement. Moreover, it develops more sophisticated approximation algorithms for the measures $\mathcal{I}_{hs}$ and $\mathcal{I}_c$. Through an empirical evaluation these algorithms are compared and it is shown that they scale linearly with the size of the knowledge base while providing sufficient accuracy. In (Thimm, 2017a) a new family of inconsistency measures based on fuzzy logic is developed and it is shown that the corresponding computational problems lie on the first level of the polynomial hierarchy.

In (Brewka et al., 2017; Ulbricht et al., 2018b), a generalisation of the concept of inconsistency suitable for non-monotonic logics is developed and applied to the framework of inconsistency measurement. More formally, a set of formulas is *strongly inconsistent* if all its supersets within a knowledge base are inconsistent as well. In (Brewka et al., 2017; Ulbricht et al., 2018b) several question pertaining to computational complexity are investigated as well. The works (Ulbricht et al., 2016, 2018a) develop inconsistency measures for logic programs under the answer set semantics and also investigate computational complexity questions. Due to the higher complexity of answer set programming itself their results showed that inconsistency measurement in that context is usually one level higher in the polynomial hierarchy as similar questions in our classic setting.

6.2. Related complexity analyses

Related notions and concepts to inconsistency measurement have been analyzed wrt. their complexity. We begin with related work regarding minimal inconsistent subsets of a set of formulas (MIs). In (Davydov et al., 1998) the complexity of deciding whether a given formula in CNF is an MI is analyzed, i.e., checking whether the set of clauses is inconsistent and all proper subsets of the clauses are satisfiable. While in general this problem is D_1^p-complete (Papadimitriou and Wolfe, 1988), it is shown in (Davydov et al., 1998) that when the number of clauses is bounded by the number of variables n plus an integer k certain milder results hold. In particular, when $k = 0$, then the formula is never an MI. If $k = 1$, then the problem of deciding

whether the formula is an MI is in P. In (Marques-Silva and Previti, 2014) (based on results from (Liberatore, 2005)) complexity of checking whether a clause (a formula) is in an MI is shown. In our work, we partially rely on some of the results (e. g. D_1^p-completeness of checking whether a set is an MI). In contrast to the mentioned works, we study different computational tasks relating to MIs that arise from the considered inconsistency measures, and without bounds on variables or clauses.

Several complexity results related to our work are given in (Peñaloza and Sertkaya, 2017). There, they study a related concept in description logics, namely that of so-called minimal axiom sets (MinAs), which are minimal subsets (of a given set of axioms) that entail a given formula. If the entailed formula is unsatisfiable, then the set is a minimally inconsistent subset. Particularly related are then the results of counting MinAs in case of Horn and Krom formulas (they show the result for formulas called core in their paper, which are a fragment of Krom formulas), which they prove to be #P-complete, a result which we make use to directly show the related result of counting MIs on Horn and Krom formulas (see Proposition 32). Another related result of theirs is that counting MinAs in a more general setting is #NP-complete, which relates to our result of counting MIs of general knowledge bases, which is #·coNP-complete (recall that #NP = #·coNP). In contrast, we provide our proof directly for our setting, use a different reduction, and show further counting results for this inconsistency measure, particularly for $\textsc{Upper}_{\mathcal{I}_{\mathsf{MI}}}$, $\textsc{Lower}_{\mathcal{I}_{\mathsf{MI}}}$, and $\textsc{Exact}_{\mathcal{I}_{\mathsf{MI}}}$. In more detail, they use weakly parsimonious reductions (informally speaking, these are polynomial reductions from whose result one can read in polynomial time the number of solutions). However, we note that our reduction can be straightforwardly rewritten to be a weakly parsimonious reduction. Nevertheless, we formally link the complexities of the problem of counting MIs of general propositional logic knowledge bases and the problem studied by Peñaloza and Sertkaya (2017). Another result, related to our counting context, concerns the *enumeration complexity* of MIs of a Horn formula. Enumeration complexity studies complexity when the task is to enumerate (all) solutions. It holds that MI enumeration is possible with so-called polynomial delay for Horn formulas. This means that there are algorithms which can compute each of the solutions with a polynomial running time that is only dependent on the input size (however, in general, this does not imply that there are at most polynomially many solutions). Furthermore, efficient algorithms for the enumeration of MIs is also the topic of (Marques-Silva et al., 2016; Arif et al., 2015).

Another line of related research studies complexity of, and algorithmic approaches to, SAT and #SAT (on different forms of input). In fact, there is an abundance of research on this topic (see, e. g., the Handbook on Satisfiability (Biere et al., 2009)), however no inconsistency measure in the general case, other than $\mathcal{I}_d$, studied in this paper directly corresponds to a SAT or #SAT problem. We overview some results from this field (particularly ones that are "closer" to our analysis). It is well known that SAT is NP-complete, even in restricted cases (such as for CNF knowledge bases with at most three variables per clause), and, likewise, it is well known that #SAT is #P-complete, again, even under restricted cases such as for Krom formulas with only positive literals (Valiant, 1979b). There are also so-called dichotomy results, studying when problems are tractable and when intractable, and finding exact barriers (Creignou and Hermann, 1996). Results regarding fragments of propositional logic that exhibit milder complexity for one of the two problems is also prevalent in the literature, e. g., in (Pretolani, 1996) it is analyzed how to extend tractable classes of SAT. In (Andrei et al., 2009) the complexity of both problems are studied when clauses are (not) related (literals appearing positive and negative in different clauses). Also the optimization problem, MaxSAT, has been studied in detail. We make use of the well-known result that MaxSAT is NP-complete, even in the restricted case of Horn formulas (Jaumard and Simeone, 1987).

We move on to problems related to maximal consistent subsets. In the field of databases, a related concept to maximal consistent subsets, as studied in this paper, is that of database repairs, and complexity issues relating to such repairs (Arenas et al., 1999, 2003; Chomicki and Marcinkowski, 2005; Flesca et al., 2010; Staworko and Chomicki, 2010; Maslowski and Wijsen, 2013; Wijsen, 2014; Chomicki and Wijsen, 2016; Livshits and Kimelfeld, 2017; Parisi and Grant, 2017). Repairs in the database context can take many forms. A common approach to repairs of (inconsistent) databases is to consider those database instances with minimal distance to the given database that satisfy certain integrity constraints (Arenas et al., 1999). For instance, a repair can be seen as parts to add and remove from a database to restore consistency. If only removals are permitted, repairs may take the form of maximal consistent subsets of the database that respect the constraints (Chomicki and Marcinkowski, 2005). Similar to forms of databases, constraints can be heterogeneous with varying complexity (see, e.g., (Arenas et al., 2003; Chomicki and Marcinkowski, 2005; Flesca et al., 2010; Staworko and Chomicki, 2010; Maslowski and Wijsen, 2013; Wijsen, 2014; Parisi and

Grant, 2017)) to check whether a constraint is satisfied. Maximal consistent subsets, as studied in this article, are, naturally, a related formalism. Maximality (minimal distance) of repairs relates to maximality of subsets of knowledge bases, and checking integrity constraints relates to consistency checking of the subset. Depending on the concrete conditions of what is a repair, complexity varies considerably. For instance, checking an integrity constraint may be coNP-hard, which can lead to complexity of reasoning based on repairs (e. g., considering the intersection of all repairs) being on the second level of the polynomial hierarchy. On the other hand, constraints may also refer to (computationally) simpler problems, such as checking whether no two tuples have the same value for a primary key (which is then decidable in polynomial time). In such a case, also maximality may be decidable in polynomial time. Counting complexity was analyzed for repairs in (Maslowski and Wijsen, 2013; Wijsen, 2014; Livshits and Kimelfeld, 2017), with results that counting can be, e. g., $\#$P-hard. Further related works to complexity of maximal consistent subsets are given in (Cayrol et al., 1998; Marques-Silva and Previti, 2014), where reasoning based on maximal consistent subsets is analyzed with and without preferences over such maximal consistent subsets. However, we are not aware of a corresponding result to our $\#\cdot$coNP-completeness result of measure $\mathcal{I}_{mc}$ (Proposition 19).

Another related field is that of (propositional) abduction. In propositional abduction, the aim is to find explanations for certain observations, given certain background knowledge. For instance, one problem in this field asks for a minimal (e. g. for subset-inclusion) set of hypotheses that explain (formally: entail) an observation and that is consistent with a given formula. Complexity of this problem, and variants, has been studied in (Eiter and Gottlob, 1995). Related to our results is a complexity analysis of counting the number of such explanations in (Hermann and Pichler, 2010). Their results imply similar counting complexity for some variants to inconsistency measures that have $\#\cdot$coNP complexity.

Finally, another related field is that of belief revision (Alchourrón et al., 1985), as mentioned in the introduction, in that one can cope with inconsistency by revision. Complexity has been studied, as well, in (Eiter and Gottlob, 1992; Liberatore and Schaerf, 2001; Liberatore, 1997; Pfandler et al., 2015). Complexity is, naturally, also high for revision: many problems reside on the second level of the polynomial hierarchy. Related to our measures is revision by the Dalal operator (Dalal, 1988), which is based on a similar distance notion that is also used in $\mathcal{I}_{\text{dalal}}^{\Sigma}$, $\mathcal{I}_{\text{dalal}}^{\text{hit}}$, and $\mathcal{I}_{\text{dalal}}^{\max}$.

7. Discussion and Summary

The contributions of this paper provide new insights into the challenge of measuring inconsistency and allow for a broader comparison of existing measures in terms of their complexity. One of the key insights in this paper is the partitioning of inconsistency measures in four categories of complexity, i. e., measures on the first three levels of the polynomial hierarchy and those beyond the third level. This also shows that inconsistency measurement is sometimes not computationally harder than solving the classical satisfiability problem SAT (for the measures residing on the first level of the polynomial hierarchy, see Section 4.1). However, our results also show that inconsistency measurement can be computationally demanding, as shown in Sections 4.2, 4.3, and 4.4.

It is also interesting to note that the computational complexity of inconsistency measures does not necessarily correlate with their "conceptual" complexity. For example, the measure $\mathcal{I}_{\mathsf{MI}}$ was one of the first inconsistency measures presented and follows a simple idea to measure inconsistency, i. e., simply taking the number of minimal inconsistent subsets. Both the results for the corresponding decisions problems and the $\#\cdot\mathsf{coNP}$-completeness result of the function problem show that $\mathcal{I}_{\mathsf{MI}}$ belongs to the computationally most complex inconsistency measures. Compare this to e. g. the measure $\mathcal{I}_{\mathrm{dalal}}^{\Sigma}$ which features a quite complex definition, involving distances between propositional interpretations, but belongs to the easiest class of measures. Similar observations can be made when comparing the complexity and *expressivity* of an inconsistency measure. In (Thimm, 2016a) *expressivity* has been introduced as a quantitative measure to assess how many levels of inconsistent knowledge bases an inconsistency measure can distinguish. Our Lemmas 1–3 provided some simplifying observations on this aspect. There, one can see that, although there is a general tendency of more expressive measures (such as $\mathcal{I}_{\mathsf{MI}^c}$ and $\mathcal{I}_{\mathrm{is}}$) having higher computational complexity, there is no universal correlation. For example, the measure $\mathcal{I}_{\eta}$ may exhibit an exponential number of inconsistency values, see Lemma 2, but the problems related to it are on the first level of the polynomial hierarchy. On the other hand, the measure $\mathcal{I}_{CC}$ has rather low expressivity, see Lemma 1, but exhibits large computational complexity.

The results of this paper can be used for future work in two directions. First, our general methodology of analysing computational complexity can help researchers to conduct similar analyses on novel inconsistency measures.

The general relations established e. g. in Lemmas 4 and 6 show that it is sufficient to focus on the decision problems $\text{UPPER}_{\mathcal{I}}$ or $\text{LOWER}_{\mathcal{I}}$. Providing a completeness result for one of these problems is sufficient to obtain a rather good classification of the computational complexity of an inconsistency measure. Furthermore, the problems and reductions we used to show completeness of problems $\text{EXACT}_{\mathcal{I}}$ and $\text{VALUE}_{\mathcal{I}}$—in particular those discussed in Section 4.1—may be used directly to obtain corresponding completeness proofs of novel measures or may inspire new reductions. Part of future work is the analysis of recently proposed measures such as (Jabbour et al., 2017; De Bona and Hunter, 2017; De Bona et al., 2018) wrt. our methodology. Second, the next logical step for making use of our analysis is to develop new algorithms for the considered measures, also taking our initial study on tractable cases into account, cf. Section 5. Current implementations such as *Tweety@Web*[7] (Thimm, 2017c) mostly rely on naive brute-force approaches to determine inconsistency values. The insights and reductions gained in our analysis can help in devising new algorithms that rely on e. g. reductions to QBF and QBF solvers (Lonsing et al., 2016).

Acknowledgments

References

Alchourrón, C. E., Gärdenfors, P., Makinson, D., 1985. On the logic of theory change: Partial meet contraction and revision functions. Journal of Symbolic Logic 50 (2), 510–530.

Amgoud, L., Ben-Naim, J., 2017. Measuring disagreement in argumentation graphs. In: Moral, S., Pivert, O., Sánchez, D., Marín, N. (Eds.), Proceedings of SUM 2017. Vol. 10564 of Lecture Notes in Computer Science. Springer, pp. 208–222.

Ammoura, M., Raddaoui, B., Salhi, Y., Oukacha, B., 2015. On measuring inconsistency using maximal consistent sets. In: Destercke, S., Denoeux,

---

[7]`http://tweetyproject.org/w/incmes`

T. (Eds.), Proceedings of ECSQARU 2015. Vol. 9161 of Lecture Notes in Computer Science. Springer, pp. 267–276.

Ammoura, M., Salhi, Y., Oukacha, B., Raddaoui, B., 2017. On an MCS-based inconsistency measure. International Journal of Approximate Reasoning 80, 443–459.

Andrei, S., Grigoras, G., Rinard, M. C., Yap, R. H. C., 2009. A hierarchy of tractable subclasses for SAT and counting SAT problems. In: Watt, S. M., Negru, V., Ida, T., Jebelean, T., Petcu, D., Zaharie, D. (Eds.), Proceedings of SYNASC 2009. IEEE Computer Society, pp. 61–68.

Ansótegui, C., Bonet, M. L., Levy, J., 2013. SAT-based MaxSAT algorithms. Artificial Intelligence 196, 77–105.

Arenas, M., Bertossi, L. E., Chomicki, J., 1999. Consistent query answers in inconsistent databases. In: Vianu, V., Papadimitriou, C. H. (Eds.), Proceedings of PODS. ACM Press, pp. 68–79.

Arenas, M., Bertossi, L. E., Chomicki, J., He, X., Raghavan, V., Spinrad, J. P., 2003. Scalar aggregation in inconsistent databases. Theoretical Computer Science 296 (3), 405–434.

Arif, M. F., Mencía, C., Marques-Silva, J., 2015. Efficient MUS enumeration of horn formulae with applications to axiom pinpointing. In: Heule, M., Weaver, S. A. (Eds.), Proceedings of SAT. Vol. 9340 of Lecture Notes in Computer Science. Springer, pp. 324–342.

Besnard, P., 2016. Forgetting-based inconsistency measure. In: Schockaert, S., Senellart, P. (Eds.), Proceedings of SUM 2016. Vol. 9858 of Lecture Notes in Computer Science. Springer, pp. 331–337.

Biere, A., Heule, M., van Maaren, H., Walsh, T. (Eds.), 2009. Handbook of Satisfiability. Vol. 185 of Frontiers in Artificial Intelligence and Applications. IOS Press.

Brewka, G., Eiter, T., Truszczynski, M., 2011. Answer set programming at a glance. Communications of the ACM 54 (12), 92–103.

