Jérôme Lang
Pierre Marquis

Removing Inconsistencies in Assumption-based Theories Through Knowledge-Gathering Actions*

Abstract. In this paper, the problem of purifying an assumption-based theory KB, i.e., identifying the right extension of KB using knowledge-gathering actions (tests), is addressed. Assumptions are just normal defaults without prerequisite. Each assumption represents all the information conveyed by an agent, and every agent is associated with a (possibly empty) set of tests. Through the execution of tests, the epistemic status of assumptions can change from "plausible" to "certainly true", "certainly false" or "irrelevant", and the KB must be revised so as to incorporate such a change. Because removing all the extensions of an assumption-based theory except one enables both identifying a larger set of plausible pieces of information and renders inference computationally easier, we are specifically interested in finding out sets of tests allowing to purify a KB (whatever their outcomes). We address this problem especially from the point of view of computational complexity.

Keywords: inconsistency handling, knowledge-gathering actions, tests, assumption-based reasoning.

1. Introduction

In the following, knowledge bases given by propositional assumption-based theories $KB = \langle W, \Delta \rangle$ (or equivalently, prerequisite-free normal default theories), are considered. W is a finite set of propositional formulas representing certain information about *the real world*, the latter being characterized by a specific (a priori unknown) model I. Δ is a finite set of defaults (or assumptions); defaults are just propositional formulas that are plausible, i.e., considered as true in absence of any contradictory information. Intuitively, defaults are to be used as much as possible in any inference, unless a contradiction occurs. Since defaults may be jointly contradictory, several belief states (called extensions) may result. Equivalently, we can also consider KB as a (typically inconsistent) stratified belief base, with two strata (the first one W being consistent).

The purpose of reasoning from KB is to identify formulas that are true in I, ideally all of them. Of course, in the general case, only some of the formulas true in I are accessible. Anyway, an important question is: what

* This paper is an extended version of the paper "In search of the right extension", Proc. of the 7^{th} International Conference on Principles of Knowledge Representation and Reasoning (KR'2000), 625-636, 2000.

Studia Logica **67**: 179–214, 2001.

are the logical links between what is derivable from KB and what is true in I? When dealing with assumption-based theories, the following representation (meta)assumption is typically made:

(D) There exists a extension of KB (noted E_I) that is true in I.

Note that E_I is necessarily unique (because any two extensions of KB are jointly contradictory). It is important to note that (D) is a representation assumption (just like the one consisting in assuming that W is true in I). Eventually, it can be defeated at the light of certain information. For instance, it can be the case that a default belonging to every extension of KB is actually false in I. Thus, there is no guarantee that every formula derived from KB through skeptical inference (i.e., every formula belonging to all the extensions of KB) is true in I (defaults represent plausible pieces of information, not certain ones). Nevertheless, such a (meta)assumption is mandatory to take defaults into account in reasoning. Without it, only the logical consequences of W would be considered as true in I and pure deduction from W is definitely an over-cautious form of inference in presence of defaults (since it does not exploit them at all). Under assumption (D), skeptical inference enables characterizing all the pieces of information that must be true in I. Among the skeptically derivable formulas are all the logical consequences of W, plus some additional ones in the general case.

Now, what can be done if we want to identify some other formulas as true in I? A classical way ([2]) consists in taking into consideration representation (meta)assumptions that are stronger than (D). Especially, if some priorities on defaults are available (e.g., a pre-order over Δ reflecting the fact that some defaults are more plausible than others is given), it can be valuable to take it into account and to consider that at least one *preferred* extension of KB is true in I. Indeed, restricting the set of extensions of KB by focusing on the preferred ones may only lead to enlarge their intersection (hence the set of formulas that are considered as true in I). An extreme, most informative, case is achieved when all the extensions of KB are filtered out, except one. In this situation, every formula from the remaining extension is considered true in I. Additionally, provided that KB has only one extension, assumption-based reasoning becomes computationally easier (from the second level of the polynomial hierarchy to the first level). This is another reason that makes it highly desirable to have KB with one extension.

In this paper, a radically different but complementary approach is developed. We assume that every default from Δ represents all the pieces of information conveyed by one agent (a source of information); in the case where more than one elementary piece of information is offered by an agent, the conjunction of them is considered as a single default. W represents the set of all pieces of information on which all agents agree (e.g., what must be true by nature). We also assume that each agent can perform some knowledge-gathering actions (informative tests). Like in everyday life, agents have the opportunity to perform tests ("act") to reason better. At the light of tests outcomes, the group of agents can change its mind about the epistemic status of defaults. As a consequence, revising KB with the outcome of a test can easily modify the set of its extensions. Specifically, contrariwise to what happens in the preferred extension approach, revising KB can lead to remove defaults in some extensions or even to generate new extensions, thus questioning the (meta)assumption `(D)`.

In the following, we first assume that each agent is associated with a single test whose outcome directly validates the corresponding default (changing its status from "plausible" to "certainly true"), invalidates the default (changing its status from "plausible" to "certainly false") or cancels it (changing its status to "irrelevant"). Then we show how such default verification tests can be generated from more general tests through a pre-processing phase, and we analyze the complexity of the pre-processing. In both situations, the objective is to find sequences of tests that enable to *purify* the KB, i.e., to generate an assumption-based theory with a unique extension once revised, whatever the outcomes of the tests. What makes things difficult is that (i) the outcome of a test cannot be predicted, and (ii) not every default is testable. Moreover, tests are often associated with costs (the cost of performing a test); in this situation, it is rational to look for sequences of tests of minimal aggregated costs. We analyze the purification problem, especially from the computational complexity point of view.

2. Formal preliminaries

We assume the reader acquainted with basic notions of propositional logic. Let PS be a denumerable set of propositional variables. $PROP_{PS}$ is the propositional language built up from PS, the connectives and the Boolean constants $true$ and $false$ in the usual way. For every $V \subseteq PS$, $PROP_V$ denotes the sublanguage of $PROP_{PS}$ generated from the variables of V only. A *literal* of $PROP_V$ is either a variable of V (positive literal) or the negation of a variable of V (negative literal). A clause δ (resp. a term γ) of $PROP_V$ is a (possibly empty) finite disjunction (resp. conjunction) of literals of $PROP_V$. A CNF (resp. a DNF) formula of $PROP_V$ is a finite conjunction of clauses (resp. a finite disjunction of terms) of $PROP_V$. As usual, every finite set of formulas from $PROP_V$ is identified with the conjunction of its elements. Formulas are interpreted in the classical way.

In the following, we make use of the following notations: if A is a set of subsets of a given set, then $A_{\min\subset} = \{B \in A \mid \text{there is no } B' \in A \text{ s.t. } B' \subset B\}$ and $A_{\min\supset} = \{B \in A \mid \text{there is no } B' \in A \text{ s.t. } B' \supset B\}$.

The complexity results we give in this paper also refer to some complexity classes which deserve some recalls. More about them can be found in Papadimitriou's textbook [16]. We assume that the classes P, NP and coNP are known to the reader. The following classes will also be considered:

- $\Delta_2^p[\mathcal{O}(n)]$ is the class of problems that can be decided using only a linear number of calls to an NP oracle.
- $\Pi_2^p = \mathsf{co}\Sigma_2^p = \mathsf{coNP}^{\mathsf{NP}}$. The canonical Π_2^p-complete problem 2-$\overline{\text{QBF}}$ is the set of all triples $\langle A = \{a_1, ..., a_m\}, B = \{b_1, ..., b_n\}, \Phi\rangle$ where A and B are two disjoint sets of propositional variables and Φ is a formula of $PROP_{A\cup B}$. A positive instance of this problem is a triple $\langle A, B, \Phi\rangle$ such that for every A-world ω_A there exists a B-world ω_B for which $(\omega_A, \omega_B) \models \Phi$.

NP, coNP and $\Delta_2^p[\mathcal{O}(n)]$ are located at the first level of the so-called polynomial hierarchy [23] which plays a prominent role in knowledge representation and reasoning. Contrastingly, Σ_2^p and Π_2^p are at the second level of the polynomial hierarchy.

3. Basics of assumption-based reasoning

Throughout this paper, knowledge bases are represented by propositional assumption-based theories [17]. The following notions related to assumption-based reasoning are to be used:

Definition 1. (assumption-based reasoning)

- An *assumption-based theory* KB is an ordered pair $KB = \langle W, \Delta\rangle$, where W is a (conjunctively-interpreted) finite subset of $PROP_{PS}$, and $\Delta = \{\phi_1, \ldots, \phi_n\}$ is a finite subset of $PROP_{PS}$.
- A *nogood* of KB is a minimal subset N of Δ (w.r.t. $\subseteq$) such that $N \cup W$ is not consistent. $Nogood(KB)$ is the set of all nogoods of KB.
- A *maximal scenario* of KB is a maximal subset S of Δ (w.r.t. $\subseteq$) such that $S \cup W$ is consistent. $MaxScen(KB)$ is the set of all maximal scenarios of KB.
- An *extension* E_S of KB is the deductive closure $Cn(S \cup W)$, where S is a maximal scenario of KB. S is called the *generating scenario* of E_S. $Ext(KB)$ is the set of all extensions of KB.

EXAMPLE 1. Let $KB = \langle W, \Delta \rangle$ with $W = \emptyset$ and $\Delta = \{a \wedge b, \neg a, \neg b \wedge c, d\}$. We have $Nogood(KB) = \{\{a \wedge b, \neg a\}, \{a \wedge b, \neg b \wedge c\}\}$, $MaxScen(KB) = \{\{a \wedge b, d\}, \{\neg a, \neg b \wedge c, d\}\}$, and $Ext(KB) = \{Cn(\{a \wedge b, d\}), Cn(\{\neg a, \neg b \wedge c, d\})\}$.

Without loss of generality, we assume that Δ is totally ordered (w.r.t. any (fixed) ordering)), so that we can associate to every default ϕ_i of Δ its index i ($i \in 1..n$) that fully identifies it. Since we assume that all the information conveyed by a given source is encoded as a single default, every index i represents a source. At a first glance, considering Δ as a set can appear too restrictive since it can be the case that two different sources convey the same piece of information: hence, a bag (or multiset) would be more convenient. Nevertheless, for simplicity and without loss of generality, Δ will be considered as a set: if two identical defaults are to be considered, it is sufficient to make them syntactically different while preserving equivalence.

Now, let us recall that every assumption-based theory KB has at least a maximal scenario whenever W is consistent, and more than one whenever it has a nogood containing more than one assumption. It is easy to see that $MaxScen(KB)$ and $Ext(KB)$ are two similar sets (they have the same number of elements) since every extension of KB is generated by a unique maximal scenario of KB (the maximality requirement ensures that the union of two distinct maximal scenarios of KB is inconsistent with W). This is particularly interesting since computing extensions (as deductively closed sets) would be infeasible from a practical point of view. Accordingly, determining whether KB has only one extension can be decided in linear time when either $Nogood(KB)$ or $MaxScen(KB)$ are given (but both sets may contain exponentially many elements w.r.t. the size of KB in the worst case). Due to the maximality requirement for scenarios, the truth value of every default ϕ_i of Δ is determined in every extension E_S of KB: $\forall \phi_i \in \Delta \forall E_S \in Ext(KB), \phi_i \in E_S$ or (exclusive) $\neg\phi_i \in E_S$. As a consequence, every extension of KB is also fully characterized by the set of defaults of Δ that do not belong to it. Finally, it is well-known that $Nogood(KB)$ can be generated from $MaxScen(KB)$ and *vice-versa*, through the computation of minimal hitting sets [19] and set-theoretic complements in Δ.

Further on, we make use of the following (easy) result: KB has only one extension iff $W \cup \{\phi_i \mid \phi_i \in \Delta, W \cup \{\phi_i\}$ consistent$\}$ is consistent. Interestingly, this result shows that considering KB with only one extension is computationally valuable; indeed, computing the corresponding maximal scenario can be achieved through only n calls to an NP-oracle (one consistency check per ϕ_i in Δ); once this is done, skeptical inference (that coincides in this case with credulous inference and argumentative inference) amounts to classical inference, and is "only" coNP-complete (while it is Π_2^p-complete in the general case [7] [22]).