Brewka, G., Thimm, M., Ulbricht, M., 2017. Strong inconsistency in non-monotonic reasoning. In: Sierra, C. (Ed.), Proceedings of IJCAI 2017. ijcai.org, pp. 901–907.

Cayrol, C., Lagasquie-Schiex, M., Schiex, T., 1998. Nonmonotonic reasoning: From complexity to algorithms. Annals of Mathematics and Artificial Intelligence 22 (3-4), 207–236.

Chomicki, J., Marcinkowski, J., 2005. Minimal-change integrity maintenance using tuple deletions. Information and Computation 197 (1-2), 90–121.

Chomicki, J., Wijsen, J., 2016. Consistent query answering for atemporal constraints over temporal databases. In: Dyreson, C. E., Hansen, M. R., Hunsberger, L. (Eds.), Proceedings TIME. IEEE Computer Society, pp. 149–156.

Chvátal, V., 1983. Linear Programming. Freeman, San Francisco.

Condotta, J., Raddaoui, B., Salhi, Y., 2016. Quantifying conflicts for spatial and temporal information. In: Baral, C., Delgrande, J. P., Wolter, F. (Eds.), Proceedings of KR 2016. AAAI Press, pp. 443–452.

Creignou, N., Hermann, M., 1996. Complexity of generalized satisfiability counting problems. Information and Computation 127, 1–12.

Dalal, M., 1988. Investigations into a theory of knowledge base revision. In: Shrobe, H. E., Mitchell, T. M., Smith, R. G. (Eds.), Proceedings of the 7th National Conference on Artificial Intelligence. AAAI Press / The MIT Press, pp. 475–479.

Davydov, G., Davydova, I., Büning, H. K., Nov 1998. An efficient algorithm for the minimal unsatisfiability problem for a subclass of CNF. Annals of Mathematics and Artificial Intelligence 23 (3), 229–245.

De Bona, G., Finger, M., 2015. Measuring inconsistency in probabilistic logic: Rationality postulates and dutch book interpretation. Artificial Intelligence 227, 140–164.

De Bona, G., Grant, J., Hunter, A., Konieczny, S., 2018. Towards a unified framework for syntactic inconsistency measures. In: McIlraith, S. A., Weinberger, K. Q. (Eds.), Proceedings of AAAI 2018. AAAI Press, pp. 1803–1810.

De Bona, G., Hunter, A., 2017. Localising iceberg inconsistencies. Artificial Intelligence 246, 118–151.

Decker, H., 2011. Measure-based inconsistency-tolerant maintenance of database integrity. In: Schewe, K., Thalheim, B. (Eds.), Proceedings of the 5th International Workshop on Semantics in Data and Knowledge Bases (SDKB'11), Revised Selected Papers. Vol. 7693 of Lecture Notes in Computer Science. Springer, pp. 149–173.

Doder, D., Raskovic, M., Markovic, Z., Ognjanovic, Z., 2010. Measures of inconsistency and defaults. International Journal of Approximate Reasoning 51, 832–845.

Durand, A., Hermann, M., Kolaitis, P. G., 2005. Subtractive reductions and complete problems for counting complexity classes. Theoretical Computer Science 340 (3), 496–513.

Eiter, T., Gottlob, G., 1992. On the complexity of propositional knowledge base revision, updates, and counterfactuals. Artificial Intelligence 57 (2-3), 227–270.

Eiter, T., Gottlob, G., 1995. The complexity of logic-based abduction. Journal of the ACM 42 (1), 3–42.

Flesca, S., Furfaro, F., Parisi, F., 2010. Querying and repairing inconsistent numerical databases. ACM Transactions on Database Systems 35 (2), 14:1–14:50.

Gomes, C. P., Sabharwal, A., Selman, B., 2009. Model counting. In: Biere, A., Heule, M., van Maaren, H., Walsh, T. (Eds.), Handbook of Satisfiability. IOS Press, pp. 633–654.

Graham, R. L., Knuth, D. E., Patashnik, O., 1994. Concrete Mathematics: A Foundation for Computer Science, 2nd Edition. Addison-Wesley.

Grant, J., 1978. Classifications for Inconsistent Theories. Notre Dame Journal of Formal Logic 19 (3), 435–444.

Grant, J., Hunter, A., 2008. Analysing inconsistent first-order knowledge-bases. Artificial Intelligence 172 (8-9), 1064–1093.

Grant, J., Hunter, A., 2011. Measuring consistency gain and information loss in stepwise inconsistency resolution. In: Liu, W. (Ed.), Proceedings of EC-SQARU 2011. Vol. 6717 of Lecture Notes in Computer Science. Springer, pp. 362–373.

Grant, J., Hunter, A., 2013. Distance-based measures of inconsistency. In: van der Gaag, L. C. (Ed.), Proceedings of ECSQARU 2013. Vol. 7958 of Lecture Notes in Computer Science. Springer, pp. 230–241.

Hansson, S. O., 2001. A Textbook of Belief Dynamics. Kluwer Academic Publishers.

Hemaspaandra, L. A., Vollmer, H., 1995. The satanic notations: counting classes beyond #P and other definitional adventures. SIGACT News 26 (1), 2–13.

Hermann, M., Pichler, R., 2010. Counting complexity of propositional abduction. Journal of Computer and System Sciences 76 (7), 634–649.

Horn, A., 1951. On sentences which are true of direct unions of algebras. Journal of Symbolic Logic 16 (1), 14–21.

Hunter, A., Konieczny, S., 2004. Approaches to measuring inconsistent information. In: Inconsistency Tolerance. Springer, pp. 189–234.

Hunter, A., Konieczny, S., 2008. Measuring inconsistency through minimal inconsistent sets. In: Brewka, G., Lang, J. (Eds.), Proceedings of KR 2008. AAAI Press, pp. 358–366.

Hunter, A., Konieczny, S., 2010. On the measure of conflicts: Shapley inconsistency values. Artificial Intelligence 174 (14), 1007–1026.

Jabbour, S., 2016. On inconsistency measuring and resolving. In: Kaminka, G. A., Fox, M., Bouquet, P., Hüllermeier, E., Dignum, V., Dignum, F., van Harmelen, F. (Eds.), Proceedings of ECAI 2016. Vol. 285 of Frontiers in Artificial Intelligence and Applications. IOS Press, pp. 1676–1677.

Jabbour, S., Ma, Y., Raddaoui, B., 2014a. Inconsistency measurement thanks to MUS decomposition. In: Bazzan, A. L. C., Huhns, M. N., Lomuscio, A., Scerri, P. (Eds.), Proceedings of AAMAS 2014,. IFAAMAS/ACM, pp. 877–884.

Jabbour, S., Ma, Y., Raddaoui, B., Sais, L., 2014b. Prime implicates based inconsistency characterization. In: Schaub, T., Friedrich, G., O'Sullivan, B. (Eds.), Proceedings ECAI 2014. Vol. 263 of Frontiers in Artificial Intelligence and Applications. IOS Press, pp. 1037–1038.

Jabbour, S., Ma, Y., Raddaoui, B., Sais, L., 2017. Quantifying conflicts in propositional logic through prime implicates. International Journal of Approximate Reasoning.

Jabbour, S., Ma, Y., Raddaoui, B., Sais, L., Salhi, Y., 2015. On structure-based inconsistency measures and their computations via closed set packing. In: Weiss, G., Yolum, P., Bordini, R. H., Elkind, E. (Eds.), Proceedings of AAMAS 2015. ACM, pp. 1749–1750.

Jabbour, S., Ma, Y., Raddaoui, B., Sais, L., Salhi, Y., 2016. A MIS partition based framework for measuring inconsistency. In: Baral, C., Delgrande, J. P., Wolter, F. (Eds.), Proceedings of KR 2016. AAAI Press, pp. 84–93.

Jabbour, S., Sais, L., 2016. Exploiting MUS structure to measure inconsistency of knowledge bases. In: Kaminka, G. A., Fox, M., Bouquet, P., Hüllermeier, E., Dignum, V., Dignum, F., van Harmelen, F. (Eds.), Proceedings of ECAI 2016. Vol. 285 of Frontiers in Artificial Intelligence and Applications. IOS Press, pp. 991–998.

Jaumard, B., Simeone, B., 1987. On the complexity of the maximum satisfiability problem for Horn formulas. Information Processing Letters 26 (1), 1–4.

Knight, K. M., 2002. A theory of inconsistency. Ph.D. thesis, University Of Manchester.

Konieczny, S., Roussel, S., 2013. A reasoning platform based on the MI shapley inconsistency value. In: van der Gaag, L. C. (Ed.), Proceedings EC-SQARU 2013. Vol. 7958 of Lecture Notes in Computer Science. Springer, pp. 315–327.

Kyburg, H., 1961. Probability and the Logic of Rational Belief. Wesleyan University Press.

Lang, J., Marquis, P., 2010. Reasoning under inconsistency: A forgetting-based approach. Artificial Intelligence 174 (12–13), 799–823.

Liberatore, P., 1997. The complexity of iterated belief revision. In: Afrati, F. N., Kolaitis, P. G. (Eds.), Proceedings of ICDT 1997. Vol. 1186 of Lecture Notes in Computer Science. Springer, pp. 276–290.

Liberatore, P., 2005. Redundancy in logic I: CNF propositional formulae. Artif. Intell. 163 (2), 203–232.

Liberatore, P., Schaerf, M., 2001. Belief revision and update: Complexity of model checking. Journal of Computer and System Sciences 62 (1), 43–72.

Liffiton, M. H., Previti, A., Malik, A., Marques-Silva, J., 2016. Fast, flexible MUS enumeration. Constraints 21 (2), 223–250.

Livshits, E., Kimelfeld, B., 2017. Counting and enumerating (preferred) database repairs. In: Sallinger, E., den Bussche, J. V., Geerts, F. (Eds.), Proceedings of PODS 2017. ACM, pp. 289–301.

Lonsing, F., Seidl, M., Gelder, A. V., 2016. The QBF gallery: Behind the scenes. Artificial Intelligence 237, 92–114.

Ma, J., Liu, W., Miller, P. C., 2012. A characteristic function approach to inconsistency measures for knowledge bases. In: Hüllermeier, E., Link, S., Fober, T., Seeger, B. (Eds.), Proceedings of SUM 2012. Vol. 7520 of Lecture Notes in Computer Science. Springer, pp. 473–485.

Ma, Y., Qi, G., Hitzler, P., Lin, Z., 2007. Measuring inconsistency for description logics based on paraconsistent semantics. In: Mellouli, K. (Ed.), Proceedings of ECSQARU 2007. Vol. 4724 of Lecture Notes in Computer Science. Springer, pp. 30–41.

Ma, Y., Qi, G., Xiao, G., Hitzler, P., Lin, Z., 2010. Computational complexity and anytime algorithm for inconsistency measurement. International Journal of Software and Informatics 4 (1), 3–21.

Makinson, D., 2005. Bridges from Classical to Nonmonotonic Logic. College Publications.

Marques-Silva, J., 2012. Computing minimally unsatisfiable subformulas: State of the art and future directions. Multiple-Valued Logic and Soft Computing 19 (1–3), 163–183.

Marques-Silva, J., Ignatiev, A., Mencía, C., Peñaloza, R., 2016. Efficient reasoning for inconsistent horn formulae. In: Michael, L., Kakas, A. C. (Eds.), Proceedings of JELIA 2016. Vol. 10021 of Lecture Notes in Computer Science. pp. 336–352.

Marques-Silva, J., Previti, A., 2014. On computing preferred MUSes and MCSes. In: Sinz, C., Egly, U. (Eds.), Proceedings of SAT 2014. Vol. 8561 of Lecture Notes in Computer Science. Springer, pp. 58–74.

Maslowski, D., Wijsen, J., 2013. A dichotomy in the complexity of counting database repairs. Journal of Computer and System Sciences 79 (6), 958–983.

McAreavey, K., Liu, W., Miller, P., 2014. Computational approaches to finding and measuring inconsistency in arbitrary knowledge bases. International Journal of Approximate Reasoning 55, 1659–1693.

Mertens, F., 1874. Über einige asymptotische Gesetze der Zahlentheorie. Journal für die reine und angewandte Mathematik 77, 289–338.

Morgado, A., Heras, F., Liffiton, M. H., Planes, J., Marques-Silva, J., 2013. Iterative and core-guided MaxSAT solving: A survey and assessment. Constraints 18 (4), 478–534.

Mu, K., Liu, W., Jin, Z., Bell, D., 2011. A Syntax-based Approach to Measuring the Degree of Inconsistency for Belief Bases. International Journal of Approximate Reasoning 52 (7), 978–999.

Mu, K., Wang, K., Wen, L., 2014. Approaches to measuring inconsistency for stratified knowledge bases. International Journal of Approximate Reasoning 55, 529–556.

Papadimitriou, C. H., 1994. Computational complexity. Addison-Wesley.

Papadimitriou, C. H., Wolfe, D., 1988. The complexity of facets resolved. Journal of Computer and System Sciences 37 (1), 2–13.

Parisi, F., Grant, J., 2017. On repairing and querying inconsistent probabilistic spatio-temporal databases. International Journal of Approximate Reasoning 84, 41–74.

Peñaloza, R., Sertkaya, B., 2017. Understanding the complexity of axiom pinpointing in lightweight description logics. Artificial Intelligence 250, 80–104.

Pfandler, A., Rümmele, S., Wallner, J. P., Woltran, S., 2015. On the parameterized complexity of belief revision. In: Yang, Q., Wooldridge, M. J. (Eds.), Proceedings of IJCAI 2015. AAAI Press, pp. 3149–3155.

Potyka, N., 2014. Linear programs for measuring inconsistency in probabilistic logics. In: Baral, C., Giacomo, G. D., Eiter, T. (Eds.), Proceeedings of KR 2014. AAAI Press, pp. 568–577.

Pretolani, D., Sep 1996. Hierarchies of polynomially solvable satisfiability problems. Annals of Mathematics and Artificial Intelligence 17 (2), 339–357.

Priest, G., 1979. Logic of Paradox. Journal of Philosophical Logic 8, 219–241.

Raddaoui, B., 2015. Computing inconsistency using logical argumentation. In: Loiseau, S., Filipe, J., Duval, B., van den Herik, H. J. (Eds.), Proceedings of ICAART 2015. SciTePress, pp. 164–172.

Roussel, O., Manquinho, V. M., 2009. Pseudo-Boolean and cardinality constraints. In: Biere, A., Heule, M., van Maaren, H., Walsh, T. (Eds.), Handbook of Satisfiability. IOS Press, pp. 695–733.

Staworko, S., Chomicki, J., 2010. Consistent query answers in the presence of universal constraints. Information Systems 35 (1), 1–22.

Thimm, M., 2013. Inconsistency measures for probabilistic logics. Artificial Intelligence 197, 1–24.

Thimm, M., 2016a. On the expressivity of inconsistency measures. Artificial Intelligence 234, 120–151.

Thimm, M., 2016b. Stream-based inconsistency measurement. International Journal of Approximate Reasoning 68, 68–87.

Thimm, M., 2017a. Measuring inconsistency with many-valued logics. International Journal of Approximate Reasoning 86, 1–23.

Thimm, M., 2017b. On the compliance of rationality postulates for inconsistency measures: A more or less complete picture. Künstliche Intelligenz 31 (1), 37–39.

Thimm, M., 2017c. The tweety library collection for logical aspects of artificial intelligence and knowledge representation. Künstliche Intelligenz 31 (1), 93–97.

Thimm, M., February 2018. On the evaluation of inconsistency measures. In: Grant, J., Martinez, M. V. (Eds.), Measuring Inconsistency in Information. Vol. 73 of Studies in Logic. College Publications.