4. Revising defaults using tests

4.1. A simple default verification framework

As evoked above, we assume in the following that each default ϕ_i of Δ represents all the information conveyed by the agent (or source) i. Some of the defaults ϕ_i are associated with a test t_i (at most one per default). Formally:

DEFINITION 2. (verification problems) A *verification problem* is an ordered pair $P = \langle KB, T \rangle$, where $KB = \langle W, \Delta \rangle$ is an assumption-based theory and $T = \{t_1, \ldots, t_k\}$ is the available set of tests. For every $\phi_i \in \Delta$, ϕ_i is *testable* iff $t_i \in T$. Note that without loss of generality, we assume that only the first k defaults $\phi_1 \ldots \phi_k$ of Δ are testable ($0 \leq k \leq \mathrm{card}(\Delta)$).

Executing a test t_i can change the epistemic status of ϕ_i (considered plausible before execution) in one of the following ways:

- ϕ_i becomes certainly true, noted $ok(i)$.
- ϕ_i becomes certainly false, noted $bad(i)$.
- ϕ_i becomes irrelevant, noted $forget(i)$.

The outcome r_i of t_i will always be one among $\{ok(i), bad(i), forget(i)\}$. Since the very purpose of testing a default ϕ_i is to validate it, $ok(i)$ is required to be among the possible outcomes of it. Formally:

DEFINITION 3. (tests) A *default verification test* t_i consists of a subset of $\{ok(i), bad(i), forget(i)\}$ containing $ok(i)$ and at least one additional element. The elements r_i of t_i are its *possible outcomes*. To each test t_i is attached a positive integer k_i representing the *cost* of performing t_i.

While such tests can be considered quite restricted at a first glance, they nevertheless capture several important notions of verification:

- *default verification*: the piece of information itself (ϕ_i) given by the source i is verified. Assume for example that undergrad student Tweety programmed an algorithm for testing primality, which resulted after some laborious hours in software S. Running S on number 348754697 returns "yes" (ϕ_i is thus "348754697 is prime"). Tweety's ability in programming is known to be average, therefore, ϕ_i should be considered a default and not a fact. Now, Lea is an expert in prime numbers. She is fully reliable, thus, her answer to the question whether a given number is prime is always correct. Furthermore, Lea always gives an answer whenever the input number does not exceed a given maximum number $N_{\max}$ (much larger than 348754697), and sometimes gives an answer when the input number exceeds $N_{\max}$. Asking Lea about the primality of 348754697 is a *default verification test* of the form $\{ok(i), bad(i)\}$: its outcome is $ok(i)$ (resp. $bad(i)$) if Lea answers "yes" (resp. "no"). Some years later, Tweety improves his software so as to deal with very large integers. Running the improved software on $N = 495729^{50876} + 34532^{3486}$ (which exceeds $N_{\max}$) returns "yes", which leads to considering the default ϕ_j "N is prime". Asking Lea about the primality of N is a default verification test of the form $\{ok(j), bad(j), forget(j)\}$, because $N > N_{\max}$. If Lea is not able to answer, i.e., if the outcome of the test is $forget(j)$, then we will not be ready to consider N as prime any longer, because Lea is an expert and the fact that she was not able to answer is a reason strong enough for changing the status of ϕ_j from plausible to irrelevant.

- *source verification*: the reliability of the source, thus the *relevance* of the piece of information, is checked. Not the piece of information itself. Bob is Tweety's teacher. Asking Bob to look carefully on Tweety's first program S returns yes or no whether S is valid or not. If it is, then the information S gave was correct and 348754697 is prime. If it is not, then anything could happen. The program might have been something like generating "yes" or "no" randomly. This cancels the default information that ϕ_i is true but certainly does not tell us that ϕ_i is false. It could well be true in I, *by chance* (here, 348754697 may well be prime even if S is not valid). Asking Bob is a test of the form $\{ok(i), forget(i)\}$.

Now, once t_i has been executed, KB must be revised so as to incorporate the information given by its resulting outcome r_i into KB. The revision process aims at characterizing the new assumption-based theory $KB * r_i$ obtained by incorporating the information given by r_i within KB:

DEFINITION 4. (revising KB with a test outcome) Let $KB = \langle W, \Delta \rangle$ an assumption-based theory, and t_i be a test concerning default ϕ_i of Δ. We have:

- **default validation**

  $$KB * ok(i) = \langle W \cup \{\phi_i\}, \Delta \setminus \{\phi_i\} \rangle$$

  The execution of t_i has validated ϕ_i. The default is definitely true in I. It can be soundly added to W and removed from Δ.

- **default invalidation**

  $$KB * bad(i) = \langle W \cup \{\neg\phi_i\}, \Delta \setminus \{\phi_i\} \rangle$$

  The execution of t_i has invalidated ϕ_i. The default is definitely false in I. Its negation can be soundly added to W and ϕ_i can be removed from Δ.

- **default cancellation**

$$KB * \mathit{forget}(i) = \langle W, \Delta \setminus \{\phi_i\}\rangle$$

The default is considered irrelevant, it is removed from Δ. A frequent reason for cancelling a default is a source verification that fails (as explained above); the failure can be viewed as an argument against the default, but of course, this argument is not sufficiently strong for invalidating the default (we just consider it as irrelevant).

Note that letting ϕ_i in Δ or not once ϕ_i (resp. $\neg\phi_i$) has been added to W has no influence on $\mathit{Ext}(KB * ok(i))$ (resp. $\mathit{Ext}(KB * bad(i))$).

As the definition above shows it, tests are assumed reliable (i.e., their outcome is true in I); they are also assumed non-intrusive (they have no effect on the world). Accordingly, revising KB by a sequence of test outcomes depends only on the outcome of each individual test, and is independent from the ordering in which tests are performed.

Once this is stated, we can define the notion of revision of an assumption-based theory by a set of tests.

DEFINITION 5. (revising KB by a set of tests) Let $P = \langle KB, T\rangle$ be a verification problem.

- A *set of tests* $X = \{t_1, \ldots t_l\}$ (given P) is any subset of T.
- Executing a set of tests $X = \{t_1, \ldots t_l\}$ results in a *state* (for P), i.e., a set $R = \{r_1, \ldots, r_l\}$ where each r_i ($i \in 1, \ldots, l$) is an element of t_i (one of its possible outcomes). R can be considered as the outcome of X. $\mathit{State}(X)$ is the set of states that can possibly result from the execution of X.
- A state R (for P) is *consistent* iff $W \cup \{\phi_i \mid ok(i) \in R\} \cup \{\neg\phi_i \mid bad(i) \in R\}$ is consistent. Clearly, since tests are assumed reliable, all states in $\mathit{State}(X)$ are consistent.
- The *revision* of KB by a consistent state $R = \{r_1, \ldots, r_l\}$, noted $KB * R$, is the assumption-based theory defined by $KB * r_1 * \cdots * r_l$.

It is convenient to partition every state R in $\{\mathit{Ok}(R), \mathit{Bad}(R), \mathit{Forget}(R)\}$ with:

- $\mathit{Ok}(R) = \{i \mid ok(i) \in R\}$,
- $\mathit{Bad}(R) = \{i \mid bad(i) \in R\}$,
- $\mathit{Forget}(R) = \{i \mid \mathit{forget}(i) \in R\}$.

We also define $\mathit{Facts}_P(R) = \{\phi_i \mid i \in \mathit{Ok}(R)\} \cup \{\neg\phi_i \mid i \in \mathit{Bad}(R)\}$.

4.2. What may happen when a default is revised?

For our purpose, it is important to identify the effect of each possible default revision (validation, invalidation, cancellation) on $Ext(KB)$. Since it is possible to determine efficiently whether KB has a unique extension given $Nogood(KB)$ or $MaxScen(KB)$, it is also important to analyze how both sets evolve after a revision.

The case of default validation corresponds to a "soft" knowledge expansion. By "soft" we mean that there is no *surprise*: what was considered as a plausible piece of information becomes now a certain piece of information. The effects are a (possible) removing of some of the current extensions, but never the generation of a new extension (point 3. below).

PROPOSITION 1. (effects of a default validation)

1. $Nogood(KB * ok(i)) = \{N \setminus \{\phi_i\} \mid N \in Nogood(KB)\}_{\min\subset}$.
2. $MaxScen(KB * ok(i)) = \{S \setminus \{\phi_i\} \mid S \in MaxScen(KB)$ *and* $\phi_i \in S\}$.
3. $Ext(KB * ok(i)) \subseteq Ext(KB)$.

Thus, the extensions of $KB * ok(i)$ are exactly those extensions of $Ext(KB)$ which contain ϕ_i. The other ones are removed. Consider Example 1 again with $\phi_1 = a \wedge b$. We have $Nogood(KB * ok(1)) = \{\{\neg a\}, \{\neg b \wedge c\}\}$ and $MaxScen(KB * ok(1)) = \{\{d\}\}$. Accordingly, $KB * ok(1)$ has a unique extension $Cn(\{a \wedge b, d\})$. Clearly enough, point 3. can be easily generalized (by induction) to *sequences of validation*: namely, if a state R contains only $ok(i)$ variables, then $Ext(KB * R) \subseteq Ext(KB)$.

PROPOSITION 2. (effects of a default cancellation)

1. $Nogood(KB * forget(i)) = \{N \in Nogood(KB) \mid \phi_i \notin N\}$.
2. $MaxScen(KB * forget(i)) = \{S \setminus \{\phi_i\} \mid S \in MaxScen(KB)\}_{\min\supset}$.
3. $\forall E \in Ext(KB * forget(i))$, *there is a* $E' \in Ext(KB)$ *such that* $E \subseteq E'$.

Default cancellation can be considered as a simple form of belief contraction. However, a default ϕ_i can survive its contraction in some sense; indeed, it can be the case that ϕ_i is a (skeptical) consequence of $KB * forget(i)$ (just consider the case where $\Delta = \{\phi_i, \phi_j\}$, $W \cup \{\phi_j\}$ is consistent and $\phi_j \models \phi_i$ holds). This is not an undesirable effect: if ϕ_i is a logical consequence of W when some defaults of Δ are validated (or only considered as true), the fact that the corresponding verification test t_i has failed is not a sufficient argument for considering ϕ_i as irrelevant. The discrepancy with the belief contraction situation can be explained by the fact that, while both cancellation and contraction change the status of some information to "irrelevant", the status of a default before change is "plausible", not "certainly true".

With default cancellation, preservation of extensions is not guaranteed; indeed, it may be the case that one of the extensions of KB leads to a weaker one. Consider Example 1 again, where $\phi_2 = \neg a$. We have $\mathit{Nogood}(KB * \mathit{forget}(2)) = \{\{a \wedge b, \neg b \wedge c\}\}$ and $\mathit{MaxScen}(KB * \mathit{forget}(2)) = \{\{a \wedge b, d\}, \{\neg b \wedge c, d\}\}$. Thus $KB * \mathit{forget}(2)$ has two extensions, $E_1 = Cn(\{a \wedge b, d\})$ and $E_2 = Cn(\{\neg b \wedge c, d\})$. While E_1 is an extension of KB, E_2 is only a proper subset of the extension $Cn(\{\neg a, \neg b \wedge c, d\})$ of KB.

When it comes to sequential revision, generalizing points 3. of both Propositions 1 and 2 by induction on $\mathrm{card}(R)$ easily gives the following result: if a state R is "bad-free", i.e., $\mathit{Bad}(R) = \emptyset$, then $\forall E \in \mathit{Ext}(KB * R)$, there is a $E' \in \mathit{Ext}(KB)$ s.t. $E \subseteq E'$. As a corollary, we have that $\mathrm{card}(\mathit{Ext}(KB * R)) \leq \mathrm{card}(\mathit{Ext}(KB))$: revising a KB with a "bad-free" state R does not increase the *number* of extensions of KB.

Default invalidation can be considered as a knowledge expansion with the negation of the default. Nevertheless, it is trickier than default validation: there may be new extensions appearing, and the number of extensions may increase. These more chaotic effects are in agreement with the intuition which tells that a default invalidation change the status of "plausible" to "certainly false", which leads to a *surprise* (contrariwise to default validation). Consider Example 1 again. We have $\mathit{Nogood}(KB * \mathit{bad}(2)) = \{\{a \wedge b, \neg b \wedge c\}\}$ and $\mathit{MaxScen}(KB * \mathit{bad}(2)) = \{\{a \wedge b, d\}, \{\neg b \wedge c, d\}\}$, thus $KB * \mathit{bad}(2)$ has two extensions: E_1 as above, and $E_3 = Cn(\{a, \neg b \wedge c, d\})$ *which is a new extension* — incompatible with any extension of KB.