Thimm, M., Wallner, J. P., 2016. Some complexity results on inconsistency measurement. In: Baral, C., Delgrande, J. P., Wolter, F. (Eds.), Proceedings of KR 2016. AAAI Press, pp. 114–124.

Toda, S., 1989. On the computational power of PP and +P. In: Proceedings of FOCS 1989. IEEE Computer Society, pp. 514–519.

Toda, S., Watanabe, O., 1992. Polynomial time 1-Turing reductions from #PH to #P. Theoretical Computer Science 100 (1), 205–221.

Ulbricht, M., Thimm, M., Brewka, G., 2016. Measuring inconsistency in answer set programs. In: Michael, L., Kakas, A. C. (Eds.), Proceedings of JELIA 2016. Vol. 10021 of Lecture Notes in Computer Science. pp. 577–583.

Ulbricht, M., Thimm, M., Brewka, G., February 2018a. Inconsistency measures for disjunctive logic programs under answer set semantics. In: Grant, J., Martinez, M. V. (Eds.), Measuring Inconsistency in Information. Vol. 73 of Studies in Logic. College Publications.

Ulbricht, M., Thimm, M., Brewka, G., 2018b. Measuring strong inconsistency. In: McIlraith, S. A., Weinberger, K. Q. (Eds.), Proceedings of AAAI 2018. AAAI Press, pp. 1989–1996.

Valiant, L. G., 1979a. The complexity of computing the permanent. Theoretical Computer Science 8, 189–201.

Valiant, L. G., 1979b. The complexity of enumeration and reliability problems. SIAM Journal on Computing 8 (3), 410–421.

Wagner, K. W., 1986. The complexity of combinatorial problems with succinct input representation. Acta Informatica 23 (3), 325–356.

Wijsen, J., 2014. A survey of the data complexity of consistent query answering under key constraints. In: Beierle, C., Meghini, C. (Eds.), Proceedings of FoIKS 2014. Vol. 8367 of Lecture Notes in Computer Science. Springer, pp. 62–78.

Xiao, G., Ma, Y., 2012. Inconsistency measurement based on variables in minimal unsatisfiable subsets. In: Raedt, L. D., Bessière, C., Dubois, D., Doherty, P., Frasconi, P., Heintz, F., Lucas, P. J. F. (Eds.), Proceedings of ECAI 2012. Vol. 242 of Frontiers in Artificial Intelligence and Applications. IOS Press, pp. 864–869.

Zhou, L., Huang, H., Qi, G., Ma, Y., Huang, Z., Qu, Y., 2009. Measuring inconsistency in dl-lite ontologies. In: Proceedings of Web Intelligence. IEEE Computer Society, pp. 349–356.

Appendix A. Proofs of Technical Results

Lemma 1. *For* $\mathcal{I} \in \{\mathcal{I}_d, \mathcal{I}_c, \mathcal{I}_p, \mathcal{I}_{hs}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{\max}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{mv}, \mathcal{I}_{nc}, \mathcal{I}_{mcsc}, \mathcal{I}_{forget},$ $\mathcal{I}_{CC}\}$ *there is* $k \in \mathbb{N}$ *such that* $|\mathcal{C}_\mathcal{I}(n)| \in O(n^k)$. *Moreover, each* $x \in \mathcal{C}_\mathcal{I}(n)$ *can be represented in space* $O(\log n)$.

Proof.

- For $\mathcal{I}_d$ we trivially have $|\mathcal{C}_{\mathcal{I}_d}(n)| \in O(1)$ as $\mathcal{I}_d$ only has two different values. These two values can also be represented in constant space.

- For $\mathcal{I}_c$ observe that a knowledge base $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ cannot mention more than n different propositions. Therefore $\mathcal{I}_c(\mathcal{K}) \leq n$ for every $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ and $|\mathcal{C}_{\mathcal{I}_c}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_p$ note that $\mathcal{I}_p(\mathcal{K}) \leq |\mathcal{K}| \leq \mathtt{len}(\mathcal{K}) \leq n$ and therefore $|\mathcal{C}_{\mathcal{I}_p}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_{hs}$, observe that in the worst-case when every formula of $\mathcal{K}$ is pairwise inconsistent with each other, there is an H with $|H| = |\mathcal{K}|$ that is a minimal hitting set of $\mathcal{K}$. So either $\mathcal{I}_{hs}(\mathcal{K}) \leq |\mathcal{K}| \leq n$ or $\mathcal{I}_{hs}(\mathcal{K}) = \infty$ (if there is a contradictory formula) and therefore $|\mathcal{C}_{\mathcal{I}_{hs}}(n)| \in O(n)$.

72

Using constant space for the special number ∞ and using binary representations for the others, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_{\mathrm{dalal}}^{\Sigma}$ observe that both the number of formulas and the number of propositions in $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ are bounded by n. Then $d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) \leq n$ for every $\alpha \in \mathcal{K}$ and $\omega \in \Omega(\mathsf{At})$ and therefore

$$\mathcal{I}_{\mathrm{dalal}}^{\Sigma}(\mathcal{K}) = \min\{\sum_{\alpha \in \mathcal{K}} d_{\mathrm{d}}(\mathsf{Mod}(\alpha), \omega) \mid \omega \in \Omega(\mathsf{At})\}$$
$$\leq \sum_{\alpha \in \mathcal{K}} n \leq n^2$$

and hence $|\mathcal{C}_{\mathcal{I}_{\mathrm{dalal}}^{\Sigma}}(n)| \in O(n^2)$. It also follows $|\mathcal{C}_{\mathcal{I}_{\mathrm{dalal}}^{\max}}(n)| \in O(n)$ and $|\mathcal{C}_{\mathcal{I}_{\mathrm{dalal}}^{\mathrm{hit}}}(n)| \in O(n)$.

- For $\mathcal{I}_{mv}$ observe that $|\bigcup_{M \in \mathsf{MI}(\mathcal{K})} \mathsf{At}(M)| \leq n$ for $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ as $\mathcal{K}$ cannot mention more than n propositions. It follows $|\mathcal{C}_{\mathcal{I}_{mv}}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_{nc}$ observe $\mathcal{I}_{nc}(\mathcal{K}) \in \{0, \ldots, |\mathcal{K}|\}$ and therefore $|\mathcal{C}_{\mathcal{I}_{nc}}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_{mcsc}$, it holds that the intersection of any set of subsets ranges from 0 to $|\mathcal{K}|$ and therefore $|\mathcal{C}_{\mathcal{I}_{mcsc}}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_{\mathrm{forget}}$ observe that a knowledge base $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ cannot mention more than n different propositions. Therefore $\mathcal{I}_{\mathrm{forget}}(\mathcal{K}) \leq n$ for every $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ and $|\mathcal{C}_{\mathcal{I}_{\mathrm{forget}}}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$.

- For $\mathcal{I}_{CC}$, note that a set $S = \{K_1, \ldots, K_m\}$ can only be a conditional independent MUS partition if the K_i are pair-wise disjoint, $i = 1, \ldots, m$, and therefore $|S| \leq |\mathcal{K}|$ and we have $|\mathcal{C}_{\mathcal{I}_{CC}}(n)| \in O(n)$. Using binary representations, each of these numbers only needs space $O(\log n)$. $\quad\square$

Lemma 2. *For* $\mathcal{I} \in \{\mathcal{I}_{\mathsf{MI}}, \mathcal{I}_{\eta}, \mathcal{I}_{mc}\}$ *there is* $k \in \mathbb{N}$ *such that* $|\mathcal{C}_{\mathcal{I}}(n)| \in O(2^{n^k})$. *Moreover, each* $x \in \mathcal{C}_{\mathcal{I}}(n)$ *can be represented in space* $O(n^k)$.

Proof.

- For $\mathcal{I}_{\mathsf{MI}}$, note that the number of minimal inconsistent subsets is bounded from above by the number of all subsets, which is $2^{|\mathcal{K}|}$. So we have $|\mathcal{C}_{\mathcal{I}_{\mathsf{MI}}}(n)| \in O(2^n)$.[8] Using binary representations, each of these numbers needs space $O(n)$.

- In (Thimm, 2016a) it has been shown that

$$|\mathcal{C}_{\mathcal{I}_{\eta}}(n)| \leq \Phi\left(\binom{n}{\lfloor n/2 \rfloor}\right)$$

  where $\Phi(x)$ is the number of fractions in the Farey series if order x.[9] More formally[10]

$$\Phi(x) = |\{k/l \mid l = 1, \ldots, x, k = 0, \ldots, l\}|$$

  While there is no closed form for $\Phi(x)$ its asymptotic behaviour is polynomial, more concretely, $\Phi(x) \sim 3x^2/\pi^2 + O(n \log n)$ (Mertens, 1874; Graham et al., 1994) and therefore $|\mathcal{C}_{\mathcal{I}_{\eta}}(n)| \in O(2^{n^2})$.

  Using binary representations, each $y \in \Phi(x)$ can be represented by two numbers in space $O(\log x)$ (one for the nominator and one for the denominator). Therefore, each number in $\mathcal{C}_{\mathcal{I}_{\eta}}(n)$ can be represented in $O(n^2)$.

- For $\mathcal{I}_{mc}$, note that the number of maximal consistent subsets is bounded from above by the number of all subsets, which is $2^{|\mathcal{K}|}$. So we have $|\mathcal{C}_{\mathcal{I}_{mc}}(n)| \in O(2^n)$. Using binary representations, each of these numbers needs space $O(n)$. □

---

[8]Note that a smaller upper bound for the maximal number of minimal inconsistent subsets is $\binom{n}{\lfloor n/2 \rfloor}$, which is, however, still exponential in n, cf. (Thimm, 2016a)

[9]Actually, a stronger result has been shown in (Thimm, 2016a), namely that the right-hand side of the above equation is an upper bound for the number of attainable inconsistency values on knowledge bases with at most n formulas

[10]See also `http://oeis.org/A005728`

Lemma 3. *For $\mathcal{I} \in \{\mathcal{I}_{\mathsf{MI^c}}, \mathcal{I}_{is}\}$, $|\mathcal{C}_{\mathcal{I}}(n)| \in O(2^{2^n})$. Moreover, each $x \in \mathcal{C}_{\mathcal{I}}(n)$ can be represented in space $O(n^k)$.*

Proof. For $\mathcal{I}_{\mathsf{MI^c}}$ it has been shown in (Thimm, 2016a) that

$$|\mathcal{C}_{\mathcal{I}_\eta}(n)| \leq \Psi(n)$$

where $\Psi(n)$ is the number of monotone Boolean functions of n variables.[11] Clearly, $\Psi(n) \leq 2^{2^n}$ as the latter number is the number of all Boolean function of n variables and therefore $|\mathcal{C}_{\mathcal{I}_{\mathsf{MI^c}}}(n)| \in O(2^{2^n})$.

Let $|\mathcal{K}| = n$. For representing the number for this inconsistency measure, notice that

$$\Sigma_{M \in \mathsf{MI}(\mathcal{K})} \frac{1}{|M|} = \Sigma_{M \in \mathsf{MI}(\mathcal{K}), |M|=1} \frac{1}{|M|} + \cdots + \Sigma_{M \in \mathsf{MI}(\mathcal{K}), |M|=n} \frac{1}{|M|}$$

where each sum is bounded by $\frac{2^n}{i}$ for $1 \leq i \leq n$. Computing the sum can be done, naively, by

$$\Sigma_{M \in \mathsf{MI}(\mathcal{K}), |M|=1} \frac{1 \cdot \left(\frac{n!}{1}\right)}{|M| \cdot \left(\frac{n!}{1}\right)} + \cdots + \Sigma_{M \in \mathsf{MI}(\mathcal{K}), |M|=n} \frac{1 \cdot \left(\frac{n!}{n}\right)}{|M| \cdot \left(\frac{n!}{1}\right)}$$

which equals

$$\Sigma_{M \in \mathsf{MI}(\mathcal{K}), |M|=1} \frac{1 \cdot \left(\frac{n!}{1}\right)}{n!} + \cdots + \Sigma_{M \in \mathsf{MI}(\mathcal{K}), |M|=n} \frac{1 \cdot \left(\frac{n!}{n}\right)}{n!}$$

and each term is bounded from above by $\frac{2^n \cdot n^n}{n!} \leq \frac{n^{2 \cdot n}}{n^n}$. The latter term's rational number can be represented by two integers with polynomial space (wrt. n).

For $\mathcal{I}_{is}$, observe that in the extreme case, every subset of a set of sets may be an independent set. As the number of minimal inconsistent subsets is bounded by 2^n, the number of independent sets is therefore bounded by 2^{2^n} and we have $|\mathcal{C}_{\mathcal{I}_{is}}(n)| \in O(2^{2^n})$.

For representing the number of this inconsistency measure, analogous reasoning leads to the result: each individual independent set can contain a formula of $\mathcal{K}$ at most once, thus has at most $|\mathcal{K}|$ many sets (also, the independent set is bounded polynomially in size). The sum of independent sets of sizes $1, \ldots, |\mathcal{K}|$ yields the overall result; again each sum is bounded by the maximum number of MIs ($2^{|\mathcal{K}|}$).

$\square$

---

[11]See also `http://oeis.org/A220880`

Lemma 4. *Let $\mathcal{I}$ be some inconsistency measure and $i > 0$ an integer. If $\text{UPPER}_{\mathcal{I}}$ is in Σ_i^p or in Π_i^p, the range $\mathcal{C}_{\mathcal{I}}(n)$ is representable in space $O(\log f(n))$ and $|\mathcal{C}_{\mathcal{I}}(n)| \in O(f(n))$ for some function $f : \mathbb{N} \to \mathbb{N}$, then $\text{VALUE}_{\mathcal{I}}$ is in $\mathsf{FP}^{\Sigma_i^p[\log f(n)]}$.*

Proof. This follows directly from the fact that binary search has logarithmic runtime. $\square$

Lemma 5. *The measures $\mathcal{I}_d, \mathcal{I}_{\mathsf{MI}}, \mathcal{I}_\eta, \mathcal{I}_c, \mathcal{I}_{mc}, \mathcal{I}_p, \mathcal{I}_{hs}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{\max}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{mv}, \mathcal{I}_{nc}, \mathcal{I}_{mcsc}, \mathcal{I}_{forget}, \mathcal{I}_{CC}, \mathcal{I}_{is}$ are well-serializable.*

Proof. The measures $\mathcal{I}_d, \mathcal{I}_{\mathsf{MI}}, \mathcal{I}_c, \mathcal{I}_{mc}, \mathcal{I}_p, \mathcal{I}_{hs}, \mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{\max}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{mv}, \mathcal{I}_{nc}, \mathcal{I}_{mcsc}, \mathcal{I}_{forget}, \mathcal{I}_{CC}$ are integer-valued, so immediate successors and predecessor can be determined by adding or subtracting one, respectively.

Regarding $\mathcal{I}_\eta$, it has been noted in (Thimm, 2016a) that the set of attainable values of $\mathcal{I}_\eta$ on knowledge bases $\mathcal{K}$ which mention at most n propositions is the set

$$S = \{k/l \mid l = 1, \ldots, 2^n, k = 0, \ldots, l\}$$

Observe that successors and predecessors within S can be computed in polynomial time (the set has exponential size in n, so binary search needs polynomial time in n). Now recall that every knowledge base $\mathcal{K}$ with $\mathtt{len}(\mathcal{K}) \leq n$ mentions at most n propositions, which shows that $\mathcal{I}_\eta$ is well-serializable.