Furthermore, while Propositions 1 and 2 show how both the nogoods and the maximal scenarios of a revised KB can be computed (in polynomial time) from, respectively, the nogoods and the maximal scenarios of KB whenever the revision operation is a default validation or a default cancellation, no similar results can be achieved when considering a default invalidation. Indeed, there is not enough information in $\mathit{Nogood}(KB)$ (or $\mathit{MaxScen}(KB)$) to compute $\mathit{Nogood}(KB * \mathit{bad}(i))$ (or $\mathit{MaxScen}(KB * \mathit{bad}(i))$) in the general case:

EXAMPLE 2. Let $KB = \langle W, \Delta \rangle$, with $W = \{a \vee \neg b \vee \neg c\}$ and $\Delta = \{a, b, c\}$. We have $\mathit{Ext}(KB) = \{Cn(W \cup \{a, b, c\})\}$, $\mathit{MaxScen}(KB) = \{\{a, b, c\}\}$, and $\mathit{Nogood}(KB) = \emptyset$. Now, given that $\phi_1 = a$, $\mathit{Ext}(KB * \mathit{bad}(1)) = \{Cn(W \cup \{\neg a\} \cup \{b\}), Cn(W \cup \{\neg a\} \cup \{c\})\}$, $\mathit{MaxScen}(KB * \mathit{bad}(1)) = \{\{b\}, \{c\}\}$, and $\mathit{Nogood}(KB * \mathit{bad}(1)) = \{\{b, c\}\}$. Let $KB' = \langle \emptyset, \Delta \rangle$. We have $\mathit{MaxScen}(KB') = \mathit{MaxScen}(KB)$ and $\mathit{Nogood}(KB') = \mathit{Nogood}(KB)$. Nevertheless, we do *not* have $\mathit{Ext}(KB' * \mathit{bad}(1)) = \mathit{Ext}(KB * \mathit{bad}(1))$ since $\mathit{Ext}(KB' * \mathit{bad}(1)) = \{Cn(\{\neg a, b, c\})\}$.

Accordingly, while knowing only nogoods or maximal scenarios is sufficient to determine whether KB, or the revision of KB by a sequence of validations and cancellations, has only one extension, this is no longer the case when default invalidations are possible. This calls for a more informative notion, like the one of *generalized nogood*:

DEFINITION 6. A *generalized nogood* of KB is a minimal (w.r.t. $\subseteq$) subset G of $\Delta \cup \{\neg\phi_i \mid \phi_i \in \Delta\}$ such that $G \cup W$ is not consistent. $GNogood(KB)$ is the set of all generalized nogoods of KB.

Clearly enough, we have by definition:

$$GNogood(\langle W, \Delta\rangle) = Nogood(\langle W, \Delta \cup \{\neg\phi_i \mid \phi_i \in \Delta\}\rangle)$$

and, as a consequence of point 1. of Proposition 2, we obtain:

$$Nogood(\langle W, \Delta\rangle) = \{G \in GNogood(\langle W, \Delta\rangle) \mid G \subseteq \Delta\}.$$

Accordingly, $Nogood(KB)$ can be computed in polynomial time from $GNogood(KB)$ by removing from it every element that is not a subset of Δ.

The following proposition shows how generalized nogoods evolve when defaults are revised:

PROPOSITION 3. (computing $GNogood(KB * r_i)$)

1. $GNogood(KB * ok(i)) =$
   $\{G \setminus \{\phi_i\} \mid G \in GNogood(KB) \text{ and } \neg\phi_i \notin G\}_{min\subset}$.
2. $GNogood(KB * forget(i)) = \{G \in GNogood(KB) \mid \phi_i \notin G \text{ and } \neg\phi_i \notin G\}$.
3. $GNogood(KB * bad(i)) =$
   $\{G \setminus \{\neg\phi_i\} \mid G \in GNogood(KB) \text{ and } \phi_i \notin G\}_{\min\subset}$.

Accordingly, computing $GNogood(KB * R)$ from $GNogood(KB)$ and a consistent state R can be achieved in polynomial time.

5. Purifying KB

In this section, we focus on the problem consisting of the identification of *the* right extension E_I of KB. To this end, we look for sets of tests which discriminate among all extensions, so as to find E_I. This calls for the notion of *purification* of KB by a set of tests X.

5.1. N-purification vs. C-purification

As seen before, revising KB with a state may eventually generate extensions that are not in $Ext(KB)$, hence questioning the (meta)assumption `(D)`. As a consequence, two different notions of purification must be considered, related to the fact that we are ready to defeat `(D)` or not. If `(D)` can be relaxed at the light of a new evidence, the objective of the revision process is that the revised KB has only one extension. If `(D)` cannot be questioned, a valuable revision process can lead to a revised KB with more than one extension, but only one of them must be among the initial extensions of KB.

DEFINITION 7. (purifying states) Given a verification problem $P = \langle KB, T\rangle$ and a consistent state R:

- *R N-purifies KB* iff $KB * R$ has only one extension.
- *R C-purifies KB* iff $KB * R$ has only one extension E s.t. $E \in Ext(KB)$.

Certainly, the uniqueness requirement in both definitions is quite idealistic and in practice one would rather try to do as good as possible (identifying as many inconsistencies as possible, even if it cannot be done totally). For the sake of simplicity, we ignore this issue in this paper.

It is important to see why both definitions of purification do not coincide (and none of them implies or excludes the other one):

EXAMPLE 3. Let $KB = \langle \emptyset, \{\neg a \wedge \neg b, a, b\}\rangle$. We have $Nogood(KB) = \{\{\neg a \wedge \neg b, a\}, \{\neg a \wedge \neg b, b\}\}$ and $MaxScen(KB) = \{\{\neg a \wedge \neg b\}, \{a, b\}\}$; thus, KB has two extensions, namely $E = Cn(\{\neg a \wedge \neg b\})$ and $E' = Cn(\{a, b\})$. Let $\phi_1 = \neg a \wedge \neg b$, $\phi_2 = a$, and $\phi_3 = b$. Let $R = \{bad(2)\}$. $KB * R = \langle\{\neg a\}, \{\neg a \wedge \neg b, b\}\rangle$ has *two* extensions, namely $Cn(\{\neg a \wedge \neg b\})$ and $Cn(\{\neg a, b\}\}$, and therefore R does not N-purify KB. Contrastingly, since $Ext(KB) \cap Ext(KB * R) = \{Cn(\{\neg a \wedge \neg b\})\}$ is a singleton, R C-purifies KB. Now, let $R' = \{forget(1), bad(2)\}$; $KB * R' = \langle\{\neg a\}, \{b\}\rangle$ has only one extension $Cn(\{\neg a, b\})$ and thus R' N-purifies KB, while R' does not C-purify KB. Lastly, $R'' = \{ok(3)\}$ both N-purifies and C-purifies KB.

In N- and C-purification, N and C stand respectively for *no completion* and *completion*, which is justified by the following result where "forget-free" states are considered, only:

PROPOSITION 4. (C- and N-purification) *Let $P = \langle KB, T\rangle$ be a verification problem, with $KB = \langle W, \Delta\rangle$, and let R be a consistent "forget-free" state (i.e., $Forget(R) = \emptyset$). The following statements are equivalent:*

(a) *R C-purifies KB.*

(b) *R N-purifies $\langle Compl_\Delta(W), \Delta\rangle$, where $Compl_\Delta(W) = W \cup \{\bigvee \{S \mid S \in MaxScen(KB)\}\}$ is the* completion *of W w.r.t. Δ.*

(c) *There is a single extension of KB containing $Facts_P(R)$.*

(d) *There is a single extension of KB consistent with $Facts_P(R)$.*

(e) *For each* $\phi_i \in \Delta$, *either* $Compl_\Delta(W) \wedge Facts_P(R) \models \phi_i$ *or* $Compl_\Delta(W) \wedge Facts_P(R) \models \neg\phi_i$ *(the truth value of each default is fully determined from* R *and the completion of* W *w.r.t.* Δ*).*

This result does not extend to general states, as shown by the following example.

EXAMPLE 4. Let $KB = \langle \emptyset, \{a, b \wedge c, b \wedge \neg c\}\rangle$. We have $Nogood(KB) = \{\{b \wedge c, b \wedge \neg c\}\}$; $MaxScen(KB) = \{\{a, b \wedge c\}, \{a, b \wedge \neg c\}\}$; $Ext(KB) = \{Cn(a \wedge b \wedge c), Cn(a \wedge b \wedge \neg c)\}$. Let $R = \{forget(2) \wedge forget(3)\}$. $Ext(KB * R) = \{Cn(a)\}$, from which we infer that (a) R N-purifies KB and (b) R does not C-purify KB. Now, $Compl_\Delta(W) \equiv (a \wedge b \wedge c) \vee (a \wedge b \wedge \neg c) \equiv a \wedge b$. Hence, $Compl(KB) * R = \langle (a \wedge b \wedge c) \vee (a \wedge b \wedge \neg c), \{a\}\rangle$. Consequently, $Ext(Compl(KB) * R) = \{Cn(a \wedge b)\}$, and R N-purifies $Compl(KB)$. We also get that $Compl_\Delta(W) \wedge Facts_P(R) \not\models b \wedge c$ and $Compl_\Delta(W) \wedge Facts_P(R) \not\models b \wedge \neg c$, thus ϕ_2 (and ϕ_3 as well) is not determined in the completion of KB.

Clearly enough, replacing W by $Compl_\Delta(W)$ in KB amounts to considering that `(D)` cannot be questioned; indeed, the deductive closure of $Compl_\Delta(W)$ is equal to the deductive closure of W interpreted under `(D)`.

Which one of both definitions should be used in practice, i.e., whether we are ready to defeat `(D)` or not, is a nontrivial problem (which is to be considered as well in model-based diagnosis). Stated otherwise, this question becomes: when taking account of an observation for discriminating among extensions (or diagnoses), should we start from the initial set of extensions and filter out those that are incompatible with the observation (C-purification) or expand the certain knowledge with the observation and then recomputing the extensions (N-purification)?

5.2. Computing purifying sets

In the following, we focus on N-purification, only. We are interested in determining sets of tests that enable purifying KB whatever their outcomes. We are specifically interested in the minimal ones w.r.t. $\subseteq$, called *purification bases.*

DEFINITION 8. (purifying sets of tests, bases) Given a verification problem $P = \langle KB, T\rangle$, and a set $X \subseteq T$ of tests:

- X *N-purifies* KB iff for any consistent state $R \in State(X)$, R N-purifies KB. X is said a *purifying set* for KB.
- X is a *N-purification base* (or base for short) for KB iff X N-purifies KB and no proper subset of it does it. $Base(KB)$ is the set of all N-purification bases for KB.

- KB is *N-purifiable* iff there exists a N-purification base (or, equivalently, if there exists a purifying set) for it.

In order to figure out the complexity of computing purifying sets, we investigate the complexity of the two following decision problems:

- PURIFYING STATE: does a given state R N-purify KB?
- PURIFYING SET: does a given set X of tests N-purify KB?

PROPOSITION 5. PURIFYING STATE *is in* $\Delta_2^p[\mathcal{O}(n)]$, *and is both* NP*-hard and* coNP*-hard.*

Accordingly, checking whether a given assumption-based theory has a unique extension is not in $\mathsf{NP} \cup \mathsf{coNP}$, unless $\mathsf{NP} = \mathsf{coNP}$ (which is considered very unlikely).

PROPOSITION 6. PURIFYING SET *is* Π_2^p*-complete.*

These complexity results show the computation of purifying sets highly intractable. Indeed, checking whether a given set X is a purifying set for KB is already Π_2^p-hard. This is why it is important to identify restrictions for which the computational task can prove easier.

Accordingly, in the following, we assume that the available set T of tests contains only source verification tests, i.e., tests t_i of the form $\{ok(i), forget(i)\}$. Stated otherwise, we assume that performing any set X of tests can only lead to "bad-free" states. This assumption is called `(bad-free)` in the following. A common situation where the `(bad-free)` assumption holds is when the only tests considered are *source verification tests.*

Interestingly, purifying sets for KB can be characterized from $Nogood(KB)$ under this assumption:

PROPOSITION 7. (purification under `(bad-free)`) *Let* $P = \langle KB, T \rangle$ *be a verification problem satisfying the* `(bad-free)` *assumption. For any nogood* N *of* KB*, let* $Index(N) = \{i \mid \phi_i \in N\}$.