Finally, we consider $\mathcal{I}_{is}$. Let $n \in \mathbb{N}$ and $x \in \mathcal{C}_{\mathcal{I}_{is}}(n)$. Then $y = \log(e^x + 1)$ is the immediate successor of x in $\mathcal{C}_{\mathcal{I}_{is}}(n)$. Analogously, the immediate predecessor of x is $z = \log(e^x - 1)$. Both expressions can be evaluated in polynomial time. $\square$

Lemma 6. *Let $\mathcal{I}$ be some well-serializable inconsistency measure and $i > 0$ an integer. Let $C \in \{\Sigma_i^p, \Pi_i^p\}$.*

- *$\text{UPPER}_{\mathcal{I}}$ is C-complete iff $\text{LOWER}_{\mathcal{I}}$ is co-C-complete;*

- *if $\text{UPPER}_{\mathcal{I}}$ or $\text{LOWER}_{\mathcal{I}}$ is in C, then $\text{EXACT}_{\mathcal{I}}$ is in D_i^p.*

Proof. Let $\mathcal{K}$ and x be an instance of $\text{UPPER}_{\mathcal{I}}$. This is a yes instance iff $\mathcal{K}$ together with y—where y is the immediate successor of x in $\mathcal{C}_{\mathcal{I}}(n)$—is a no instance of $\text{LOWER}_{\mathcal{I}}$. If $\text{UPPER}_{\mathcal{I}}$ is Σ_i^p-complete (Π_i^p-complete) then $\text{LOWER}_{\mathcal{I}}$ is Π_i^p-complete (Σ_i^p-complete). For the second item note that $\mathcal{I}(\mathcal{K}) = x$ holds iff $\mathcal{I}(\mathcal{K}) \geq x$ and $\mathcal{I}(\mathcal{K}) \leq x$. $\square$

Proposition 1. $\text{UPPER}_{\mathcal{I}_d}$ *is NP-complete.*

Proof. Note that $\mathcal{I}_d(\mathcal{K}) = 0$ if and only if $\mathcal{K}$ is consistent. Therefore, $\text{UPPER}_{\mathcal{I}_d}$ with $x = 1$ is trivial and with $x = 0$ is equivalent to SAT and therefore NP-complete. $\square$

Proposition 5. This proposition lists several results for many measures; we go through each individually. For each measure we restate the formal statements first and proceed to the proofs.

$\text{UPPER}_{\mathcal{I}_\eta}$ *is NP-complete,* $\text{LOWER}_{\mathcal{I}_\eta}$ *is coNP-complete,* $\text{EXACT}_{\mathcal{I}_\eta}$ *is D_1^p-complete, and* $\text{VALUE}_{\mathcal{I}_\eta}$ *is in $\mathsf{FP}^{\mathsf{NP}[n]}$.*

Proof. Note that the problem to compute $\mathcal{I}_\eta(\mathcal{K})$ can be represented as a linear program over an exponential number of variables (the possible worlds) and a linear number of equalities and inequalities (Knight, 2002). Any solution to this problem is nonnegative and due to the small-model-property of linear programs (Chvátal, 1983), there is a solution where only a polynomial number of variables receive a non-zero value. We can therefore guess a set of polynomial many variables, set the objective function to the given upper bound x, and solve the corresponding program using a polynomial-time algorithm (as linear programming is in P). If it is feasible, x is indeed an upper bound. Completeness for NP follows from the fact that we can reduce SAT to $\text{UPPER}_{\mathcal{I}_\eta}$ with $x = 0$.

Except D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_\eta}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_\eta}$ by reducing the problem SAT-UNSAT to $\text{EXACT}_{\mathcal{I}_\eta}$ via the reduction $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, a, \neg a \vee \psi\}$. Note that if (ϕ, ψ) is a positive instance of SAT-UNSAT then (and only then) $\mathcal{I}_\eta(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1/2$ (the two formulas a and $\neg a \vee \psi$ are mutually exclusive as ψ is unsatisfiable, so any probability function P satisfies $P(a) + P(\neg a \vee \psi) = 1$; if ϕ is satisfiable, having $P(\phi) \geq 0.5$ is consistently possible). If ϕ is unsatisfiable, we have $\mathcal{I}_\eta(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$ as $P(\phi) = 0$ for every probability function P. If both ϕ and ψ are satisfiable, $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ is consistent and we have $\mathcal{I}_\eta(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 0$. $\square$

$\text{UPPER}_{\mathcal{I}_{dalal}^{hit}}$ *is NP-complete,* $\text{LOWER}_{\mathcal{I}_{dalal}^{hit}}$ *is coNP-complete,* $\text{EXACT}_{\mathcal{I}_{dalal}^{hit}}$ *is D_1^p-complete, and* $\text{VALUE}_{\mathcal{I}_{dalal}^{hit}}$ *is $\mathsf{FP}^{\mathsf{NP}[\log n]}$-complete.*

Proof. Hardness for $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ follows since we can reduce SAT to $\text{UP-PER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ if we set bound $x = 0$.

Further, $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ is in NP, since we can non-deterministically guess a $\omega \in \Omega(\mathsf{At})$ for a given $\mathcal{K}$, and verify that the d_{d}-distance is 0 for at least $|\mathcal{K}| - \lfloor x \rfloor$ many elements in $\mathcal{K}$, wrt. ω, for a given real x. This implies that $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ is NP-complete.

Except D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ and $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness of $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ by reducing the problem $\mathsf{SAT\text{-}UNSAT}$ to $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ via the reduction $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, \phi_{\text{copy}}, a, \neg a \vee \psi\}$. In particular, note that if (ϕ, ψ) is a positive instance of $\mathsf{SAT\text{-}UNSAT}$ then $\mathcal{I}_{\text{dalal}}^{\text{hit}}(R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$. If ϕ is unsatisfiable and ψ is satisfiable, $\mathcal{I}_{\text{dalal}}^{\text{hit}}(R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 2$. If ϕ is unsatisfiable and ψ is unsatisfiable, $\mathcal{I}_{\text{dalal}}^{\text{hit}}(R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 3$. Finally, if both ϕ and ψ are satisfiable then $\mathcal{I}_{\text{dalal}}^{\text{hit}}(R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 0$.

Regarding $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness of $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$, let $\phi_s = \{c_1, \ldots, c_n\}$ be an instance of $\mathsf{MaxSATSize}$. We show that for $\mathcal{K} = \phi_s$ we have $\mathcal{I}_{\text{dalal}}^{\text{hit}}(\mathcal{K}) = n - k$ with k the maximum number of clauses that can be simultaneously satisfied in ϕ_s, i.e. k is the solution to ϕ_s in the $\mathsf{MaxSATSize}$ problem.

$$n - \max\{|C| \mid C \subseteq \mathcal{K}, \bigwedge_{c \in C} \not\models \bot\}$$
$$= n - \max\{|\{\alpha \in \mathcal{K} \mid d_{\text{d}}(\mathsf{Mod}(\alpha), \omega) = 0\}| \mid \omega \in \Omega(\mathsf{At})\}$$
$$= \min\{|\{\alpha \in \mathcal{K} \mid d_{\text{d}}(\mathsf{Mod}(\alpha), \omega) > 0\}| \mid \omega \in \Omega(\mathsf{At})\}$$
$$= \mathcal{I}_{\text{dalal}}^{\text{hit}}(\mathcal{K})$$

Thus, we have reduced $\mathsf{MaxSATSize}$ to $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}$ showing $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness for the latter. $\qquad\square$

$\text{UPPER}_{\mathcal{I}_{dalal}^{\Sigma}}$ *is* NP*-complete,* $\text{LOWER}_{\mathcal{I}_{dalal}^{\Sigma}}$ *is* coNP*-complete,* $\text{EXACT}_{\mathcal{I}_{dalal}^{\Sigma}}$ *is* D_1^p*-complete, and* $\text{VALUE}_{\mathcal{I}_{dalal}^{\Sigma}}$ *is* $\mathsf{FP}^{\mathsf{NP}[\log n]}$*-complete.*

Proof. NP-membership of $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ follows from considering the following algorithm. For $\mathcal{K} = \{\alpha_1, \ldots, \alpha_n\}$ guess $(\omega_0, \omega_1, \ldots, \omega_n) \in \Omega(\mathsf{At})^{n+1}$. For each α_i, $i = 1, \ldots, n$, check whether $\omega_i \models \alpha_i$ (this is a polynomial test). Then compute $x = \sum_{i=1,\ldots,n} d_{\text{d}}(\omega_i, \omega_0)$, also in polynomial time. It follows that x is an upper bound for $\mathcal{I}_{\text{dalal}}^{\Sigma}(\mathcal{K})$. NP-hardness follows from the fact that SAT can be reduced to $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ with $x = 0$.

Except D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ and $\text{FP}^{\text{NP}[\log n]}$-hardness of $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ by reducing the problem SAT-UNSAT to $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ via the reduction $R_{\text{SAT-UNSAT}}(\phi, \psi) = \{\phi, a, \neg a \vee \psi\}$. In particular, note that if (ϕ, ψ) is a positive instance of SAT-UNSAT then $\mathcal{I}_{\text{dalal}}^{\Sigma}(R_{\text{SAT-UNSAT}}(\phi, \psi)) = 1$. If ϕ is unsatisfiable, $\mathcal{I}_{\text{dalal}}^{\Sigma}(R_{\text{SAT-UNSAT}}(\phi, \psi)) = \infty$. Finally, if both ϕ and ψ are satisfiable then $\mathcal{I}_{\text{dalal}}^{\Sigma}(R_{\text{SAT-UNSAT}}(\phi, \psi)) = 0$.

We show $\text{FP}^{\text{NP}[\log n]}$-hardness of $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ by reduction from MaxSATSize. Let $\phi_s = \{c_1, \ldots, c_n\}$ be an instance of MaxSATSize with ϕ_s over variables $\{x_1, \ldots, x_m\}$. We construct $\mathcal{K} = \{\alpha_1, \alpha_2\}$ with $\alpha_1 = \bigwedge_{1 \leq i \leq n}(c_i \vee (\neg y_i))$ and $\alpha_2 = \bigwedge_{1 \leq i \leq n} y_i$ with fresh variables y_i. We now show that for k the maximum number of clauses in ϕ_s that can be simultaneously satisfied, $n - k = \mathcal{I}_{\text{dalal}}^{\Sigma}(\mathcal{K})$. First, we prove that $\forall \omega \in \Omega(\text{At})$

$$d_{\text{d}}(\text{Mod}(\alpha_1), \omega) + d_{\text{d}}(\text{Mod}(\alpha_2), \omega) \geq n - k. \tag{A.1}$$

Define shorthands $d_{\text{d}}(\text{Mod}(\alpha_1), \omega) = a(\omega)$ and $d_{\text{d}}(\text{Mod}(\alpha_2), \omega) = b(\omega)$. Suppose the contrary, i.e. $\exists \omega \in \Omega(\text{At}), a(\omega) + b(\omega) < n - k$. The interpretation ω assigns $b(\omega)$ many y_i variables to false and $n - b(\omega)$ many to true. By presumption, there must exist a model ω_1 of α_1 s.t. $d_{\text{d}}(\omega_1, \omega) < n - k - b(\omega)$. Consider now the maximum number c of y_i variables that ω_1 assigns to false. Under the previous constraint on the symmetric difference, ω_1 can assign all y_i variables to false that are also assigned to false by ω ($b(\omega)$ many), and additionally less than $n - k - b(\omega)$ (remainder of symmetric difference). Thus we can bound c by $c < b(\omega) + n - k - b(\omega)$ and in turn by $c < n - k$. This implies that ω_1 satisfies at least $n - c$ clauses of ϕ_s, i.e. strictly more than k clauses, a contradiction. Thus, Equation (A.1) holds. There always exists an $\omega_2 \in \Omega(\text{At})$ that assigns all y_i to true s.t. Equation A.1 holds with equality. This implies our claim, i.e. $n - k = \mathcal{I}_{\text{dalal}}^{\Sigma}(\mathcal{K})$. We conclude that $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\Sigma}}$ is $\text{FP}^{\text{NP}[\log n]}$-complete. $\square$

$\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\max}}$ *is* NP-*complete,* $\text{LOWER}_{\mathcal{I}_{\text{dalal}}^{\max}}$ *is* coNP-*complete,* $\text{EXACT}_{\mathcal{I}_{\text{dalal}}^{\max}}$ *is* D_1^p-*complete, and* $\text{VALUE}_{\mathcal{I}_{\text{dalal}}^{\max}}$ *is in* $\text{FP}^{\text{NP}[\log n]}$.

Proof. NP-membership of $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\max}}$ follows from considering the following algorithm. For $\mathcal{K} = \{\alpha_1, \ldots, \alpha_n\}$ guess $(\omega_0, \omega_1, \ldots, \omega_n) \in \Omega(\text{At})^{n+1}$. For each α_i, $i = 1, \ldots, n$, check whether $\omega_i \models \alpha_i$ (this is a polynomial test). Then compute $x = \max_{i=1,\ldots,n} d_{\text{d}}(\omega_i, \omega_0)$, also in polynomial time. It holds that x

is an upper bound for $\mathcal{I}_{\mathrm{dalal}}^{\max}(\mathcal{K})$. NP-hardness follows from the fact that SAT can be reduced to $\mathrm{UPPER}_{\mathcal{I}_{\mathrm{dalal}}^{\max}}$ with $x = 0$.

Except D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_{\mathrm{dalal}}^{\max}}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_{\mathrm{dalal}}^{\max}}$ by reducing the problem SAT-UNSAT to $\mathrm{EXACT}_{\mathcal{I}_{\mathrm{dalal}}^{\max}}$ via the reduction $R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, a, \neg a \vee \psi\}$. In particular, note that if (ϕ, ψ) is a positive instance of SAT-UNSAT then $\mathcal{I}_{\mathrm{dalal}}^{\max}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 1$. If ϕ is unsatisfiable, $\mathcal{I}_{\mathrm{dalal}}^{\max}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = \infty$. Finally, if both ϕ and ψ are satisfiable then $\mathcal{I}_{\mathrm{dalal}}^{\max}(R_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)) = 0$. $\qquad\square$

$\mathrm{UPPER}_{\mathcal{I}_{forget}}$ is *NP-complete*, $\mathrm{LOWER}_{\mathcal{I}_{forget}}$ is *coNP-complete*, $\mathrm{EXACT}_{\mathcal{I}_{forget}}$ is D_1^p-complete, and $\mathrm{VALUE}_{\mathcal{I}_{forget}}$ is $\mathsf{FP}^{\mathsf{NP}[\log n]}$-complete.

Proof. For NP-containment of $\mathrm{UPPER}_{\mathcal{I}_{\mathrm{forget}}}$, consider the following non-deterministic algorithm:

1. Non-deterministically guess triples $(a_1, i_1, \phi_1), \ldots, (a_x, i_x, \phi_x)$ with $a_j \in$ At, $i_j \in \mathbb{N}$, $\phi_j \in \{\bot, \top\}$, $j = 1, \ldots, x$.
2. Non-deterministically guess an interpretation ω.
3. Replace occurrences of propositions in $\bigwedge \mathcal{K}$ as prescribed by the triples guessed in step 1.
4. Check whether ω satisfies the new formula (return TRUE iff this is the case).

Note that verifying whether an interpretation satisfies a formula can be done in polynomial time. Therefore, $\mathrm{UPPER}_{\mathcal{I}_{\mathrm{forget}}} \in$ NP. NP-completeness follows from the fact that SAT can be reduced to $\mathrm{UPPER}_{\mathcal{I}_{\mathrm{forget}}}$ with $x = 0$.