1. *A consistent state R N-purifies KB if and only if for every nogood N of KB, at least one of the following three statements is true:*
   (i) $Index(N) \cap Forget(R) \neq \emptyset$;
   (ii) $Index(N) \setminus Ok(R)$ *is a singleton;*
   (ii) $\exists N' \in Nogood(KB)$ *s.t.* $Index(N') \setminus Ok(R)$ *is a singleton* $\{i\}$ *with* $i \in Index(N)$.
2. *A set X of tests N-purifies KB if and only if for every nogood N of KB, we have* $\mathrm{card}(Index(N) \setminus \{i \mid t_i \in X\}) \leq 1$.

The first point means that R purifies KB if (i) every nogood N is broken by the cancellation of a default $\phi_i \in N$, or if (ii) all defaults of N except one have been validated, which enables to deduce that the remaining one has to be cancelled, or (iii) if it has been possible to deduce by reasoning as in (ii) about another nogood N' that a given default ϕ_i of N has to be cancelled[1]. The second point means that X purifies KB if and only if every nogood contains at most one default whose associated test t_i is not in X.

The above proposition clearly shows that the complexity of both PURIFYING STATE and PURIFYING SET decreases to P when the input is $Nogood(KB)$ and X. As an additional corollary, we have:

COROLLARY 1. *Let* $P = \langle KB, T\rangle$ *be a verification problem satisfying the* `(bad-free)` *assumption.*

- *If* X *and* X' *are two sets of tests such that* $X \subseteq X'$ *and* X *N-purifies* KB*, then* X' *N-purifies* KB *as well.*
- KB *is N-purifiable iff* T *N-purifies* KB.

Clearly enough, this corollary would not hold in the general case, if the `(bad-free)` assumption were not made (see Example 3 with $X = \emptyset$ and $X' = \{\{ok(1), bad(1)\}\}$ for a counterexample).

Therefore, while the `(bad-free)` assumption does not lower the complexity of PURIFYING SET in the general case, it enables to determine whether a KB is N-purifiable in polynomial time whenever the input is $Nogood(KB)$ and T. Moreover, in the case where KB is N-purifiable, a base for KB can be computed in polynomial time from $Nogood(KB)$ and T thanks to the following greedy algorithm:

`One-base`$(Nogood(KB), T)$:
1. Start with $B = T$;
2. For every element t_i of B, remove t_i from B if $B \setminus \{t_i\}$ still N-purifies KB.

Once all the tests of T have been processed, the resulting set B is, by construction, a base for KB. This algorithm can also be used to determine in polynomial time whether or not a given set X of tests is a base for KB (just start the algorithm with $B = X$: if the resulting set of tests is X, then X is a base for KB, otherwise it is not). Accordingly, the computation of $Base(KB)$ can be achieved through a simple generate-and-test procedure using the previous greedy algorithm as a routine. The search can be organized using the SE-tree framework given in [20].

---

[1] Note that (ii) can be omitted since (iii) is a logical consequence of it (just take N' = N); we keep it only for the sake of clarity.

EXAMPLE 5. Let $P = \langle\langle W, \Delta\rangle, T\rangle$ a verification problem satisfying the `(bad-free)` assumption. Let $\Delta = \{\phi_1, \ldots, \phi_7\}$, and let us assume that each default ϕ_i ($i \in 1..7$) is testable. Assume that

$$Nogood(KB) = \{\{\phi_1, \phi_2, \phi_5\}, \{\phi_2, \phi_6\}, \{\phi_3, \phi_6\}, \{\phi_4, \phi_5, \phi_6\}, \{\phi_7\}\}.$$

We have:

$$Base(KB) = \{\{t_1, t_2, t_4, t_6\}, \{t_1, t_3, t_4, t_6\}, \{t_1, t_5, t_6\}, \{t_2, t_3, t_4, t_5\}, \{t_2, t_5, t_6\}\}.$$

Interestingly, for any verification problem $P = \langle KB, T\rangle$ where T contains only source verification tests, it is always possible to reduce the problem of computing $Base(KB)$ to the problem of computing the bases corresponding to a verification problem for which *every source is testable.* This calls for the notion of *necessary* source.

DEFINITION 9. (necessary sources) Let $P = \langle KB, T\rangle$ be a verification problem where T consists of source verification tests only, and let i be a testable source (i.e., $t_i \in T$). i is *necessary* if and only if (a) t_i belongs to every purification base for KB and (b) $Base(KB)$ is not empty. We note $Nec(P)$ the set of all necessary sources.

We have the following characterizations of necessary sources.

PROPOSITION 8. *Let $P = \langle KB, T\rangle$ be a verification problem where T consists of source verification tests, only. Let $KB = \langle W, \{\phi_1, \ldots, \phi_n\}\rangle$. The following statements are equivalent:*

1. *i is necessary.*
2. *$Base(KB)$ is not empty, and there exists a nogood N of KB s.t. $\phi_i \in N$ and $|Index(N) \cap \{j \in \{1, \ldots, n\} \mid t_j \notin T\}| = 1$.*
3. *T is a purifying set for KB but $T \setminus \{t_i\}$ is not.*

Point 2. of Proposition 8 shows that (i) no source is necessary when every source is testable and (ii) $Nec(P)$ can be computed in $\mathcal{O}(|Nogood(KB)| * \mathrm{card}(T))$ time.

Now, an approach to compute $Base(KB)$ consists in simplifying $Nogood(KB)$ thanks to the following algorithm:

`Simplify`$(Nogood(KB), Nec(P), T)$:

1. If there exists a nogood of KB containing more than one default ϕ_i that is not testable, then return($\emptyset$) (KB is not purifiable).
2. Replace every nogood N of KB by $N \setminus \{\phi_i \mid i \in Nec(P)\}$.
3. Remove from the resulting set every set containing less than two defaults.

We have the following characterization:

PROPOSITION 9. *Let $P = \langle KB, T\rangle$ be a verification problem where T consists of source verification tests, only. Let $KB = \langle W, \Delta\rangle$. $X \cup \{\phi_i \in \Delta \mid i \in Nec(P)\}$ is a base for KB iff X is a base for KB' s.t. $Nogood(KB') =$ `Simplify`$(Nogood(KB), Nec(P), T)$.*

EXAMPLE 6. Let $P = \langle\langle W, \Delta\rangle, T\rangle$ a verification problem where T consists of source verification tests, only. Let $\Delta = \{\phi_1, \ldots, \phi_7\}$. Assume that

$$Nogood(KB) = \{\{\phi_1, \phi_2, \phi_5\}, \{\phi_2, \phi_6\}, \{\phi_3, \phi_6\}, \{\phi_4, \phi_5, \phi_6\}, \{\phi_7\}\}.$$

If all sources are testable ($T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7\}$), then no source is necessary.

If all sources except 4 are testable, then 5 and 6 are necessary. Accordingly, `Simplify`$(Nogood(KB), Nec(P), T) = \{\{\phi_1, \phi_2\}\}$. Let KB' be an assumption-based theory s.t. $Nogood(KB') = \{\{\phi_1, \phi_2\}\}$. We have $Base(KB') = \{\{\phi_1\}, \{\phi_2\}\}$; hence, we also have $Base(KB) = \{\{\phi_1, \phi_5, \phi_6\}, \{\phi_2, \phi_5, \phi_6\}\}$.

If all sources except 1 and 4 are testable, then 2, 5 and 6 are necessary. Therefore, we have `Simplify`$(Nogood(KB), Nec(P), T) = \emptyset$. Let KB' be an assumption-based theory s.t. $Nogood(KB') = \emptyset$. We have $Base(KB') = \{\emptyset\}$; hence, $Base(KB) = \{\{\phi_2, \phi_5, \phi_6\}\}$.

If all sources except 1 and 2 are testable, then KB is not purifiable and $Base(KB) = \emptyset$.

The tractability of PURIFYING SET suggests that the search for *preferred bases*, i.e., bases of minimal cost, could be envisioned under the `(bad-free)` assumption when the input is $Nogood(KB)$ and T. In order to simplify the exposure, we can also assume without loss of generality that every source is testable. If it is not the case, $Nogood(KB)$ can be simplified as exposed above so as to recover this situation; note that the simplification procedure described above runs in time polynomial in the input size (hence the assumption that every source is testable does not have a significant influence on the computational side).

Here, several aggregation policies can be considered in order to define the *cost* $k(X)$ of a set X of tests:

- $k(X) = \sum_{i \in X} k_i$ (SUMCOST policy);
- $k(X) = \max_{i \in X} k_i$ (MAXCOST policy).

The MAXCOST policy is particularly well-suited to the situation where the individual costs k_i represent running times and all tests are performed in parallel (in this case, the overall running time corresponds to the duration of the longest task). The SUMCOST policy must be considered in many of the remaining cases (especially, when the k_i represent monetary costs, or running times when the verification tasks are performed in a pure sequential way).

We have identified the following complexity results:

PROPOSITION 10. (MINIMAL COST BASE) *Let $P = \langle KB, T\rangle$ a verification problem, where T consists of source verification tests, only, and every source is testable. Given $Nogood(KB)$, T and an integer α, determining whether there exists a base $X \subseteq T$ for KB such that $k(X) \leq \alpha$ is in P under the* MAXCOST *policy and is NP-complete under the* SUMCOST *policy.*

Let us now show how minimal cost bases for KB can be computed.

Under the MAXCOST **policy**.

Clearly enough, there exists a base X for KB s.t. $k(X) \leq \alpha$ iff $\{t_i \in T \mid k_i \leq \alpha\}$ N-purifies KB, which can be tested in polynomial time. Interestingly, it is possible to derive in polynomial time the smallest α such that $\{t_i \in T \mid k_i \leq \alpha\}$ N-purifies KB. We note k^* this smallest α. To determine the value of k^*, let us consider the mapping `Secondmax` that associates to every finite multi-set of positive integers the second element of the multi-set sorted in decreasing order, and 0 if the multi-set contains less than two elements. For instance, $\texttt{Secondmax}(\{4,6,6,3,9,5,5\}) = 6$, $\texttt{Secondmax}(\{4,7,1,7\}) = 7$, $\texttt{Secondmax}(\{4\}) = 0$. We have:

PROPOSITION 11. (computing k^*)

$$k^* = \max_{N \in Nogood(KB)} \texttt{Secondmax}(\{k_i\}_{\phi_i \in N}).$$

As a direct corollary, we have:

COROLLARY 2. *$\{t_i \in T \mid k_i \leq k^*\}$ is a purifying set for KB that is of minimal cost under the* MAXCOST *policy.*

Such a set is not necessarily a minimal cost base for KB since it is not necessarily minimal w.r.t. $\subseteq$. Nevertheless, applying the greedy algorithm sketched above to this set enables deriving in polynomial time a minimal cost base for KB.

EXAMPLE 7. Let $P = \langle\langle W, \Delta\rangle, T\rangle$ be a verification problem where T consists of source verification tests, only. Let $\Delta = \{\phi_1, \ldots, \phi_7\}$. Assume that

$$Nogood(KB) = \{\{\phi_1, \phi_2, \phi_5\}, \{\phi_2, \phi_6\}, \{\phi_3, \phi_6\}, \{\phi_4, \phi_5, \phi_6\}, \{\phi_7\}\}.$$

Assuming that each test t_i has cost i, we have $k^* = 5$, hence $\{t_i \in T \mid k_i \leq k^*\} = \{t_1, t_2, t_3, t_4, t_5\}$ is a purifying set for KB of minimal cost, and the minimal cost base $\{t_2, t_3, t_4, t_5\}$ for it can be derived from the latter set in polynomial time.

Under the SUMCOST policy.

Since the decision problem associated to the optimisation problem we are interested in is NP-complete, there is little hope that a (deterministic) polynomial time algorithm for computing a minimal cost base for *KB* exists, under the SUMCOST policy. Nevertheless, this does not prevent us from designing algorithms that can exhibit a satisfying computational behaviour for many instances. Interestingly enough, as the next proposition shows it, we can take advantage of numerous algorithms for $0-1$ integer programming in order to achieve our goal:

PROPOSITION 12. (0-1 linear programming) *Let $P = \langle KB, T\rangle$ be a verification problem, where T consists of source verification tests, only, and every source is testable. $X \subseteq T$ is a minimal cost base for KB under the* SUMCOST *policy iff the vector $(x_1, \ldots, x_n)$ such that $x_i = 1$ if $t_i \in X$ and $x_i = 0$ otherwise, is a solution of the following* 0-1 *linear program:*

$$\min(\sum_{i=1}^{n} k_i x_i)$$

$$s.t. \begin{cases} \forall i \in \{1, \ldots, n\}, x_i \in \{0,1\} \\ \forall N \in Nogood(KB), \sum_{\phi_i \in N} x_i \geq \mathrm{card}(N) - 1. \end{cases}$$

EXAMPLE 8. Let $P = \langle\langle W, \Delta\rangle, T\rangle$ be a verification problem where T consists of source verification tests, only. Let $\Delta = \{\phi_1, \ldots, \phi_7\}$. Assume that

$$Nogood(KB) = \{\{\phi_1, \phi_2, \phi_5\}, \{\phi_2, \phi_6\}, \{\phi_3, \phi_6\}, \{\phi_4, \phi_5, \phi_6\}, \{\phi_7\}\}.$$

The $0-1$ linear program when every default has a unary cost is:

$$\min(x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7)$$

$$\text{s.t.} \begin{cases} x_1 + x_2 + x_5 & \geq & 2 \\ x_2 + x_6 & \geq & 1 \\ x_3 + x_6 & \geq & 1 \\ x_4 + x_5 + x_6 & \geq & 2 \\ x_7 & \geq & 0 \end{cases}$$

The following vectors are optimal solutions: $(1, 0, 0, 0, 1, 1, 0)$ and $(0, 1, 0, 0, 1, 1, 0)$. They correspond to the minimal cost bases $\{t_1, t_5, t_6\}$ et $\{t_2, t_5, t_6\}$ for *KB*.

5.3. τ-transformation

From the practical side, in order to compute purifying sets of tests by taking advantage of algorithms for assumption-based reasoning, it can prove convenient to normalize the *KB* under consideration through the following τ-transformation:

Definition 10. (τ-transformation) Let $KB = \langle W, \Delta\rangle$ be an assumption-based theory where $\Delta = \{\phi_1, \ldots, \phi_n\}$. The τ-transformation of KB is the assumption-based theory $\tau(KB) = \langle \tau(W), \tau(\Delta)\rangle$ with $\tau(W) = W \cup \{ok(i) \Rightarrow \phi_i, bad(i) \Rightarrow \neg\phi_i, forget(i) \Leftrightarrow (\neg ok(i) \wedge \neg bad(i)) \mid i \in 1 \,..\, n\}$ where $ok(1)$, $\ldots$, $ok(n)$, $bad(1), \ldots, bad(n), forget(1), \ldots, forget(n)$ are new propositional symbols, not appearing in KB, and $\tau(\Delta) = \{ok(1), \ldots, ok(n)\}$.

Slightly abusing notations, we note $\tau(\phi_i) = ok(i)$ for every $\phi_i \in \Delta$, and $\tau(S) = \{\tau(\phi) \mid \phi \in S\}$ for every $S \subseteq \Delta$.

Example 9. Let $KB = \langle W, \Delta\rangle$ with $W = \emptyset$ and $\Delta = \{a \wedge b, \neg a, \neg b \wedge c, d\}$. $\tau(KB) = \langle \tau(W), \tau(\Delta)\rangle$ with

$$\begin{aligned}
\tau(W) = \quad & (ok(1) \Rightarrow (a \wedge b)) \wedge (bad(1) \Rightarrow \neg(a \wedge b)) \\
& \wedge (forget(1) \Leftrightarrow (\neg ok(1) \wedge \neg bad(1))) \\
& \wedge (ok(2) \Rightarrow \neg a) \wedge (bad(2) \Rightarrow a) \\
& \wedge (forget(2) \Leftrightarrow (\neg ok(2) \wedge \neg bad(2))) \\
& \wedge (ok(3) \Rightarrow \neg b \wedge c) \wedge (bad(3) \Rightarrow \neg(\neg b \wedge c)) \\
& \wedge (forget(3) \Leftrightarrow (\neg ok(3) \wedge \neg bad(3))) \\
& \wedge (ok(4) \Rightarrow d) \wedge (bad(4) \Rightarrow \neg d) \\
& \wedge (forget(4) \Leftrightarrow (\neg ok(4) \wedge \neg bad(4))) \text{ and} \\
\tau(\Delta) = \quad & \{ok(1), ok(2), ok(3), ok(4)\}.
\end{aligned}$$

The τ-transformation is a linear-time transformation that can be considered as a generalization of the notion of σ-transformation given in [9], better suited to the default revision situation. τ-transforming a KB offers many advantages. First, the set of assumptions of a τ-transformed KB is (by construction) a set of variables. This enables representing nogoods and maximal scenarios as terms. Second, it is just a naming trick, so it does not deeply change the nogoods, maximal scenarios and extensions. To be more precise, we need the notion of V-equivalence of two assumption-based theories.

Definition 11. (V-equivalence of two KBs) Let $KB_1 = \langle W_1, \Delta_1\rangle$ and $KB_2 = \langle W_2, \Delta_2\rangle$ be two assumption-based theories, and let $V \subseteq PS$. KB_1 and KB_2 are *V-equivalent*, noted $KB_1 \equiv_V KB_2$, iff there exists a bijective mapping M from $Ext(KB_1)$ to $Ext(KB_2)$ such that for every $E \in Ext(KB_1)$, $E \cap PROP_V = M(E) \cap PROP_V$.

Proposition 13. *Let $KB = \langle W, \Delta\rangle$ with $\Delta = \{\phi_1, \ldots, \phi_n\}$. Let $V = PS \setminus \bigcup_{i=1}^{n}\{ok(i), bad(i), forget(i)\}$. We have $KB \equiv_V \tau(KB)$. To be more precise, we have:*

- *S is a nogood of KB iff $\tau(S)$ is a nogood of $\tau(KB)$.*

- *S is a maximal scenario of KB iff $\tau(S)$ is a maximal scenario of $\tau(KB)$.*
- *Let ψ be any formula not containing any variable from $\{ok(i), bad(i), forget(i)\}_{i\in 1..n}$ and let E_S be the extension of KB generated by the maximal scenario S. $\psi \in E_S$ iff $\psi \in E_{\tau(S)}$, where $E_{\tau(S)}$ is the extension of $\tau(KB)$ generated by $\tau(S)$.*

Additionally, while $\tau(KB)$ can have exponentially many nogoods and exponentially many maximal scenarios, there are situations in which computing $Nogood(\tau(KB))$ or $MaxScen(\tau(KB))$ is easy. Indeed, as the next proposition shows it, computing $MaxScen(\tau(KB))$ is tractable when $\tau(W)$ is a DNF formula. Moreover, when $\tau(W)$ is given by its prime implicates, computing $Nogood(\tau(KB))$ is tractable.

PROPOSITION 14. *Let $KB = \langle W, \Delta\rangle$ be an assumption-based theory.*

- *If $\tau(W) = \psi_1 \vee \cdots \vee \psi_m$ is a DNF formula, then $MaxScen(\tau(KB)) = \{\{ok(i) \in \tau(\Delta) \mid \psi_j \wedge ok(i) \text{ consistent}\} \mid j \in 1..m\}_{\min\supseteq}$.*
- *If $\tau(W) = \psi_1 \wedge \cdots \wedge \psi_p$ is given by the conjunction of its prime implicates, then $Nogood(\tau(KB)) = \{S \subseteq \tau(\Delta) \mid \neg S$ (disjunctively interpreted) is a prime implicate of $\tau(W)\}$.*

Finally, contrariwise to σ-transformed KB, the default revision process for τ-transformed KB can be achieved through a simple expansion of $\tau(W)$.

PROPOSITION 15. *Let $KB = \langle W, \Delta\rangle$ with $\Delta = \{\phi_1, \ldots, \phi_n\}$. Let $V = PS \setminus \bigcup_{i=1}^{n}\{ok(i), bad(i), forget(i)\}$. For any i in $1..n$:*

- *$KB * ok(i) \equiv_V \langle \tau(W) \cup \{ok(i)\}, \tau(\Delta)\rangle$.*
- *$KB * bad(i) \equiv_V \langle \tau(W) \cup \{bad(i)\}, \tau(\Delta)\rangle$.*
- *$KB * forget(i) \equiv_V \langle \tau(W) \cup \{forget(i)\}, \tau(\Delta)\rangle$.*

Accordingly, computing $KB * R$ simply consists in expanding W with R. This proves particularly valuable in the situation where W is a DNF formula (resp. the conjunction of its prime implicates). Indeed, in this situation, a DNF (resp. the set of prime implicates) of $W \cup R$ can be easily computed (see, e.g., [11]), hence it is also the case for $MaxScen(KB * R)$ (resp. $Nogood(KB * R)$). As a consequence, determining whether revising KB with R N-purifies KB can be decided tractably in these two restricted cases.

6. Designing default verification tests

In the formal framework developed in this paper, only tests enabling direct default validation, cancellation or invalidation are considered. At a first glance, this restriction may seem unrealistic since, in the general case, there might be actions of verification that do not check the default ϕ_i itself but a relevant formula (maybe weaker than ϕ_i, maybe stronger, maybe neither of both). For instance, Charlie (he's an engineer) wrote a program C that checks whether an integer is divisible by 2, 3, 5 or 7. A positive answer on 348754697 results in a formula which implies (but is not equivalent to) "348754697 is not prime". Moreover, if Charlie wrote as well a program D that tests whether an integer is divisible by an integer strictly greater than 5, then running C and D not only enables us determining whether "348754697 is prime" but gives us as well extra-information which might be relevant for another default of Δ.

In the following, we show how default verification tests can be generated from more general tests.

We propose the following propositional formulation of a test. More complex formulations can be found in the literature [14], [21], [24] and [12].

DEFINITION 12. (general tests) A *test* t is a (finite) set of propositional formulas $\{\psi_1, \ldots, \psi_m\}$ where $m \geq 2$. Each ψ_i represents a possible outcome of the test. To each test t is attached a positive integer k, the cost of performing t.

A *test environment* is an ordered pair $\langle W, T\rangle$ where W is a finite set of propositional formulas (facts) and $T = \{t_1, \ldots, t_p\}$ is a finite set of available tests. An *outcome of* T is a set composed of an outcome ψ^i_j of each of the tests t_i of T.

We make the following assumptions:

- Tests always give *a (unique) outcome*. This assumption is technical and does not mean that tests do not have preconditions or never fail: simply, when the test is not executable or when it fails, its outcome will be the tautology $\top$.
- Tests are *reliable*: if the outcome of $t \in T$ is ψ_i, then ψ_i is true in I. We can assume without loss of generality that for all $i \in 1..m$, $W \cup \{\psi_i\}$ is consistent — otherwise, ψ_i will never be reached as an outcome and is thus irrelevant.
- Tests are *non-intrusive*: performing t does not change the actual world I. Subsequently, the order with respect to which tests are performed does not matter: the outcome R of a set $X \subseteq T$ of tests is simply the set of the outcomes of each individual test t of X. $State(X)$ is the set of all possible resulting outcomes of a set X of tests.

Note that we do not require tests to have exclusive outcomes. This enables taking into account tests that may succeed or fail in a graduate way.

For instance, $t = \{a, a \vee b, \top\}$ used for determining the truth value of a can either succeed (the outcome is a), fail completely (the outcome is $\top$: no information) or only partially (the outcome $a \vee b$ is relevant to a). The verification actions defined in Section 3 are special cases of tests. Namely, $\{ok(i), bad(i)\}$, $\{ok(i), forget(i)\}$ and $\{ok(i), bad(i), forget(i)\}$ respectively correspond to the tests $\{\phi_i, \neg\phi_i\}$, $\{\phi_i, \top\}$ and $\{\phi_i, \neg\phi_i, \top\}$.

Sets of tests can be characterized according to their ability to help in determining the truth value of a given formula ϕ_i (for our purposes, ϕ_i is a default of Δ):

DEFINITION 13. ((in)validation, discrimination) Let $\langle W, T\rangle$ a test environment and ϕ_i be a formula from $PROP_{PS}$:

- T *potentially validates* ϕ_i iff at least one of its outcomes R is such that $W \cup R \models \phi_i$. T *potentially invalidates* ϕ_i iff it potentially validates $\neg\phi_i$.
- T *discriminates* ϕ_i iff for each of its outcomes R, we have either $W \cup R \models \phi_i$ or $W \cup R \models \neg\phi_i$.