Except D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_{\mathrm{forget}}}$ and $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness of $\mathrm{VALUE}_{\mathcal{I}_{\mathrm{forget}}}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_{\mathrm{forget}}}$ by reducing the problem SAT-UNSAT to $\mathrm{EXACT}_{\mathcal{I}_{\mathrm{forget}}}$ via the reduction $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi) = \{\phi, \phi_{\mathrm{copy}}, a, \neg a \vee \psi\}$. In particular, note that if (ϕ, ψ) is a positive instance of SAT-UNSAT then $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ can be made consistent by replacing the first occurrence of a by $\top$. If ϕ is unsatisfiable then at least two propositions have to be replaced in ϕ and ϕ_{copy}. Finally, if both ϕ and ψ are satisfiable then $R'_{\mathsf{SAT\text{-}UNSAT}}(\phi, \psi)$ is consistent.

For $\mathsf{FP}^{\mathsf{NP}[\log n]}$-hardness of $\mathrm{VALUE}_{\mathcal{I}_{\mathrm{forget}}}$, we provide a reduction of MaxSATSize to $\mathrm{VALUE}_{\mathcal{I}_{\mathrm{forget}}}$. Let $\phi = \{c_1, \ldots, c_n\}$ be an instance of MaxSATSize. Observe

that forgetting a variable in a clause c_i, $i = 1, \ldots, n$, renders this clause tautological. Therefore, if forgetting k variables in ϕ suffices to make ϕ consistent then $n - k$ clauses are jointly satisfiable. So for sets of clauses the problem $\text{VALUE}_{\mathcal{I}_{\text{forget}}}$ is the complement of the problem MaxSATSize, i.e., the solution to MaxSATSize is $n - \mathcal{I}_{\text{forget}}(\phi)$. $\qquad\square$

$\text{UPPER}_{\mathcal{I}_{mcsc}}$ *is NP-complete,* $\text{LOWER}_{\mathcal{I}_{mcsc}}$ *is coNP-complete,* $\text{EXACT}_{\mathcal{I}_{mcsc}}$ *is D_1^p-complete, and* $\text{VALUE}_{\mathcal{I}_{mcsc}}$ *is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.*

Proof. Before we give the proofs we first have to make some general observations on MC-covers and $\mathcal{I}_{mcsc}$. Recall that a set of maximal consistent subsets $\mathcal{C} \subseteq \mathsf{MC}(\mathcal{K})$ is called an MC-cover if

$$\bigcup_{C \in \mathcal{C}} C = \mathcal{K} \tag{A.2}$$

and it is normal if no proper subset of $\mathcal{C}$ is an MC-cover. A normal MC-cover is maximal if

$$\lambda(\mathcal{C}) = |\bigcap_{C \in \mathcal{C}} C|$$

is maximal. Then $\mathcal{I}_{mcsc}(\mathcal{K}) = |\mathcal{K}| - \lambda(\mathcal{C})$ for any maximal MC-cover $\mathcal{C}$. How many elements can a normal MC-cover $\mathcal{C}$ contain? An MC-cover is only normal if there is no $C \in \mathcal{C}$ such that all formulas of C are not contained in some other set $C' \in \mathcal{C}$. Therefore, every $C \in \mathcal{C}$ has to contain at least one formula that is not contained in any other $C' \in \mathcal{C}$. In order to have (A.2) we have $|\mathcal{C}| \in \{1, \ldots, |\mathcal{K}|\}$. In the following, we are devising an algorithm for $\text{UPPER}_{\mathcal{I}_{mcsc}}$ so we can relax several notions in the definition of $\mathcal{I}_{mcsc}$. First, an MC-cover is supposed to contain *maximal* consistent sets. But note that any set of consistent sets that cover $\mathcal{K}$ can be extended to an MC-cover as every consistent set has a maximal consistent superset. By doing so, the number of subsets can only decrease (if consistent sets become the same when taking their maximal consistent supersets) and the size of the intersection of these sets—i.e., the value $\lambda(\mathcal{C})$—can only increase (as more formulas are added to possibly each set), therefore yielding a smaller value of $\mathcal{I}_{mcsc}$. Therefore, any set of (not necessarily maximal) consistent subsets yields an upper bound for the true value of $\mathcal{I}_{mcsc}(\mathcal{K})$. Second, for the same reason considering non-maximal MC-covers—or more specifically, non-maximal sets of consistent subsets that cover $\mathcal{K}$—also yields an upper bound for the true value if $\mathcal{I}_{mcsc}$.

Now let $(\mathcal{K}, x)$ be an instance of $\text{UPPER}_{\mathcal{I}_{mcsc}}$ and consider the following non-deterministic algorithm:

1. Non-deterministically guess a number $m \in \{1, \ldots, |\mathcal{K}|\}$, sets $C_1, \ldots, C_m \subseteq \mathcal{K}$, and interpretations $\omega_1, \ldots, \omega_m$.
2. For each $i = 1, \ldots, m$, if $\omega_i \not\models C_i$ return FALSE.
3. If not $C_1 \cup \ldots \cup C_m = \mathcal{K}$ return FALSE.
4. If $x \geq |\mathcal{K}| - |C_1 \cap \ldots \cap C_m|$ return TRUE, otherwise return FALSE.

First observe that the above algorithm runs in non-deterministic polynomial time. In step 1.) only a constant (m), a linear number of sets (of linear size each) and a linear number of interpretations (each of linear size) are guessed. Verifying $\omega_i \models C_i$ in step 2.) is polynomial. The remaining computations in steps 3.) and 4.) are obviously polynomial as well.

If the algorithms terminates with TRUE it means that there is a set $\mathcal{C} = \{C_1, \ldots, C_m\}$ of consistent sets (step 2.) that cover $\mathcal{K}$ (step 3.) such that $x \geq |\mathcal{K}| - \lambda(\mathcal{C})$ (step 4.). From the discussion from before if follows that $\mathcal{I}_{mcsc}(\mathcal{K}) \leq |\mathcal{K}| - \lambda(\mathcal{C})$ as $\mathcal{C}$ may contain non-maximal consistent sets and $\mathcal{C}$ is not necessarily maximal. However, the algorithm solves $\text{UPPER}_{\mathcal{I}_{mcsc}}$.

NP-completeness follows from the fact that SAT can be reduced to $\text{UPPER}_{\mathcal{I}_{mcsc}}$ with $x = 0$.

Except D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_{mcsc}}$, all other results follow from Lemmas 4 and 6.

We show D_1^p-hardness of $\text{EXACT}_{\mathcal{I}_{mcsc}}$ by reducing the problem SAT-UNSAT to $\text{EXACT}_{\mathcal{I}_{mcsc}}$ via the reduction $R_{\text{SAT-UNSAT}}(\phi, \psi) = \{\phi, a, \neg a \vee \psi\}$. In particular, note that if (ϕ, ψ) is a positive instance of SAT-UNSAT then $\mathcal{I}_{mcsc}(R_{\text{SAT-UNSAT}}(\phi, \psi)) = 1$. If ϕ is unsatisfiable, $\mathcal{I}_{mcsc}(R_{\text{SAT-UNSAT}}(\phi, \psi)) = \infty$ (as there is no MC-cover). Finally, if both ϕ and ψ are satisfiable then $\mathcal{I}_{mcsc}(R_{\text{SAT-UNSAT}}(\phi, \psi)) = 0$. $\qquad\square$

Proposition 6. *ChromaticNumber can be polynomially reduced to* $\text{VALUE}_{\mathcal{I}_{hs}}$.

Proof. Let $G = (V, E)$ be an undirected graph and let $Adj(v) = \{w \mid \{v, w\} \in E\}$ be the set of neighbours of $v \in V$. We identify each vertex $v \in V$ with an atom and write

$$\phi_v = v \wedge \neg w_1 \wedge \ldots \wedge \neg w_n$$

where $\{w_1, \ldots, w_n\} = Adj(v)$. Define a knowledge base $R_{\text{ChromaticNumber}}(G)$ via

$$R_{\text{ChromaticNumber}}(G) = \{\phi_v \mid v \in V\}$$

82

Observe that $R_{\mathsf{ChromaticNumber}}(G)$ can be constructed in polynomial time. We claim $\gamma(G) = \mathcal{I}_{hs}(R_{\mathsf{ChromaticNumber}}(G)) + 1$, which provides a polynomial reduction from $\mathsf{ChromaticNumber}$ to $\mathrm{VALUE}_{\mathcal{I}_{hs}}$.

To show the claim, let $\gamma(G) = k$ and let $col : V \to \{1, \dots, k\}$ be a corresponding colouring. Then for $i = 1, \dots, k$ the set

$$K_i = \{\phi_v \mid col(v) = i\} \subseteq R_{\mathsf{ChromaticNumber}}(G)$$

is consistent (it could only be inconsistent iff it contains ϕ_v and $\phi_{v'}$ of two adjacent nodes). Let ω_i be an interpretation with $\omega_i \models K_i$, for $i = 1, \dots, k$. Then $\{\omega_1, \dots, \omega_k\}$ is a hitting set of $R_{\mathsf{ChromaticNumber}}(G)$ and we have $\gamma(G) \geq \mathcal{I}_{hs}(R_{\mathsf{ChromaticNumber}}(G)) + 1$. The other direction is analogous. $\qquad\square$

Proposition 9. $\mathrm{UPPER}_{\mathcal{I}_p}$ *is* Π_2^p-*complete,* $\mathrm{LOWER}_{\mathcal{I}_p}$ *is* Σ_2^p-*complete,* $\mathrm{EXACT}_{\mathcal{I}_p}$ *is* D_2^p-*complete., and* $\mathrm{VALUE}_{\mathcal{I}_p}$ *is in* $\mathsf{FP}^{\Sigma_2^p[\log n]}$.

Proof. Let the knowledge base $\mathcal{K}$ together with x be an arbitrary instance of $\mathrm{LOWER}_{\mathcal{I}_p}$. Eiter and Gottlob (1992) have shown that checking whether some $\phi \in \mathcal{K}$ is contained in any MI is in Σ_2^p. For membership of $\mathrm{LOWER}_{\mathcal{I}_p}$ in Σ_2^p, we guess $\mathcal{K}' \subseteq \mathcal{K}$ with $|\mathcal{K}'| = \lfloor x \rfloor$ and use the non-deterministic algorithm utilizing a coNP oracle given by (Eiter and Gottlob, 1992) to verify that each $\phi \in \mathcal{K}'$ is contained in an MI of $\mathcal{K}$.

For hardness, we utilize a similar, but simpler, reduction as in Proposition 8. Let $\phi = \exists X \forall Y \psi$ be a QBF with $X = \{x_1, \dots, x_n\}$. Construct $\mathcal{K} = \bigcup_{1 \leq i \leq n} \{x_i, \neg x_i\} \cup \{\neg\psi\}$. We now claim that ϕ is true iff $\mathcal{I}_p(\mathcal{K}) = 2 \cdot n + 1 = |\mathcal{K}|$. That is, $2 \cdot n + 1$ is a lower bound iff ϕ is true. It is immediate that for any ϕ we have $\mathcal{I}_p(\mathcal{K}) \geq 2 \cdot n$, since for any i with $1 \leq i \leq n$ it holds that $\{x_i, \neg x_i\}$ is an MI. It holds that

$$
\begin{aligned}
&\mathcal{I}(\mathcal{K}) \geq 2 \cdot n + 1 \\
&\text{iff } \exists M \in \mathsf{MI}(\mathcal{K}) \text{ s.t. } \neg\psi \in M \\
&\text{iff } \exists \mathcal{K}' \subseteq \mathcal{K} \text{ s.t.} \\
&\qquad \mathcal{K}' \subseteq (\mathcal{K} \setminus \{\neg\psi\}) \not\models \bot \text{ and } \mathcal{K}' \cup \{\neg\psi\} \models \bot \\
&\text{iff } \exists \omega \text{ defined on } X \text{ s.t. } \neg\psi[\omega] \models \bot \\
&\text{iff } \phi \text{ is true.}
\end{aligned}
$$

Except D_2^p-hardness of $\mathrm{EXACT}_{\mathcal{I}_p}$, all other results follow from Lemmas 4 and 6.

To show D_2^p-hardness of $\textsc{Exact}_{\mathcal{I}_p}$, we reduce $\forall\exists\mathsf{QBF}_2$ to it. Let $\sigma = (\exists X \forall Y \phi, \forall Y' \exists X' \psi)$ be a pair of QBFs. We assume that ϕ and ψ do not share any atoms. Let $\forall Y'' \exists X'' \psi'$ be an exact copy of $\forall Y' \exists X' \psi$ with fresh atoms not appearing in either ϕ or ψ. Consider the following construction

$$\mathcal{K}_\sigma = \bigcup_{x \in X} \{x, \neg x\} \cup \{\neg \phi\} \cup \bigcup_{y' \in Y'} \{y', \neg y'\} \cup \{\psi\} \cup \bigcup_{y'' \in Y''} \{y'', \neg y''\} \cup \{\psi'\}$$

We claim that σ is a positive instance of $\forall\exists\mathsf{QBF}_2$, i.e., both QBFs in σ evaluate to true, if and only if $\mathcal{I}_p(\mathcal{K}_\sigma) = 2|X| + 4|Y'| + 1$. This shows that $\forall\exists\mathsf{QBF}_2$ can be reduced to $\textsc{Exact}_{\mathcal{I}_p}$.

In order to show the claim, recall that the proof of Proposition 9 already established that $\exists X \forall Y \phi$ evaluates to true if and only if

$$\mathcal{I}_p(\bigcup_{x \in X} \{x, \neg x\} \cup \{\neg \phi\}) = 2|X| + 1$$

If $\exists X \forall Y \phi$ evaluates to false the above becomes $2|X|$. Similarly, observe that $\forall Y' \exists X' \psi$ evaluates to true if and only if

$$\mathcal{I}_p(\bigcup_{y' \in Y'} \{y', \neg y'\} \cup \{\psi\}) = 2|Y'|$$

If $\forall Y' \exists X' \psi$ evaluates to false the above becomes $2|Y'| + 1$. As ϕ, ψ, ψ' do not share any atoms, respectively, we have

$$\mathcal{I}_p(\mathcal{K}_\sigma) = \mathcal{I}_p(\bigcup_{x \in X} \{x, \neg x\} \cup \{\neg \phi\}) + 2 * \mathcal{I}_p(\bigcup_{y' \in Y'} \{y', \neg y'\} \cup \{\psi\})$$

It follows, if σ is a positive instance then $\mathcal{I}_p(\mathcal{K}_\sigma) = 2|X| + 4|Y'| + 1$. If $\exists X \forall Y \phi$ evaluates to false and $\forall Y' \exists X' \psi$ evaluates to true then $\mathcal{I}_p(\mathcal{K}_\sigma) = 2|X| + 4|Y'|$. If $\exists X \forall Y \phi$ evaluates to true and $\forall Y' \exists X' \psi$ evaluates to false then $\mathcal{I}_p(\mathcal{K}_\sigma) = 2|X| + 4|Y'| + 3$. Finally, if $\exists X \forall Y \phi$ evaluates to false and $\forall Y' \exists X' \psi$ evaluates to false then $\mathcal{I}_p(\mathcal{K}_\sigma) = 2|X| + 4|Y'| + 2$. $\square$

Proposition 11. $\textsc{Lower}_{\mathcal{I}_{CC}}$ *is in* Σ_3^p.

Proof. Using the the insight from Proposition 10 we can devise an algorithm for $\textsc{Lower}_{\mathcal{I}_{CC}}$ that runs in $\Sigma_3^p = \mathsf{NP}^{\Sigma_2^p} = \mathsf{NP}^{\Pi_2^p}$. Let $(\mathcal{K}, x)$ be an instance of $\textsc{Lower}_{\mathcal{I}_{CC}}$.