We now briefly show how to generate default verification tests from a set $T = \{t_1, \ldots, t_p\}$ of more "general" tests: if T discriminates ϕ_i (which is the most favorable case), then ϕ_i is associated to the test $\{ok(i), bad(i)\}$; else, if T potentially validates and potentially invalidates ϕ_i, then the associated test is $\{ok(i), bad(i), forget(i)\}$; if T only potentially validates ϕ_i, then the test is $\{ok(i), forget(i)\}$; if T does not potentially validate ϕ_i (even if it potentially invalidates it), then no default verification test is associated to ϕ_i (ϕ_i is not testable). Note that it would not cause any technical difficulty to consider an action $\{bad(i), forget(i)\}$ when T only potentially invalidates ϕ_i; but, as explained previously, the assumption that a default is plausible makes them of little relevance.

When tests of T have costs, a default verification test generated from T as described above has an associated cost too. For instance, the cost of a $\{ok(i), bad(i)\}$ default verification test is the cost k of a set $X \subseteq T$ of minimal cost that discriminates ϕ_i (note that k is obtained by aggregating the costs of the individual tests of X). Several aggregation policies and a trade-off between the costs and the discrimination power of tests can be considered.

EXAMPLE 10. Let $\langle W, T\rangle$ be a test environment s.t. $W = \{a \vee b, b \Leftrightarrow c, a \wedge b \Rightarrow f, b \vee g, a \vee \neg f, a \vee \neg g\}$ and $T = \{t_a, t_b, t_c\}$ where $t_a = \{a, \neg a\}$, $t_b = \{b, \neg b\}$ and $t_c = \{c, \neg c\}$. Let $\phi_1 = f$, $\phi_2 = f \vee g$, $\phi_3 = f \wedge \neg g$.

- T both potentially validates and potentially invalidates ϕ_1 (consider respectively $R = \{a, b, c\}$ and $R' = \{\neg a, b, c\}$), but does not discriminate ϕ_1 (consider $R'' = \{a, \neg b, \neg c\}$).

- T discriminates ϕ_2. Since $W \models (\phi_2 \Leftrightarrow a)$, $\{t_a\}$ discriminates ϕ_2 (hence T discriminates ϕ_2 as well).
- T does not potentially validate ϕ_3 (but it potentially invalidates it).

Clearly enough, the complexity of designing default verification tests from more general tests merely depends on the complexity of determining whether T potentially validates ϕ_i (resp. discriminates it). We call the corresponding decision problems VALIDABILITY (resp. DISCRIMINABILITY).[2]

PROPOSITION 16. VALIDABILITY and DISCRIMINABILITY

- VALIDABILITY *is* Σ_2^p*-complete.*
- DISCRIMINABILITY *is* coNP*-complete.*

It is particularly interesting to note that discriminability (which is a stronger, and perhaps more useful notion than validability and invalidability) is also computationally easier. Introducing a name for each test outcome (under the form of a new variable) and binarizing tests with more than two possible outcomes enables characterizing the sets of tests we are looking for using prime implicates. The validability (resp. discriminability) issue is strongly related to logic-based abduction (resp. definability), and it is possible to take advantage of the complexity results from [6] (resp. [10]) to derive the results above.

7. Related work and conclusion

This paper is (to our knowledge) the first approach aiming at "debugging" default theories by performing appropriate default verification actions, but is is closely connected (and often complementary) to the following areas: tests in diagnosis, revision of default theories and inference from default theories.

The model-based diagnosis community has been interested for long in test selection strategies for isolating faulty components. Discriminating among several extensions and discriminating among plausible diagnoses are two almost identical tasks: the parallel between the complements of maximal scenarios (often called candidates) and preferred (consistency-based) diagnoses is well-known [19, 5]. A first class of approaches, along the works on GDE [4], consists in selecting the next best test to perform (*one-step lookahead* strategy) so as to maximize the expected gain of information. Much closer to our formalization of tests is the work of McIlraith and Reiter [13, 12] where tests are classified in function of their potentiality to confirm or discard one

[2] Obviously, the complexity of INVALIDABILITY and of VALIDABILITY coincide.

or several hypotheses; the search for "good" tests is characterized in abductive terms and consequently, test generation can be automated using results and algorithms for the computation of abductive explanations. This line of work (focusing on test generation) can be thought of as upstream whereas ours (rather concerned with what can be done with tests once they are fixed) is downstream.

Williams and Antoniou [25] propose a methodology for revising default theories. They consider four types of revision, namely (a) contraction (removal of a fact), (b) denial (ensuring that a given formula appears in no extension), (c) incorporation of information consistent with W and (d) incorporation of information inconsistent with W. Their revisions lead to change W while ours lead to change both Δ and W. Our validations, consisting of "moving" a formula from Δ to W, are a special case of (c) — no matter whether the default remains in Δ or not. Our invalidations are also a special case of (c) (note that we never question the truth of formulas of W, neither the outcomes of tests, which guarantees that test outcomes are consistent with W). Lastly, our cancellations are a different type of contraction than (a) and they induce a smaller amount of change than fact contraction.

There is a large amount of work on inference from default theories [2], and, relatedly, on belief base revision [15] and on reasoning from inconsistent knowledge bases [3]. There is much to say, first, on the effects of tests outcomes on inference: to be short, default validations strengthen skeptical inference and weaken credulous inference, cancellations weaken credulous inference and have various kinds of effects on skeptical inference (depending on the "role" played by the cancelled default before its cancellation); things are more complex with invalidations. Now, there is also much to say and to do about the *cooperation* between both tasks (inference and discrimination). Inference enables pointing on relevant tests: to give an example, it can be shown that a test $\{ok(i), forget(i)\}$ is useful for discrimination iff the corresponding ϕ_i belongs to at least one extension but not all, i.e., if it is credulously inferred but not skeptically. Other questions of interest are: until when should we discriminate among extensions so as to make the inference informative enough and/or sound enough? Which verification actions should we focus on when we are no more interested in the whole purification task but rather in the inference (or not) of a particular formula? Some of our results can be considered as a first step towards these issues.

Acknowledgements

The second author has been partly supported by a "Contrat d'objectifs de la Région Nord/Pas-de-Calais" and by the IUT de Lens.

Appendix

Proof of Proposition 1. Recall that $KB * ok(i) = \langle W \cup \{\phi_i\}, \Delta \setminus \{\phi_i\}\rangle$.

1. ($\subseteq$) Let $N \in Nogood(KB * ok(i))$, which means that N is a minimal (w.r.t. $\subseteq$) subset of $\Delta \setminus \{\phi_i\}$ such that $N \cup W \cup \{\phi_i\}$ is inconsistent. Let us first show that $N \in \{N' \setminus \{\phi_i\} \mid N \in Nogood(KB)\}$. Two cases are to be considered:

Case 1: $N \cup W$ is inconsistent.

Then, N contains a nogood N' of KB. If N' were strictly included in N, then N would not be a nogood of $KB * ok(i)$ (because it would not be minimal w.r.t. $\subseteq$); hence, $N' = N = N' \setminus \{\phi_i\}$, which means that $N \in \{N' \setminus \{\phi_i\} \mid N' \in Nogood(KB)\}$.

Case 2: $N \cup W$ is consistent.

Since $N \cup W \cup \{\phi_i\}$ is inconsistent, $N \cup \{\phi_i\}$ contains a nogood N' of KB. Since $N \cup W$ is consistent, N' necessarily contains ϕ_i. If N' were strictly included in $N \cup \{\phi_i\}$ then N would not be a nogood of $KB * ok(i)$; therefore, $N' = N \cup \{\phi_i\}$ and $N \in \{N' \setminus \{\phi_i\} \mid N' \in Nogood(KB)\}$.

In both cases, if N were not minimal w.r.t $\subseteq$ in $\{N' \setminus \{\phi_i\} \mid N' \in Nogood(KB)\}$ then there would exist a $N'' \subset N$ such that $N'' \cup W$ is inconsistent, which would contradict the fact that $N \in Nogood(KB * ok(i))$ (because it would not be minimal). Therefore, $N \in \{N' \setminus \{\phi_i\} \mid N' \in Nogood(KB)\}_{\min\subset}$.

($\supseteq$) Let $N \in Nogood(KB)$.

Case 1: $\phi_i \in N$.

Then $N \setminus \{\phi_i\}$ contains a nogood N' of $KB * ok(i)$; if N' were strictly included in $N \setminus \{\phi_i\}$, then $N' \cup \{\phi_i\}$ would contain a nogood of KB, which would contradict $N \in Nogood(KB)$ because minimality would be violated. Therefore, $N \setminus \{\phi_i\} \in Nogood(KB * ok(i))$.

Case 2: $\phi_i \notin N$.

Then N contains a nogood N' of $KB * ok(i)$. Either $N' \subset N$, and in this case $N \setminus \{\phi_i\}$ is not minimal in $\{N' \setminus \{\phi_i\} \mid N' \in Nogood(KB)\}$; or $N' = N$, and then $N \setminus \{\phi_i\} \in Nogood(KB * ok(i))$.

Thus, for any $N \in Nogood(KB)$, if $N \setminus \{\phi_i\} \in \{N' \setminus \{\phi_i\} \mid N' \in Nogood(KB)\}_{\min\subset}$, then $N \setminus \{\phi_i\} \in Nogood(KB * ok(i))$.

2. ($\subseteq$) Let $S \in MaxScen(KB * ok(i))$. Then $W \cup \{\phi_i\} \cup S$ is maximal (w.r.t. $\subseteq$) consistent. Therefore, $S \cup \{\phi_i\} \in MaxScen(KB)$.

($\supseteq$) Let $S \in MaxScen(KB)$ such that $\phi_i \in S$. Then $W \cup \{\phi_i\} \cup (S \setminus \{\phi_i\})$ is maximal (w.r.t. $\subseteq$) consistent, which means that $S \setminus \{\phi_i\} \in MaxScen(KB * ok(i))$.

3. Straightforward from point 2. above and the correspondance between extensions and maximal scenarios (the extensions of $KB * ok(i)$) are generated by the maximal scenarios of KB containing ϕ_i). ■

Proof of Proposition 2. Recall that $KB * \mathit{forget}(i) = \langle W, \Delta \setminus \{\phi_i\}\rangle$.

1. ($\subseteq$) Let $N \in \mathit{Nogood}(KB * \mathit{forget}(i))$. $N \cup W$ is inconsistent, thus, N contains a nogood N' of KB. If N' were strictly included in N, then N would not be a nogood of $KB * \mathit{forget}(i)$ because it would not be minimal w.r.t. $\subseteq$. Therefore, $N' = N$ and $N \in \{\mathit{Nogood}(KB) \mid \phi_i \notin N\}$.

($\supseteq$) Let $N \in \mathit{Nogood}(KB)$ such that $\phi_i \notin N$. $N \cup W$ is inconsistent, thus, N contains a nogood N' of $KB * \mathit{forget}(i)$. If N' were strictly included in N, then N would not be a nogood of KB because it would not be minimal w.r.t. $\subseteq$. Therefore, $N' = N$ and $N \in \mathit{Nogood}(KB * \mathit{forget}(i))$.

2. ($\subseteq$) Let $S \in \mathit{MaxScen}(KB * \mathit{forget}(i))$.

If $S \cup \{\phi_i\} \cup W$ is consistent, then $S \cup \{\phi_i\}$ is contained in a maximal scenario S' of KB; if S' were a strict superset of S, then S would not be maximal w.r.t. $\subseteq$ and thus would not be in $\mathit{MaxScen}(KB * \mathit{forget}(i))$. Therefore, $S' = S$.

If $S \cup \{\phi_i\} \cup W$ is inconsistent then $S \setminus \{\phi_i\}$ is contained in a maximal scenario S'' of KB not containing ϕ_i. If S'' were a strict superset of $S \setminus \{\phi_i\}$, then S would not be a maximal scenario of $KB * \mathit{forget}(i)$. Therefore, $S'' = S \setminus \{\phi_i\}$.

In both cases, $S \setminus \{\phi_i\} \in \mathit{MaxScen}(KB)$. With a line of reasoning similar to that of point 1. of Proposition 1, it is then easy to prove that $S \in \{S \setminus \{\phi_i\} \mid S \in \mathit{MaxScen}(KB)\}_{\min \supset}$.

($\supseteq$) Let $S \in \mathit{MaxScen}(KB)$.

If $\phi_i \notin S$, then S is also a maximal scenario of $KB * \mathit{forget}(i)$.

If $\phi_i \in S$, then $S \setminus \{\phi_i\}$ is contained in a maximal scenario S' of $KB * \mathit{forget}(i)$. Either $S' \supset S \setminus \{\phi_i\}$ and in this case $S \setminus \{\phi_i\}$ is not maximal in $\{S'' \setminus \{\phi_i\} \mid S'' \in \mathit{MaxScen}(KB)\}$; or $S' = S \setminus \{\phi_i\}$ and in this case $S \setminus \{\phi_i\} \in \mathit{MaxScen}(KB * \mathit{forget}(i))$.

3. Let $E \in \mathit{Ext}(KB * \mathit{forget}(i))$ and let S be its generating scenario. After point 2. above, $S = S' \setminus \{\phi_i\}$ for some $S' \in \mathit{MaxScen}(KB)$. Now, we have $E \setminus \mathit{Cn}(W \cup S')$ and $\mathit{Cn}(W \cup S') \in \mathit{Ext}(KB)$, hence the result follows.■

Proof of Proposition 3.

- We have $\mathit{GNogood}(KB * ok(i)) = \mathit{GNogood}(\langle W \cup \{\phi_i\}, \Delta \setminus \{\phi_i\}\rangle) = \mathit{Nogood}(\langle W \cup \{\phi_i\}, (\Delta \setminus \{\phi_i\}) \cup \{\neg\phi_j \mid \phi_j \in \Delta \setminus \{\phi_i\}\}\rangle)$. As a consequence of point 1. of Proposition 1, we obtain $\mathit{GNogood}(KB * ok(i)) = \{N \setminus \{\phi_i\} \mid N \in \mathit{Nogood}(\langle W, \Delta \cup \{\neg\phi_j \mid \phi_j \in \Delta \setminus \{\phi_i\}\}\rangle)\}_{\min \subset}$. As a consequence of point 1. of Proposition 2, we have $Nogood(\langle W, \Delta \cup \{\neg\phi_j \mid \phi_j \in \Delta \setminus \{\phi_i\}\}\rangle) = \{N \in Nogood(\langle W, \Delta \cup \{\neg\phi_j \mid \phi_j \in \Delta\}\rangle) \mid \neg\phi_i \notin N\}$. Hence, we have $GNogood(KB * ok(i)) = \{N \setminus \{\phi_i\} \mid N \in Nogood(\langle W, \Delta \cup \{\neg\phi_j \mid \phi_j \in \Delta\}\rangle)$ and $\neg\phi_i \notin N\}_{\min\subset} = \{N \setminus \{\phi_i\} \mid N \in GNogood(\langle W, \Delta\rangle)$ and $\neg\phi_i \notin N\}_{\min\subset}$.

- We have $GNogood(KB * forget(i)) = GNogood(\langle W, \Delta \setminus \{\phi_i\}\rangle) = Nogood(\langle W, (\Delta \setminus \{\phi_i\}) \cup \{\neg\phi_j \mid \phi_j \in \Delta \setminus \{\phi_i\}\}\rangle)$. As a consequence of point 1. of Proposition 2, we obtain $GNogood(KB * forget(i)) = \{N \in Nogood(\langle W, \Delta \cup \{\neg\phi_j \mid \phi_j \in \Delta\}\}\rangle) \mid \phi_i \notin N$ and $\neg\phi_i \notin N\} = \{N \in GNogood(\langle W, \Delta\rangle) \mid \phi_i \notin N$ and $\neg\phi_i \notin N\}$.

- Replacing in a uniform way ϕ_i by $\neg\phi_i$ in the proof of point 1. above is sufficient. ■

PROOF OF PROPOSITION 4. (a) $\Rightarrow$ (c) If R C-purifies KB, then $Ext(KB) \cap Ext(KB * R) = \{E\}$. (i) E necessarily contains $\{\phi_i \mid i \in Ok(R)\}$ because ϕ_i is implied by the factual part of $KB * R$. (ii) Since E is an extension of KB, E contains either ϕ_i or $\neg\phi_i$ for each i. For all $i \in Bad(R)$, E contains $\neg\phi_i$, because otherwise it would contain ϕ_i and would be inconsistent with R. Therefore, E contains $Facts_P(R)$. (iii) Let E' be an extension of KB containing $Facts_P(R)$. Then E' is an extension of $KB * R$ as well, and since $Ext(KB) \cap Ext(KB * R) = \{E\}$, we get $E' = E$.

(c) $\Leftrightarrow$ (d) Assume that E' is an extension of KB consistent with $Facts_P(R)$. By maximality of extensions, any extension E of KB that is *consistent with* $Facts_P(R)$ *contains* $Facts_P(R)$, and the converse is straightforward. The equivalence between (c) and (d) comes easily from the latter equivalence.

(d) $\Rightarrow$ (e) $Compl_\Delta(W)$ is the disjunction of all extensions of KB. If there is a single extension E of KB consistent with $Facts_P(R)$ then $Compl_\Delta(W) \wedge Facts_P(R)$ is equivalent to $E \wedge Facts_P(R)$, therefore $\forall i$, either $\phi_i \in E$ and $Compl_\Delta(W) \wedge Facts_P(R) \models \phi_i$, or $\neg\phi_i \in E$ and $Compl_\Delta(W) \wedge Facts_P(R) \models \neg\phi_i$.

(e) $\Rightarrow$ (b) We get immediately that $Ext(Compl(KB)*R) = \{E\}$ such that E is generated by the maximal scenario $\{\phi_i \mid Compl_\Delta(W) \wedge Facts_P(R) \models \phi_i\}$. Thus, $Compl(KB) * R$ has one and only one extension.

(b) $\Rightarrow$ (a) Let E be the single extension of $Compl(KB) * R$. It remains to be proven that E is an extension of KB. Let $S = \{\phi_i \mid \phi_i \in E\}$. S is consistent with KB since it is consistent with its completion. S is maximal w.r.t. $\subseteq$, otherwise E would not be maximal. Therefore S is a maximal scenario of KB and E, generated by S, is an extension of KB. ■

PROOF OF PROPOSITION 5. We recall that R N-purifies KB iff $KB * R$ has a unique extension. Thus, the complexity of checking whether R N-purifies

KB is the complexity of deciding whether an assumption-based theory has one and only one extension. Now, an assumption-based theory $KB = \langle W, \Delta \rangle$ has one and only one extension iff (a) W is satisfiable and (b) $W \cup \{\phi_i \in \Delta \mid \{\phi_i\} \cup W$ is satisfiable$\}$ is satisfiable.

• Membership to $\Delta_2^p[\mathcal{O}(n)]$: verifying (a) and (b) requires card(Δ) + 2 satisfiability checks, hence the membership to $\Delta_2^p[\mathcal{O}(n)]$ when the input is KB and R.

• NP-hardness: by reduction from SAT. We consider the following polynomial reduction F associating to any propositional formula Σ the assumption-based theory $F(\Sigma) = \langle \emptyset, \{\Sigma \vee new, \Sigma \vee \neg new\} \rangle$, where $new \notin Var(\Sigma)$. Σ is satisfiable iff $F(\Sigma)$ has a unique extension.

• coNP-hardness: by reduction from UNSAT. We consider the following polynomial reduction G associating to any propositional formula Σ the assumption-based theory $G(\Sigma) = \langle \emptyset, \{\Sigma \wedge new, \Sigma \wedge \neg new\} \rangle$, where $new \notin Var(\Sigma)$. Σ is not satisfiable iff $G(\Sigma)$ has a unique extension. ■

PROOF OF PROPOSITION 6.

• Membership to Π_2^p: we prove the membership of the complementary problem to Σ_2^p by the following nondeterministic algorithm: *guess* $R \in State(X)$*; show that* $KB * R$ *has a unique extension.* The latter step needs only a polynomial number of calls to an NP oracle (see above), hence the membership to Σ_2^p.

• Π_2^p-hardness: by reduction from $QBF_{2,\exists}$ to the complementary problem. We consider the following polynomial reduction F associating to a formula of the type $\exists A \forall B \Phi$ (with $A = \{a_1, \ldots, a_n\}$) the pair $\langle X, KB \rangle$ with $X = \{c_1, non{-}c_1, \ldots, c_n, non{-}c_n\}$ (X verifies $X \cap Var(\Phi) = \emptyset$) and $KB = \langle W, \Delta \rangle$ with $W = \{c_1 \Rightarrow a_1, non - c_1 \Rightarrow \neg a_1, \ldots, c_n \Rightarrow a_n, non - c_n \Rightarrow \neg a_n, \Phi \Rightarrow d\}$, $\Delta = \{c_1, non - c_1, \ldots, c_n, non - c_n, d \Rightarrow e, d \Rightarrow \neg e\}$ (thus we have as well $\{d, e\} \cap Var(\Phi) = \emptyset$), and for each default, $t_i = \{ok(i), forget(i)\}$. Let us show that $\exists A \forall B \Phi$ is valid iff X does not purify KB. Suppose that $\exists A \forall B \Phi$ is valid. Then, there exists an A-world $\vec{a}$ such that $\vec{a} \models \Phi$. Consider the state R s.t. $i \in Ok(R)$ iff $\phi_i = c_i^*$ and $i \in Forget(R)$ iff $\phi_i = \bar{c}_i^*$ for every $i \in 1..n$ (accordingly, the two defaults $d \Rightarrow e$ and $d \Rightarrow \neg e$ are not tested). Here, $c_i^* = c_i$ if $\vec{a}_i = 1$ and $c_i^* = non - c_i$, if $\vec{a}_i = 0$ (for $i \in 1..n$). Clearly enough, we have $R \in State(X)$. Now, R is consistent since $W \wedge \{\phi_i \mid i \in Ok(R)\}$ is. It is easy to see that $KB * R$ has two extensions. Indeed, since $\vec{a} \models \Phi$ is verified, we have $W \cup \{\phi_i \mid i \in Ok(R)\} \models d$, hence $KB * R$ has two extensions (the first one contains e, the other one $\neg e$). Reciprocally, suppose that X does not purify KB. All the possible states $R \in State(X)$ share the fact that $d \Rightarrow e$ and $d \Rightarrow \neg e$ have not been tested. Now, $KB * R$ has two extensions iff

$W \cup \{\phi_i \mid i \in Ok(R)\} \models d$. This is possible only if there exists an A-world $\vec{a}$ such that $W \cup \{\phi_i \mid i \in Ok(R)\} \models \vec{a}$ and $\vec{a} \models \Phi$. The existence of such a $\vec{a}$ shows the validity of $\exists A \forall B \Phi$. ∎

PROOF OF PROPOSITION 7. 1. R N-purifies KB iff $KB * R$ only has one extension iff every nogood of $KB * R$ contains at most one element. As a straightforward consequence of Propositions 1 and 2, we have $Nogood(KB * R) = \{N \setminus \{\phi_i\} \mid N \in Nogood(KB)$ and $Index(N) \cap Forget(R) = \emptyset$ and $i \in Ok(R)\}_{\min\subset}$. Accordingly, every nogood of $Nogood(KB * R)$ contains at most one element iff for every nogood N of KB, if $Index(N) \cap Forget(R) = \emptyset$, then there exists a nogood N' of KB s.t. $Index(N) \cap Forget(R) = \emptyset$ and $N' \subseteq N$ and N' contains at most one element.

2. If every nogood contains at most one non-testable source j, then for every state $R \in State(X)$, for every $N \in Nogood(KB)$, either R contains $forget(i)$ for at least one $i \in Index(N)$ or R contains $ok(i)$ for every $i \in Index(N) \setminus \{j\}$. Since $Nogood(KB * R) = \{N \setminus \{\phi_i\} \mid N \in Nogood(KB)$ and $Index(N) \cap Forget(R) = \emptyset$ and $i \in Ok(R)\}_{\min\subset}$, we obtain that for every state $R \in State(X)$, every nogood from $Nogood(KB * R)$ contains at most one element. Accordingly, X N-purifies KB. Conversely, assume that there exists a nogood N of KB s.t. $Index(N)$ contains at least two non-testable sources s_1 and s_2. Then, consider the state R for which every testable source of $Index(N)$ is in $Ok(R)$ and all the remaining testable sources are in $Forget(R)$. Clearly enough, since $Nogood(KB * R) = \{N \setminus \{\phi_i\} \mid N \in Nogood(KB)$ and $Index(N) \cap Forget(R) = \emptyset$ and $i \in Ok(R)\}_{\min\subset}$, we have $\{s_1, s_2\} \in Nogood(KB * R)$; indeed, by construction, $\{\phi_{s_1}, \phi_{s_2}\} \in \{N \setminus \{\phi_i\} \mid N \in Nogood(KB)$ and $Index(N) \cap Forget(R) = \emptyset$ and $i \in Ok(R)\}$; hence, $\{\phi_{s_1}, \phi_{s_2}\}$ would not belong to $Nogood(KB * R)$ if it were the case for a proper subset of $\{\phi_{s_1}, \phi_{s_2}\}$. But this would be possible only if a nogood N' of KB s.t. $Index(N')$ contains only sources of $Index(N)$ and not both s_1 and s_2 would exist. Hence, we would have $N' \subset N$, and this would contradict the fact that N is a nogood of KB. ∎

PROOF OF COROLLARY 1.