1. Guess $\{K_1, \ldots, K_n\}$
2. Use Π_2^p-oracle to verify that $\{K_1, \ldots, K_n\}$ is a conditional independent MUS partition of $\mathcal{K}$
3. If $n \geq x$ return TRUE, otherwise return FALSE

Note that the above algorithm is polynomial as only sets $\{K_1, \ldots, K_n\}$ with $n \leq |\mathcal{K}|$ have to be considered as each pair must be disjoint. $\qquad\square$

Proposition 12. LOWER$_{\mathcal{I}_{\mathsf{MI}}}$ and UPPER$_{\mathcal{I}_{\mathsf{MI}}}$ are CNP-complete, EXACT$_{\mathcal{I}_{\mathsf{MI}}}$ is C$_=$NP-hard.

Proof. Membership for LOWER$_{\mathcal{I}_{\mathsf{MI}}}$ follows from the fact that there is a problem in $B \in \mathsf{D}^p$ s.t. $(\mathcal{K}, M)$ is a yes instance iff M is a MI of $\mathcal{K}$. For a given instance of LOWER$_{\mathcal{I}_{\mathsf{MI}}}$, i.e., a knowledge base $\mathcal{K}$ and a number k (which we round up to the next integer), $(\mathcal{K}, k)$ is a yes instance of LOWER$_{\mathcal{I}_{\mathsf{MI}}}$ iff there are at least k MI's of $\mathcal{K}$ iff $\mathsf{C}^k_{M \in 2^{\mathcal{K}}}(\mathcal{K}, M) \in B$. It holds that the size of any subset of $\mathcal{K}$ is polynomially bounded by the size of $\mathcal{K}$. Thus, LOWER$_{\mathcal{I}_{\mathsf{MI}}}$ is in CNP (recall that CNP = CD^P). Membership for UPPER$_{\mathcal{I}_{\mathsf{MI}}}$ holds since an instance $(\mathcal{K}, k)$ is a yes instance of UPPER$_{\mathcal{I}_{\mathsf{MI}}}$ iff $(\mathcal{K}, k+1)$ is a no instance of LOWER$_{\mathcal{I}_{\mathsf{MI}}}$. The latter can be characterized as a no instance of a problem in CNP. It holds that CNP is closed under complementation, i.e., co-CNP = CNP (Wagner, 1986, Theorem 4).

Hardness for LOWER$_{\mathcal{I}_{\mathsf{MI}}}$ follows from the reduction $R_{\#\mathsf{MI}}(\chi) = \mathcal{K}$ for the CNP-complete problem of deciding whether there are at least k many models of $\chi = \forall Y \phi(X, Y)$. By Lemma 7, it holds that $|\mathsf{Mod}(\chi)| = |\mathsf{MI}(\mathcal{K})| - |X|$. This implies that $|\mathsf{Mod}(\chi)| \geq k$ iff $|\mathsf{MI}(\mathcal{K})| - |X| \geq k$ iff $|\mathsf{MI}(\mathcal{K})| \geq k + |X|$. Thus, $(\mathcal{K}, k + |X|)$ is a yes instance of LOWER$_{\mathcal{I}_{\mathsf{MI}}}$ iff $|\mathsf{Mod}(\chi)| \geq k$ iff (χ, k) is a yes instance of the problem of deciding whether there are at least k models of χ.

Regarding C$_=$NP-hardness of EXACT$_{\mathcal{I}_{\mathsf{MI}}}$, for a given instance of the C$_=$NP-complete problem of deciding whether a given open QBF of form $\chi = \forall Y \phi(X, Y)$ has exactly k models, it follows, by Lemma 7, that $|\mathsf{Mod}(\chi)| = |\mathsf{MI}(\mathcal{K})| - |X|$ with $R_{\#\mathsf{MI}}(\forall Y \phi(X, Y)) = \mathcal{K}$. Thus, (χ, k) is a yes instance of the problem of deciding whether χ has exactly k models iff $|\mathsf{MI}(\mathcal{K})| = k + |X|$ iff $(\mathcal{K}, k + |X|)$ is a yes instance of EXACT$_{\mathcal{I}_{\mathsf{MI}}}$. $\qquad\square$

Proposition 13. VALUE$_{\mathcal{I}_{\mathsf{MI}}}$ is #·coNP-complete under subtractive reductions.

Proof. Regarding membership, we use the fact that $\#\cdot\mathsf{coNP} = \#\cdot\Delta_2^{\mathsf{p}}$ (Hemaspaandra and Vollmer, 1995, Theorem 1.5) and further that it holds that verifying whether a given subset is an MI is a D_1^p-complete problem (Papadimitriou and Wolfe, 1988). This means $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}}}$ is in $\#\cdot\mathsf{coNP}$, since MI is the witness function producing finite subsets for a given knowledge base, all such subsets are polynomially bounded in size of the given knowledge base, and checking whether such a set is indeed an MI is in Δ_2^p.

For hardness, let $\chi(X) = \forall Y \phi(X, Y)$ with $X = \{x_1, \ldots, x_n\}$ be an arbitrary instance of the $\#\cdot\mathsf{coNP}$-complete problem $\#\,\Pi_1\mathsf{SAT}$. In this problem we have to compute the number of assignments on X that satisfy χ, which contains also variables over set Y.

By Lemma 7, it holds that $|\mathsf{Mod}(\chi)| = |\mathsf{MI}(\mathcal{K})| - |X|$ with $R_{\#\mathsf{MI}}(\forall Y \phi(X, Y)) = \mathcal{K}$. Let $\mathcal{K}' = \bigcup_{1 \le i \le n}\{\phi_i\}$ and $\overline{\mathcal{K}'} = \bigcup_{1 \le i \le n}\{\overline{\phi_i}\}$. By construction, it follows that $\mathsf{MI}(\mathcal{K}' \cup \overline{\mathcal{K}'}) \subseteq \mathsf{MI}(\mathcal{K})$, due to monotonicity of MI, i.e. if $\mathcal{K}_1 \subseteq \mathcal{K}_2$ then $\mathsf{MI}(\mathcal{K}_1) \subseteq \mathsf{MI}(\mathcal{K}_2)$. It holds that $\mathsf{MI}(\mathcal{K}' \cup \overline{\mathcal{K}'}) = |X|$. To see this, consider $M \in \mathsf{MI}(\mathcal{K}' \cup \overline{\mathcal{K}'})$. If there is an integer i with $1 \le i \le |X|$ s.t. neither ϕ_i nor $\overline{\phi_i}$ is in M, then M is consistent. If both ϕ_i and $\overline{\phi_i}$ are in M for some i, then for all $j \ne i$ it must hold that exactly one of $\{\phi_j, \overline{\phi_j}\}$ must be in M, otherwise M would either be not minimal or consistent. Thus, there are exactly $|X|$ minimal inconsistent subsets of $\mathcal{K}' \cup \overline{\mathcal{K}'}$.

The preceding results and lemma imply that $|\mathsf{Mod}(\chi)| = |\mathsf{MI}(\mathcal{K})| - |\mathsf{MI}(\mathcal{K}' \cup \overline{\mathcal{K}'})|$ and $\mathsf{MI}(\mathcal{K}) \supseteq \mathsf{MI}(\mathcal{K}' \cup \overline{\mathcal{K}'})$, i.e., we have a subtractive reduction from $\#\,\Pi_1\mathsf{SAT}$ to $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}}}$. $\qquad\square$

Proposition 14. $\mathrm{LOWER}_{\mathcal{I}_{\mathsf{MIC}}}$ *and* $\mathrm{UPPER}_{\mathcal{I}_{\mathsf{MIC}}}$ *are both* CNP*-hard problems, and* $\mathrm{EXACT}_{\mathcal{I}_{\mathsf{MIC}}}$ *is a* $\mathsf{C}_=\mathsf{NP}$ *hard problem.*

Proof. We reduce from the CNP-complete problem of deciding whether a given open QBF of the form $\chi = \forall Y \phi(X, Y)$ has at least k models, for a given integer k. Let $\mathcal{K} = R_{\#\mathsf{MI}}(\chi)$ be the constructed knowledge base from reduction $R_{\#\mathsf{MI}}$. By Lemma 7, it follows that $\mathsf{Mod}(\chi) = |\mathsf{MI}(\mathcal{K})| - |X|$. From the proof of that lemma, it also follows that $\forall M \in \mathsf{MI}(\mathcal{K}) : |M| = |X| + 1$.

$$\mathsf{Mod}(\chi) = \underbrace{|\mathsf{MI}(\mathcal{K})|}_{\forall M \in \mathsf{MI}(\mathcal{K}):|M|=|X|+1} - |X|, \text{ and}$$

$$\mathcal{I}_{\mathsf{MIC}}(\mathcal{K}) = \sum_{M \in \mathsf{MI}(\mathcal{K})} \left(\frac{1}{|X| + 1}\right)$$

By straightforward manipulations we get

$$\frac{\mathsf{Mod}(\chi)}{|X|+1} = \frac{|\mathsf{MI}(\mathcal{K})|}{|X|+1} - \frac{|X|}{|X|+1}$$

$$\frac{\mathsf{Mod}(\chi)}{|X|+1} = \mathcal{I}_{\mathsf{MI}^c}(\mathcal{K}) - \frac{|X|}{|X|+1}$$

$$\mathcal{I}_{\mathsf{MI}^c}(\mathcal{K}) = \frac{\mathsf{Mod}(\chi)}{|X|+1} + \frac{|X|}{|X|+1} = \frac{\mathsf{Mod}(\chi)+|X|}{|X|+1}$$

This implies that that (χ, k) is a yes instance of the problem of deciding whether χ has at least k models iff $\mathsf{Mod}(\chi) \geq k$ iff $\mathcal{I}_{\mathsf{MI}^c}(\mathcal{K}) \geq \frac{k+|X|}{|X|+1}$.

Analogously, we can reduce the problem of deciding whether χ has strictly less than k models to $\mathcal{I}_{\mathsf{MI}^c}(\mathcal{K}) \leq \frac{(k-1)+|X|}{|X|+1}$ (with the case $k=0$ being trivial).

Finally, hardness for $\mathrm{EXACT}_{\mathcal{I}_{\mathsf{MI}^c}}$ holds, if we ask whether $\mathcal{I}_{\mathsf{MI}^c}(\mathcal{K}) = \frac{k+|X|}{|X|+1}$. □

Proposition 15. $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}^c}}$ *is in* $FP^{\#\cdot coNP}$.

Proof. Let $\mathcal{K}$ be an arbitrary instance of $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}^c}}$. Consider the following algorithm.

1. For each $x \in \{1, \ldots, |\mathcal{K}|\}$: compute $\mathcal{M}_x = |\{M | M \in \mathsf{MI}(\mathcal{K}), |M| = x\}|$
2. Compute $\mathcal{M} = \sum_{1 \leq i \leq |\mathcal{K}|} \frac{1}{i} \cdot \mathcal{M}_i$

It holds that $\mathcal{M} = \mathcal{I}_{\mathsf{MI}^c}(\mathcal{K})$ (each $M \in \mathsf{MI}(\mathcal{K})$ has size between 1 and $\mathcal{K}$). Each step 1 of the loop can be computed via a functional oracle call $\#\cdot\mathsf{coNP}$, i.e., the problem of asking for the number of MI's of size exactly x from a given $\mathcal{K}$ is in $\#\cdot\mathsf{coNP}$. Algorithm runs in linear time (if oracle calls require constant time), since each size (from which there are linearly many) the oracle is called once. Finally, the generated number can be represented in polynomial space (see Lemma 3). □

Lemma 8. *Let* $\chi = \forall Y \phi(X, Y)$ *be an open QBF with* X *and* Y *sets of variables. For* $\mathcal{K} = R_{\#\mathsf{MC}}(\forall Y \phi(X, Y))$ *it holds that* $|\mathsf{Mod}(\chi)| = 2^{|X|} - |\mathsf{MC}(\mathcal{K})| + 1$.

Proof. Assume $M \in \mathsf{MC}(\mathcal{K})$. Consider case distinction (i) ψ is not in M or (ii) ψ is in M. For (i), then $M = \bigcup_{1 \leq i \leq |X|} \{\phi_i, \overline{\phi_i}\}$ (M is maximal). Assume now (ii), i.e., $\psi \in M$. Then for any i with $1 \leq i \leq |X|$ it holds that not both ϕ_i and $\overline{\phi_i}$ are in M (otherwise M would be unsatisfiable). By assumption,

it holds that there is a truth assignment that satisfies M, and in particular the truth assignment satisfies $\neg\phi(X, Y)$. This truth assignment assigns a value to each variable in X. If the assigned value is true for an $x \in X$, then $\phi_i \in M$, otherwise $\overline{\phi_i} \in M$ (otherwise, M is not maximal; either ϕ_i or $\overline{\phi_i}$ can be added to M without violating satisfiability). Thus, for any i with $1 \leq i \leq |X|$ it holds that $|\{\phi_i, \overline{\phi_i}\} \cap M| = 1$, i.e., exactly one of each ϕ_i or $\overline{\phi_i}$ is in M.

For an interpretation f let $f|_X$ be the restriction of f to the function assigning only values to variables in X and to these exactly those as assigned by f. We now show that if $M \in \mathsf{MC}(\mathcal{K})$ is of form (ii), then the truth assignment f defined by $f = f'|_X$ for any model f' of M (which exists by assumption and all models assign the same values to variables in X) is not in $\mathsf{Mod}(\chi)$, i.e., $f \notin \mathsf{Mod}(\chi)$. It holds that $f' \models M$, $f' \models \psi$, and $f' \models \neg\phi(X, Y)$. Thus, f is not a model of χ since f can be extended (e.g. to f') s.t. $\phi(X, Y)$ is not satisfied.

Assume now that an interpretation f defined on X is not in $\mathsf{Mod}(\chi)$. Then define M with

$$M = \{\phi \mid f(x) = \mathsf{true}\} \cup \{\overline{\phi} \mid f(x) = \mathsf{false}\} \cup \{\psi\}.$$

It holds that $M \in \mathsf{MC}(\mathcal{K})$ and M is of form (ii). To see that M is consistent, it holds that f is not a model of χ, and, thus, there is an extension of f to all the variables in Y, call it f', s.t. $f' \not\models \phi(X, Y)$. By construction of $\mathcal{K}$, this means that $f' \models M$. Maximality of M is straightforward: any addition of a formula in $\mathcal{K}$ renders M unsatisfiable (note that $\psi \in M$ and for each $1 \leq i \leq |X|$ one of ϕ_i or $\overline{\phi_i}$ is in M).

Summarizing, we have a bijection between $2^X \setminus \mathsf{Mod}(\chi)$ (interpretations that refute χ) and any maximal consistent subset of $\mathcal{K}$ that does contain ψ. Thus $2^{|X|} - |\mathsf{Mod}(\chi)| = |\mathsf{MC}(\mathcal{K})| - 1$ and the claim of the lemma follows (there is exactly one maximal consistent subset of $\mathcal{K}$ which does not contain ψ). □

Proposition 16. $\mathrm{LOWER}_{\mathcal{I}_{mc}}$ and $\mathrm{UPPER}_{\mathcal{I}_{mc}}$ are CNP-complete, $\mathrm{EXACT}_{\mathcal{I}_{mc}}$ is $\mathsf{C}_=$NP-hard.

Proof. CNP-membership claims of the proposition are based on the fact that, for a given arbitrary knowledge base $\mathcal{K}$, verifying whether $M \subseteq \mathcal{K}$ is either (i) a singleton self-contradicting formula, or (ii) a maximal consistent subset of $\mathcal{K}$, are both problems in D^p, and can be compressed into one problem,

say B. For the complexity of B, complexity of verifying whether a set is a maximal consistent subset of $\mathcal{K}$ is in D_1^p, and checking whether (i) holds is in coNP. In the combined problem B, the complexity is in a class of the Boolean hierarchy, more concretely in a class that consists of an intersection of a problem in NP and coNP, and, together, a union with a problem in coNP. For a given instance $(\mathcal{K}, k)$ of $\text{LOWER}_{\mathcal{I}_{mc}}$ it holds that $(\mathcal{K}, k)$ is a yes instance iff $\mathsf{C}_{M \in 2^{\mathcal{K}}}^{k+1}(\mathcal{K}, M) \in B$. Thus, $\text{LOWER}_{\mathcal{I}_{mc}}$ is in CNP (due to (Wagner, 1986, Theorem 4, item 5) it holds that CNP is equal to $\mathsf{CNP} = \mathcal{CB}(\mathsf{NP})$ with $\mathcal{B}(\mathsf{NP})$ being the Boolean closure of NP). For an instance $(\mathcal{K}, k)$ of $\text{UPPER}_{\mathcal{I}_{mc}}$, this is a yes instance iff $(\mathcal{K}, k+1)$ is a no instance of $\text{LOWER}_{\mathcal{I}_{mc}}$, and, therefore, $\text{UPPER}_{\mathcal{I}_{mc}}$ is in CNP (this class is closed under complementation (Wagner, 1986, Theorem 4)).

For hardness, we reduce from the CNP-complete problem of deciding whether there are at least k models of an open QBF of form $\chi = \forall Y \phi(X, Y)$. First, note that this problem remains CNP-complete if restricted to non-tautological formulas ϕ (one can reduce the former problem to the latter via conjoining ϕ with a fresh variable). Assume that ϕ is non-tautological. From instance χ and k, we construct an instance of $\text{UPPER}_{\mathcal{I}_{mc}}$ by applying the reduction $\mathcal{K} = R_{\#\mathsf{MC}}(\forall Y \phi(X, Y))$ defined above. By Lemma 8, it holds that $|\mathsf{Mod}(\chi)| = 2^{|X|} - |\mathsf{MC}(\mathcal{K})| + 1$. Note that $\mathcal{K}$ is free of self-contradicting formulas (each is individually satisfiable by assumption that ϕ is non-tautological). Therefore, it holds that $\mathcal{I}_{mc}(\mathcal{K}) = |\mathsf{MC}(\mathcal{K})| - 1$. Thus, $|\mathsf{Mod}(\chi)| \geq k$ iff $2^{|X|} - |\mathsf{MC}(\mathcal{K})| + 1 \geq k$ iff $2^{|X|} - \mathcal{I}_{mc}(\mathcal{K}) + 2 \geq k$ iff $2^{|X|} + 2 - k \geq \mathcal{I}_{mc}(\mathcal{K})$.

Finally, an instance $(\mathcal{K}, k)$ of $\text{LOWER}_{\mathcal{I}_{mc}}$ is a yes instance iff $(\mathcal{K}, k-1)$ is a no instance of $\text{UPPER}_{\mathcal{I}_{mc}}$. Thus, we reduce the complementary problem of deciding whether there are at least k models of an open QBF of the form above to an instance of $\text{LOWER}_{\mathcal{I}_{mc}}$ to prove hardness for $\text{LOWER}_{\mathcal{I}_{mc}}$.

We show $\mathsf{C}_=\mathsf{NP}$-hardness of $\text{EXACT}_{\mathcal{I}_{mc}}$ as follows. We reduce from the $\mathsf{C}_=\mathsf{NP}$-complete problem of deciding whether an open QBF of form $\chi = \forall Y \phi(X, Y)$ has exactly k models. Similarly as above, we look at QBFs where ϕ is not tautological (this problem remains $\mathsf{C}_=\mathsf{NP}$-complete). Let $\mathcal{K} = R_{\#\mathsf{MC}}(\forall Y \phi(X, Y))$. By Lemma 8, it holds that $|\mathsf{Mod}(\chi)| = 2^{|X|} - |\mathsf{MC}(\mathcal{K})| + 1$. Further, no formula in $\mathcal{K}$ is self-contradicting. Therefore, χ has exactly k models iff $k = 2^{|X|} - |\mathsf{MC}(\mathcal{K})| + 1$ iff $k = 2^{|X|} - \mathcal{I}_{mc}(\mathcal{K}) + 2$ iff $\mathcal{I}_{mc}(\mathcal{K}) = 2^{|X|} - k + 2$. $\qquad \square$

Corollary 4. *#MaxModels is #·coNP-complete via subtractive reductions.*

Proof. Membership follows from the fact that it is in coNP to decide whether

a given interpretation is subset-maximal and satisfies the formula. Within this proof we use the notation w^c for an interpretation w to denote that w and w^c are complementary in their truth assignments, i.e. $w(x) = \textsf{true}$ iff $w^c(x) = \textsf{false}$ and $w(x) = \textsf{false}$ iff $w^c(x) = \textsf{true}$. Let ϕ be an arbitrary instance of #CIRCUMSCRIPTION, i.e. ϕ is a formula in CNF. We construct ϕ' by replacing each literal l in ϕ by its complement, i.e. if $l = p$ is a proposition with $\neg p$ and otherwise if $l = \neg p$ with p. It holds that $w \models \phi$ iff $w^c \models \phi'$. To see this, consider a clause $c \in \phi$ s.t. $w \models c$. Then there is a literal l in c s.t. $w \models l$. This implies that w^c satisfies the complementary literal of l, and thus, w^c models the clause c' where all literals are replaced by their complementary literal. Since this holds for all clauses, we have $w^c \models \phi'$. It then holds that $w \in \{\omega \models \phi \mid \nexists \omega' \text{ s.t. } \omega' <_t \omega\}$ iff $w^c \in \{\omega \models \phi' \mid \nexists \omega' \text{ s.t. } \omega <_t \omega'\}$. □

Proposition 17. *The problem of counting all maximal consistent subsets of a given knowledge base is #·coNP-complete under subtractive reductions.*

Proof. Membership follows from the fact that verifying whether a subset of a knowledge base is a maximal consistent subset is in D_1^p. We show hardness by the following subtractive reduction from #MaxModels (#·coNP-completeness proved in Corollary 4). Let ϕ be an instance of #MaxModels with $\{x_1, \ldots, x_n\}$ the vocabulary of ϕ. Construct

$$\begin{aligned}
\mathcal{K} &= \{(x_i \wedge p \wedge \phi) \mid 1 \leq i \leq n\} \cup \{\neg p\} \cup \{p \wedge \phi\}, \text{ and} \\
\mathcal{K}' &= \{\neg p\}
\end{aligned}$$

with p a fresh variable not occurring in ϕ. We claim that $|\mathsf{MC}(\mathcal{K} \cup \mathcal{K}')| - |\mathsf{MC}(\mathcal{K})|$ is equal to the number of subset maximal models of ϕ.

We distinguish two cases, (i) ϕ is satisfiable and (ii) ϕ is unsatisfiable. Independently of (i) and (ii), it holds that $\mathsf{MC}(\mathcal{K}') = \{\{\neg p\}\}$ and $\{\neg p\} \in \mathsf{MC}(\mathcal{K} \cup \mathcal{K}')$. First, assume (i), i.e., ϕ is satisfiable. For an $M \in \mathsf{MC}(\mathcal{K} \cup \mathcal{K}')$ with $M \neq \{\neg p\}$ it holds that $M \neq \emptyset$ (since $\{\neg p\}$ is always consistent), and that $(p \wedge \phi) \in M$ (M must contain a formula of $(\mathcal{K} \cup \mathcal{K}') \setminus \{\neg p\}$ by assumption and if M contains any other than $(p \wedge \phi)$ then this formula can be added, as

well). For such an M it holds that

$$M \in \mathsf{MC}(\mathcal{K} \cup \mathcal{K}')$$
$$\text{iff } M \not\models \bot \text{ and } \nexists M' \text{ s.t. } M \subset M' \text{ and } M' \not\models \bot$$
$$\text{iff } \phi \wedge \bigwedge_{(x_i \wedge p \wedge \phi) \in M} x_i \not\models \bot \text{ and}$$
$$\phi \wedge (\bigwedge_{(x_i \wedge p \wedge \phi) \in M} x_i) \wedge \bigvee_{(x_i \wedge p \wedge \phi) \in \mathcal{K} \setminus M} x_i \models \bot$$
$$\text{iff } \omega_M \models \phi \text{ with } \omega_M(x_i) = \mathsf{true} \text{ iff } (x_i \wedge p \wedge \phi) \in M \text{ and}$$
$$\omega' \not\models \phi \ \forall \omega' \text{ with } w_M <_t w'.$$

Thus, there is a bijection between the set of maximal models of ϕ and the maximal consistent subsets of $(\mathcal{K} \cup \mathcal{K}')$ which do not contain $\{\neg p\}$. It then holds that $|\mathsf{MC}(\mathcal{K} \cup \mathcal{K}')| - |\mathsf{MC}(\mathcal{K})| = N + 1 - 1$ with N the maximal models of ϕ.

Now assume (ii), i.e., ϕ is unsatisfiable. Then, by construction, we have $\mathsf{MC}(\mathcal{K} \cup \mathcal{K}') = \mathsf{MC}(\mathcal{K}') = 1$, and $|\mathsf{MC}(\mathcal{K} \cup \mathcal{K}')| - |\mathsf{MC}(\mathcal{K})| = 1 - 1 = 0$, which is equal to the number of of maximal models of ϕ (0). $\square$

Proposition 18. *The problem of counting unsatisfiable formulas in a given knowledge base is $\mathsf{FP}^{\mathsf{NP}[\log n]}$-complete.*

Proof. Membership follows from posing logarithmically many queries asking whether in a subset of size k every formula is satisfiable (guessing the set together with interpretations for each). Hardness follows from a reduction from MaxSATSize. Let $\phi = c_1 \wedge \cdots \wedge c_n$ be an arbitrary instance of MaxSATSize. Construct $\mathcal{K} = \{(\mathsf{EXACT}(i, Y) \wedge \phi' \mid 1 \leq i \leq n\}$ with $\phi' = (c_1 \vee y_1) \wedge \cdots \wedge (c_n \vee y_n)$, $Y = \{y_1, \ldots, y_n\}$ fresh variables, and $\mathsf{EXACT}(i, Y)$ a formula that evaluates to true under an assignment iff that assignment assigns exactly i many variables of Y to true. We assume, w.l.o.g., that each $\mathsf{EXACT}(i, Y)$ is built over its own vocabulary. The formula $\mathsf{EXACT}(i, Y)$ can be constructed in polynomial time wrt. the size of Y (Roussel and Manquinho, 2009, Section 22.2.3.). It follows immediately from construction that $(\mathsf{EXACT}(i, Y) \wedge \phi')$ is satisfiable iff i many clauses of ϕ can be satisfied simultaneously. $\square$

Proposition 19. *#MCAndContradictory is #·coNP-complete under subtractive reductions. Hardness holds even for knowledge bases without self-contradictory formulas.*

Proof. Membership follows from the fact that verifying whether a subset of a knowledge base is a maximal consistent subset or a self-contradicting formula is a problem in Δ_2^p. Hardness holds, under subtractive reductions, for the fragment of the problem in the proposition where there are no self-contradictory formulas. To see this, consider an instance $\mathcal{K} = (\phi_1, \ldots, \phi_n)$ of the problem of counting the number of maximal consistent subsets of $\mathcal{K}$. Construct $\mathcal{K}' = \{(\phi_i \vee d) \mid \phi_i \in \mathcal{K}\} \cup \{d, \neg d\}$ and $\mathcal{K}'' = \mathcal{K} \setminus \{\neg d\}$. The single unique maximal consistent subset of $\mathcal{K}''$ is $\mathcal{K}''$ itself (by any model assigning d to true). The same set is also a maximal consistent subset of $\mathcal{K}'$. For any other $M \in \mathsf{MC}(\mathcal{K}')$ s.t. $M \neq \mathcal{K}''$, it holds that $\neg d \in M$. Thus, there must be a model satisfying all ϕ_i's with $(\phi_i \vee d) \in M$. Due to maximality, it holds that $M' = \{\phi_i \mid (\phi_i \vee d) \in M\}$ is a maximal consistent subset of $\mathcal{K}$. Further, for any $M' \in \mathsf{MC}(\mathcal{K})$ it holds that $\{\neg d\} \cup \{(\phi_i \vee d) \mid \phi_i \in M'\}$ is a maximal consistent subset of $\mathcal{K}'$. Thus, $|\mathsf{MC}(\mathcal{K})| = |\mathsf{MC}(\mathcal{K}')| - |\mathsf{MC}(\mathcal{K}'')|$. Finally, any formula in $\mathcal{K}'$ and $\mathcal{K}''$ is (individually) satisfiable. $\square$

Lemma 9. *Let $\mathcal{K}$ be a knowledge base and $R_{\#\mathsf{IS}}(\mathcal{K}) = \mathcal{K}'$. It holds that $|\mathsf{MI}(\mathcal{K})| = |\mathsf{IS}(\mathcal{K}')| + 1$.*

Proof. If $M \in \mathsf{MI}(\mathcal{K})$ with $M = \{\beta_1, \ldots, \beta_k\}$ then $\{\beta_1 \vee p, \ldots, \beta_k \vee p, \neg p\} \in \mathsf{MI}(\mathcal{K}')$, and vice versa, so $|\mathsf{MI}(\mathcal{K})| = |\mathsf{MI}(\mathcal{K}')|$. Furthermore, every singleton set in $\mathsf{MI}(\mathcal{K}')$ is an independent set and every $M \subseteq \mathsf{MI}(\mathcal{K}')$ with $|M| > 1$ is not independent as all minimal inconsistent sets share the formula $\neg p$. As the empty set is also independent we have

$$|\mathsf{MI}(\mathcal{K}')| + 1 = |\{M \subseteq \mathsf{MI}(\mathcal{K}')|M \text{ is independent}\}. \qquad \square$$

Proposition 20. $\mathrm{LOWER}_{\mathcal{I}_{is}}$ *and* $\mathrm{UPPER}_{\mathcal{I}_{is}}$ *are* $\mathsf{C}\mathsf{NP}$*-complete and* $\mathrm{EXACT}_{\mathcal{I}_{is}}$ *is* $\mathsf{C}_{=}\mathsf{NP}$ *hard.*

Proof. For membership of $\mathrm{LOWER}_{\mathcal{I}_{is}}$ and $\mathrm{UPPER}_{\mathcal{I}_{is}}$, assume an arbitrary instance $(\mathcal{K}, k)$ of these problems. Verifying whether a given subset of $\mathcal{K}$ is a MI is a D_1^p-complete problem (Papadimitriou and Wolfe, 1988). Furthermore, checking whether all sets $M_1, \ldots, M_k$ are individually minimal inconsistent for linearly bounded k is D_1^p-complete as well. Note that each independent set's size is polynomially bounded by $|\mathcal{K}|$ (each formula $\alpha \in \mathcal{K}$ can be contained at most once in an independent set due to disjointedness). Finally, verifying whether a set $\{M_1, \ldots, M_k\}$ is independent can be done in polynomial time. Thus, both $\mathrm{LOWER}_{\mathcal{I}_{is}}$ and $\mathrm{UPPER}_{\mathcal{I}_{is}}$ are in CNP (the bound can be extracted straightforwardly from k by applying the inverse of the logarithm).

Hardness for all three problems follows from the reduction $R_{\#\mathsf{IS}}(\mathcal{K}) = \mathcal{K}'$ from the CNP-complete ($\mathsf{C}_{=}\mathsf{NP}$-complete) problem of deciding whether there are at least (exactly) k minimal inconsistent subsets of $\mathcal{K}$. The correspondence has been proven in Lemma 9. □

Proposition 21. *#is is #·coNP-complete via subtractive reductions.*

Proof. Similar as before, recall that $\#\cdot\mathsf{coNP} = \#\cdot\Delta_2^\mathsf{p}$ (Hemaspaandra and Vollmer, 1995, Theorem 1.5) and that verifying whether a given subset is an independent set is in D_1^p (see also proof of Proposition 20). So counting the number of independent subsets of a given knowledge base is in $\#\cdot\mathsf{coNP}$. As mentioned in the proof of Proposition 20, each independent set is polynomial in size: each formula $\alpha \in \mathcal{K}$ can be present at most once in an independent set.

For hardness, we reduce $\mathrm{VALUE}_{\mathcal{I}_{\mathsf{MI}}}$ to the present problem via reduction $R_{\#\mathsf{IS}}(\mathcal{K}) = \mathcal{K}'$ for a given $\mathcal{K}$. By Lemma 9 it holds that $\mathcal{I}_{\mathsf{MI}}(\mathcal{K}) + 1$ is equal to the number of independent sets of $\mathcal{K}'$. The difference of one can be remedied by having another knowledge base with only the empty set being an independent set; the difference of these two constructed knowledge bases is exactly the number of independent sets. □

Proposition 22. *For all $\mathcal{K} \subseteq \mathcal{L}^{CNF}(\mathsf{At})$, $\mathcal{I}_d(\mathcal{K}) = \mathcal{I}_{dalal}^{\max}(\mathcal{K}) = \mathcal{I}_{hs}(\mathcal{K})$.*

Proof. Let $\omega \in \Omega(\mathsf{At})$ be any interpretation and observe that $d_\mathrm{d}(\mathsf{Mod}(\alpha), \omega) \in \{0, 1\}$ for every clause α (either ω satisfies α or only one atom needs to be flipped). It follows $\mathcal{I}_{dalal}^{\max}(\mathcal{K}) \in \{0, 1\}$ for all $\mathcal{K} \in \mathcal{L}^{CNF}(\mathsf{At})$ and therefore $\mathcal{I}_d(\mathcal{K}) = \mathcal{I}_{dalal}^{\max}(\mathcal{K})$. Similarly, observe that every clause is satisfied by either 1.) the interpretation that assigns true to all atoms, or 2.) the interpretation that assigns false to all atoms. It follows $\mathcal{I}_{hs}(\mathcal{K}) \in \{0, 1\}$ for all $\mathcal{K} \in \mathcal{L}^{CNF}(\mathsf{At})$ and therefore $\mathcal{I}_d(\mathcal{K}) = \mathcal{I}_{hs}(\mathcal{K})$ as well. □

Proposition 23. *For all $\mathcal{K} \subseteq \mathcal{L}^{CNF}(\mathsf{At})$, $\mathcal{I}_{dalal}^{\Sigma}(\mathcal{K}) = \mathcal{I}_{dalal}^{hit}(\mathcal{K}) = \mathcal{I}_{forget}(\mathcal{K})$.*

Proof. Let $\mathcal{K} \subseteq \mathcal{L}^{CNF}(\mathsf{At})$. From the proof of Proposition 22, recall that for $\omega \in \Omega(\mathsf{At})$, $d_\mathrm{d}(\mathsf{Mod}(\alpha), \omega) \in \{0, 1\}$ for every clause α (either ω satisfies α or only one atom needs to be flipped). Then $\mathcal{I}_{dalal}^{\Sigma}(\mathcal{K})$ is the minimal number of clauses in $\mathcal{K}$ that are not satisfied by some $\omega \in \Omega(\mathsf{At})$. This establishes $\mathcal{I}_{dalal}^{\Sigma}(\mathcal{K}) = \mathcal{I}_{dalal}^{hit}(\mathcal{K})$. As for $\mathcal{I}_{forget}$, observe that any clause α becomes tautological if any atom in it is forgotten (i. e., replaced by $\top$). Therefore, at most one atom must be forgotten per clause. In fact, if $\mathcal{K}' \subseteq \mathcal{K}$

is a maximal (wrt. cardinality) consistent subset of $\mathcal{K}$ then for each clause in $\mathcal{K} \setminus \mathcal{K}'$, one atom must be forgotten in order to obtain a consistent set. This establishes $\mathcal{I}_{\text{dalal}}^{\Sigma}(\mathcal{K}) = \mathcal{I}_{\text{dalal}}^{\text{hit}}(\mathcal{K}) = \mathcal{I}_{\text{forget}}(\mathcal{K})$. □

Proposition 25. *For* $\mathcal{I} \in \{\mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{forget}\}$, $\text{UPPER}_{\mathcal{I}}^{H}$ *is* NP*-complete,* $\text{LOWER}_{\mathcal{I}}^{H}$ *is* coNP*-complete,* $\text{EXACT}_{\mathcal{I}}^{H}$ *is in* D_1^p, *and* $\text{VALUE}_{\mathcal{I}}^{H}$ *is in* $\text{FP}^{\text{NP}[\log n]}$.

Proof. Due to Proposition 23 we only show the statements wrt. $\mathcal{I}_{\text{dalal}}^{\text{hit}}$. In fact, all results follow from only showing that $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}^{H}$ is NP-hard. For this, we provide a reduction from the problem MaxHornSAT defined as

MaxHornSAT   **Input**: $\mathcal{K} \subseteq \mathcal{L}^{H}(\text{At})$, $K \in \mathbb{N}$
Output: TRUE iff there is $\mathcal{K}' \subseteq \mathcal{K}$ with $|\mathcal{K}'| \geq K$ and $\mathcal{K}' \not\models \bot$

The problem MaxHornSAT is NP-complete (Jaumard and Simeone, 1987) and there is an obvious correspondence between MaxHornSAT and $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}^{H}$: $(\mathcal{K}, K)$ is a positive instance of MaxHornSAT iff $(\mathcal{K}, |\mathcal{K}| - K)$ is positive instance of $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}^{H}$. This completes the proof. □

Proposition 27. *For* $\mathcal{I} \in \{\mathcal{I}_{dalal}^{\Sigma}, \mathcal{I}_{dalal}^{hit}, \mathcal{I}_{forget}\}$, $\text{UPPER}_{\mathcal{I}}^{K}$ *is* NP*-complete,* $\text{LOWER}_{\mathcal{I}}^{K}$ *is* coNP*-complete,* $\text{EXACT}_{\mathcal{I}}^{K}$ *is in* D_1^p, *and* $\text{VALUE}_{\mathcal{I}}^{K}$ *is in* $\text{FP}^{\text{NP}[\log n]}$.

Proof. Due to Proposition 23 we only show the statements wrt. $\mathcal{I}_{\text{dalal}}^{\text{hit}}$. In fact, all results follow from only showing that $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}^{K}$ is NP-hard. For this, we provide a reduction from the problem Max2SAT defined as

Max2SAT   **Input**: $\mathcal{K} \subseteq \mathcal{L}^{K}(\text{At})$, $K \in \mathbb{N}$
Output: TRUE iff there is $\mathcal{K}' \subseteq \mathcal{K}$ with $|\mathcal{K}'| \geq K$ and $\mathcal{K}' \not\models \bot$

The problem Max2SAT is NP-complete (Papadimitriou, 1994) and there is an obvious correspondence between Max2SAT and $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}^{K}$: $(\mathcal{K}, K)$ is a positive instance of Max2SAT iff $(\mathcal{K}, |\mathcal{K}| - K)$ is positive instance of $\text{UPPER}_{\mathcal{I}_{\text{dalal}}^{\text{hit}}}^{K}$. This completes the proof. □

Lemma 10. *Deciding whether a set* $M \subseteq \mathcal{L}^{K}(\text{At})$ *is a minimal inconsistent set is in* P.

Proof. Checking whether M is minimally inconsistent is equivalent to checking whether G_M (see Definition 26) is the union of two shortest paths from some a to $\neg a$ and back. More precisely, it can first be checked in polynomial time whether M is consistent by considering each atom a appearing in M

and checking whether there is a cycle in G_M involving both a and $\neg a$, cf. (Papadimitriou, 1994). Then, for each such found a, we can compute shortest paths from a to $\neg a$ and back in polynomial time and verify whether all edges in M have been used for this. If not, M is not a minimal inconsistent set. $\qquad\square$

Proposition 28. *For $X \in \{H, K\}$, $\mathrm{LOWER}^X_{\mathcal{I}_p}$ is in NP, $\mathrm{UPPER}^X_{\mathcal{I}_p}$ is in coNP, $\mathrm{EXACT}^X_{\mathcal{I}_p}$ is in D^p_1, and $\mathrm{VALUE}^X_{\mathcal{I}_p}$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$*

Proof. Consider the problem $\mathrm{LOWER}^H_{\mathcal{I}_p}$ and the following non-deterministic algorithm for input $\mathcal{K} \subseteq \mathcal{L}^H(\mathsf{At})$, $x \in \mathbb{R}^\infty_{\geq 0}$. Without loss of generality assume $x \in \mathbb{N} \setminus \{0\}$ (as $\mathcal{I}_p$ is integer-valued) and $x \leq |\mathcal{K}|$ (in the extreme case all formulas of $\mathcal{K}$ are in a minimal inconsistent set).

1. For each $i = 1, \ldots, x$ guess a set $M_i \subseteq \mathcal{K}$ (not necessarily distinct).
2. Using the algorithm from (Davydov et al., 1998) verify that each M_i is minimally inconsistent, $i = 1, \ldots, x$.
3. If $|\bigcup^x_{i=1} M_i| \geq x$ return true, otherwise return false.

It should be clear that the above algorithm solves $\mathrm{LOWER}^H_{\mathcal{I}_p}$ and, due to (Davydov et al., 1998), runs in polynomial time.

The remaining statements follow using Lemma 10 and similar reasoning as in Lemmas 4 and 6. $\qquad\square$

Proposition 29. *For $X \in \{H, K\}$, $\mathrm{LOWER}^X_{\mathcal{I}_{mv}}$ is in NP, $\mathrm{UPPER}^X_{\mathcal{I}_{mv}}$ is in coNP, $\mathrm{EXACT}^X_{\mathcal{I}_{mv}}$ is in D^p_1, and $\mathrm{VALUE}^X_{\mathcal{I}_{mv}}$ is in $\mathsf{FP}^{\mathsf{NP}[\log n]}$.*

Proof. Consider the problem $\mathrm{LOWER}^H_{\mathcal{I}_{mv}}$ and the following non-deterministic algorithm for input $\mathcal{K} \subseteq \mathcal{L}^H(\mathsf{At})$, $x \in \mathbb{R}^\infty_{\geq 0}$. Without loss of generality assume $x = y/|\mathsf{At}|$ with $y \in \mathbb{N} \setminus \{0\}$ and $y \leq |\mathsf{At}|$ (in the extreme case all atoms of $\mathcal{K}$ are mentioned in a minimal inconsistent set).

1. For each $i = 1, \ldots, y$ guess a set $M_i \subseteq \mathcal{K}$ (not necessarily distinct).
2. Using the algorithm from (Davydov et al., 1998) verify that each M_i is minimally inconsistent, $i = 1, \ldots, x$.
3. If $|\bigcup^x_{i=1} \mathsf{At}(M_i)| \geq y$ return true, otherwise return false.

It should be clear that the above algorithm solves $\mathrm{LOWER}^H_{\mathcal{I}_{mv}}$ and, due to (Davydov et al., 1998), runs in polynomial time.

The remaining statements follow using Lemma 10 and similar reasoning as in Lemmas 4 and 6. $\qquad\square$

Proposition 30. *For $X \in \{H, K\}$, $\text{LOWER}_{\mathcal{I}_{nc}}^X$ is in* NP, $\text{UPPER}_{\mathcal{I}_{nc}}^X$ *is in* coNP, $\text{EXACT}_{\mathcal{I}_{nc}}^X$ *is in* D_1^p, *and* $\text{VALUE}_{\mathcal{I}_{nc}}^X$ *is in* $\mathsf{FP}^{\mathsf{NP}[\log n]}$.

Proof. Membership of $\text{LOWER}_{\mathcal{I}_{nc}}^X$ follows directly from the proof of Proposition 8: checking whether a set $\mathcal{K}' \subseteq \mathcal{K}$ is consistent can be done in polynomial time in the Horn and Krom fragments. The remaining statements follow using a similar reasoning as in Lemmas 4 and 6. □

Proposition 31. *For $X \in \{H, K\}$, $\text{LOWER}_{\mathcal{I}_{CC}}^X$ is in Σ_2^p, $\text{UPPER}_{\mathcal{I}_{CC}}^X$ is in Π_2^p, $\text{EXACT}_{\mathcal{I}_{CC}}^X$ is in D_2^p, and $\text{VALUE}_{\mathcal{I}_{CC}}^X$ is in $\mathsf{FP}^{\Sigma_2^p[\log n]}$.*

Proof. Recall the problem CODECMUSPART from the proof of Proposition 9 and consider its restriction to Horn logic CODECMUSPART(H):

CODECMUSPART(H)   **Input**: $\mathcal{K} \subseteq \mathcal{L}^H(\mathsf{At})$, $\{K_1, \ldots, K_n\}$
                              Output: TRUE iff $\{K_1, \ldots, K_n\}$ is not a cond.
                                               independent MUS partition of $\mathcal{K}$

Observe that the same algorithm showing Σ_2^p-membership of the general problem CODECMUSPART shows NP-membership of CODECMUSPART(H) as step 6 becomes polynomially solvable due to (Davydov et al., 1998). It follows that $\text{LOWER}_{\mathcal{I}_{CC}}^H$ is in Σ_2^p and the other results as in Proposition 11 and Corollary 3.

The remaining statements follow using Lemma 10 and similar reasoning as in Lemmas 4 and 6. □

Proposition 32. *For $X \in \{H, K\}$, $\text{VALUE}_{\mathcal{I}_{MI}}^X$ is #P-complete under Turing reductions.*

Proof. For hardness, we recall the result of (Peñaloza and Sertkaya, 2017, Corollary 42), and present a proof directly adapted to our setting. Consider a directed graph $G = (V, E)$ with $s, t \in V$ (source and target). We give a parsimonious reduction from the problem of counting all simple paths in G from s to t to the problem of counting minimal unsatisfiable subsets in the Horn and Krom knowledge base $\mathcal{K} = \{s, \neg t\} \cup \{(\neg x \vee y) \mid (x, y) \in E\}$. Let $M \subseteq \mathcal{K}$ be an unsatisfiable subset (not necessarily minimal). It holds that one can reach from s the vertex t via the edge set $E' \subseteq E$ corresponding to M, by $E' = \{(x, y) \mid (\neg x \vee y) \in M\}$. To see this, by unsatisfiability, it must hold that both s and $\neg t$ are in M (otherwise assigning all to true or false satisfies M). If there is a path from s to t, via the edge set E', then M is unsatisfiable: the clauses corresponding to E' give a chain of implications that imply t to

be true (and s to be false). If there is no path from s to t in edge set E', then one can partition the edges of E' into those that are reachable from s (are in a path from s), are backwards reachable from t, and are neither. Assigning all variables to true in the first edge set (all vertices in the edges) and all variables to false in the second edge set satisfies the corresponding clauses: if a variable is "reachable" from s then this variable is not backwards reachable from t, and vice versa. Thus, this gives a partial truth assignment satisfying all clauses of the first two sets. For the third set, first consider all variables already implied by the partial variable assignment. This satisfies all clauses where a variable occurs in the partial assignment: if the variable is assigned false, then it must be negated in the clause (otherwise it would be in the second set), similarly for assignments to true. The remaining variables can be assigned to true, satisfying the remaining clauses. Thus, unsatisfiable subsets of $\mathcal{K}$ directly correspond to sub graphs of G from which one can reach t from s. If the unsatisfiable subset is minimal, the sub graph corresponds to a simple path, implying the correctness of the reduction. $\qquad\square$