- This is a consequence of point 2. of Proposition 7. Indeed, if every nogood N of KB contains at most one ϕ_i s.t. $t_i \notin X$, N also contains at most one ϕ_i s.t. $t_i \notin X'$, for every superset X' of X.

- The "if" way is trivial. The "only if" way is a immediate consequence of point 1. of this corollary, since if T does not N-purifies KB, none of its subsets does it. ∎

PROOF OF PROPOSITION 8. Let $t_i \in T$.

($1 \Rightarrow 3$) If T does not purify KB then $Base(KB) = \emptyset$ and i is not necessary. If $T\setminus\{t_i\}$ purifies KB then $\exists B \in Base(KB)$ such that $B \subseteq T\setminus\{t_i\}$ and i is not necessary.

($3 \Rightarrow 2$) Assume that T purifies KB and $T\setminus\{t_i\}$ does not. Since T purifies KB, we know from Proposition 7 that every nogood N of KB contains at most a source that is not in T. Suppose now that (2) does not hold, which means that any nogood N of KB is such that either $\phi_i \notin N$ or $Index(N) \cap \{j \in \{1,\ldots,n\} \mid t_j \notin T\}$ is empty or $|Index(N) \cap \{j \in \{1,\ldots,n\} \mid t_j \notin T\}| \geq 2$. The latter case is not possible (because N contains at most a source that is not in T). When $\phi_i \notin N$, N contains at most a source that is not in $T \setminus \{t_j\}$. When $Index(N) \cap \{j \in \{1,\ldots,n\} \mid t_j \notin T\}$ is empty, N contains at most a source (namely i) that is not in $T \setminus \{t_j\}$. Thus, using again Proposition 7, we get that $T \setminus \{t_i\}$ purifies KB, which leads to a contradiction.

($2 \Rightarrow 1$) Assume that

(a) $Base(KB) \neq \emptyset$,

(b) there exists a nogood N of KB such that $\phi_i \in N$ and $Index(N) \cap \{j \in \{1,\ldots,n\} \mid t_j \notin T\}$ is a singleton $\{t_k\}$,

(c) i is not necessary.

From (a) and (c), there exists a $B \in Base(KB)$ such that $t_i \notin B$. Thus, $Index(N)\setminus Index(B)$ contains k (from (b)) and i, hence $|Index(N)\setminus Index(B)| \geq 2$, and from Proposition 7, we conclude that B does not purify KB, which contradicts $B \in Base(KB)$. ∎

PROOF OF PROPOSITION 9. Let $NEC = \{\phi_i \mid i \in Nec(P)\}$. We first prove the following lemma:

LEMMA. *If KB is purifiable then X purifies KB' if and only if $Y = X \cup NEC$ purifies KB.*

PROOF OF LEMMA. If KB is not purifiable then $Base(KB)$ and $NEC = \emptyset$, thus $Base(KB')$ is empty and the lemma is verified. Assume now that KB is purifiable.

($\Leftarrow$) Assume that $Y = X \cup NEC$ purifies KB, and let $N' \in Nogood(KB')$ such that $N' \cap NEC = \emptyset$. By construction of $Nogood(KB')$ from the `Simplify` procedure, $\exists N \in Nogood(KB)$ such that $N = N' \cup N''$, with $N' \cap NEC = \emptyset$ and $N'' \subseteq NEC$. Since Y purifies KB, by Proposition 7 we get that Y contains at least $|N| - 1$ elements of N; hence X contains at least $|N'| - 1$ elements of N'. Since this is true for any nogood N' of KB', using again Proposition 7 we get that X purifies KB'.

($\Rightarrow$) Assume that X purifies KB', and let $N \in \mathit{Nogood}(KB)$ and $N' = N \setminus \mathit{NEC}$. If $N' \notin \mathit{Nogood}(KB')$ then it has been filtered out during the procedure because $N \setminus \mathit{NEC} = \{\phi_i\}$ or because $N \setminus \mathit{NEC} = \emptyset$, and in both cases $Y = X \cup \mathit{NEC}$ contains at least $|N - 1|$ elements of N. If $N' \in \mathit{Nogood}(KB')$ then, since X purifies KB', by Proposition 7 we get that X contains at least $|N'| - 1$ elements of N' and therefore $Y = X \cup \mathit{NEC}$ contains at least $|N| - 1$ elements of N. In all cases, the key property of Proposition 7 is satisfied, therefore Y purifies KB. □

The proof of Proposition 9 follows easily from the lemma and the equivalence between the minimality of X and the minimality of Y. ■

Proof of Proposition 10.

- MAXCOST: it is sufficient to observe that there exists a base X for KB s.t. $k(X) \leq \alpha$ iff $\{t_i \mid k_i \leq \alpha\}$ purifies KB. Proposition 7 shows that the latter test can be achieved in time polynomial in $|\mathit{Nogood}(KB)|$.
- SUMCOST:

Membership to NP: here is a nondeterministic algorithm that runs in polynomial time for checking whether there exists a base X for KB s.t. $k(X) \leq \alpha$: guess $X \subseteq T$, check that X is a base for KB and check that $k(X) \leq \alpha$.

NP-hardness: here is a polynomial reduction from the well-known NP-complete problem HITTING SET [8] to our problem. HITTING SET is the following decision problem:

- Input: a set S, a set $C \subseteq 2^S$ s.t. every element c of C contains at most two elements of S, and a non-negative integer k.
- Question: is there a subset H of S s.t. $\forall c \in C$, $H \cap c \neq \emptyset$ and $\mathrm{card}(H) \leq k$?

We associate to every $\langle S, C, k\rangle$ the ordered pair $\langle X, \mathit{Nogood}(KB), k\rangle$ s.t. $\mathit{Nogood}(KB) = \{\{\phi_i \mid i \in c\} \mid c \in C\}$. There exists a hitting set H of C containing at most α elements iff there exists a base $X = \{t_i \mid i \in H\}$ for $KB = \langle \emptyset, \{\phi_i \mid i \in S\}\rangle$ s.t. $k(H) \leq \alpha$ provided that every $k_i = 1$ ($i \in S$). Indeed, since every nogood of KB contains at most 2 elements, for every $c \in C$, we have $H \cap c \neq \emptyset$ iff $\mathrm{card}(c \setminus H) \leq 1$. Then, Proposition 7 enables to conclude the proof. ■

Proof of Proposition 11. Let X be the set of tests computed in the following way: for every nogood N of KB, we add to X (initialized as $\emptyset$) every test t_i s.t. ϕ_i belongs to N, except a most expensive one (one for which k_i is maximal in $\{k_i \mid i \in \mathit{Index}(N)\}$). Proposition 7 shows that X is a purifying set for KB. The cost of X is $k(X) = \max_{N \in \mathit{Nogood}(KB)} \mathtt{Secondmax}_{i \in \mathit{Index}(N)} k_i$.

Accordingly, we have $k(X) \geq k^*$. Now, let X' be any subset of T s.t. $k(X') < k(X)$. Necessarily, there exists a conflict N of KB that contains at least two tests t_i and t_j s.t. neither i nor j belongs to $Index(N)$. In this situation, Proposition 7 shows that X' is not a purifying set for KB. This concludes the proof. ■

PROOF OF COROLLARY 2. Trivial from the definition of k^* and Proposition 11. ■

PROOF OF PROPOSITION 12. Let $F\colon \{0,1\}^n \to 2^T$ such that for any vector $\vec{x}$ of $\{0,1\}^n$, $F(\vec{x}) = \{t_i \mid x_i = 1\}$. Proposition 7 implies that $F(\vec{x})$ purifies KB if and only if

$$(\mathcal{C}) \qquad \forall N \in Nogood(KB),\ \textstyle\sum_{\phi_i \in N} x_i \geq |N| - 1$$

Moreover, we easily get that $k(F(\vec{x})) = \sum_{i=1}^{n} k_i x_i$. Therefore, X is a minimal cost base, i.e., X purifies KB and $k(X)$ is minimal, if and only if $\vec{x} = F^{-1}(X)$ satisfies the set of constraints $(\mathcal{C})$ and minimizes $\sum_{i=1}^{n} k_i x_i$. ■

PROOF OF PROPOSITION 13. We note $\bar{V} = \bigcup_i \{ok(i), bad(i), forget(i)\}$. Recall that we have $E_S = Cn(W \cup S)$ and $E_{\tau(S)} = Cn(\tau(W) \cup \tau(S))$. By construction, $\tau(W) \models W$ and $\tau(\phi_i) \wedge W \models \phi_i$; hence, $\tau(W) \cup \tau(S) \models W \cup S$ and thus

(1) $E_S \subseteq E_{\tau(S)}$.

Let M_V be a interpretation on the variables of V such that $M_V \models E_S$. Clearly enough,

(2) the interpretations (on $V \cup \bar{V}$) of $E_{\tau(S)}$ extending M_V are $\{M_V \cup \{ok(i) \mid \phi_i \in S\} \cup \{x(i) \mid \phi_i \notin S\}\}$ where $x(i) = forget(i)$ or $x(i) = bad(i)$.

Now, let ψ be a formula of $PROP_V$ such that $\psi \in E_{\tau(S)}$. Let M_V be a models of E_S. Let us show that $M_V \models \psi$.

Let $M = M_V \cup M_{\bar{V}}$ such that $M \models E_{\tau(S)}$. By (2) we have $M = M_V \cup \{ok(i) \mid \phi_i \in S\} \cup \{x(i) \mid \phi_i \notin S\}\}$ with $x(i) = forget(i)$ or $x(i) = bad(i)$. Then, $M \models E_{\tau(S)}$ implies that $M \models \psi$ because $\psi \in E_{\tau(S)}$, which in turn implies that $M_V \models \psi$ because $\psi \in PROP_V$.

Thus we proved that $KB \equiv_V \tau(KB)$ (which is equivalent to the last point). The first two points are corollaries of this V-equivalence with together the (obvious) fact that $S \cup W$ is consistent if and only if $\tau(S) \cup \tau(W)$ is consistent. ■

PROOF OF PROPOSITION 14. 1. Assume that $\tau(W) = \psi_1 \vee \cdots \vee \psi_m$ is a DNF formula. Note that since $\tau(\Delta) = \{ok(1), \ldots, ok(n)\}$, a maximal scenario S of $\tau(KB)$ is necessarily of the form $\{ok(i) \mid i \in X\}$ for some $X \subseteq 1..n$.

Let O be a subset of $\{ok(1), \ldots, ok(n)\}$. From the expression of $\tau(W)$, $O \wedge \tau(W)$ is consistent if and only if there is a $j \in 1..m$ such that $S \wedge \psi_j$ is consistent.

Now, we have the following equivalences:

$$
\begin{aligned}
&O \in \mathit{MaxScen}(KB)\\
&\quad\Leftrightarrow\quad O = \{\{ok(i) \mid i \in X\} \text{ such that for every } i \in X \text{ there is a } j\\
&\qquad\quad \text{such that } \psi_j \wedge ok(i) \text{ is consistent and no proper subset}\\
&\qquad\quad \text{of } O \text{ satisfies this property}\}\\
&\quad\Leftrightarrow\quad O \in \{\{ok(i) \in \tau(\Delta) \mid \psi_j \wedge ok(i) \text{ consistent}\} \mid j \in 1..m\}_{\min\supset}
\end{aligned}
$$

2. Assume that $\tau(W) = \psi_1 \wedge \cdots \wedge \psi_p$ is given by the conjunction of its prime implicates. O is a nogood of $\tau(KB)$ implies that $O \wedge \tau(W)$ is inconsistent, thus that $\tau(W) \models \neg O$. Thus, $\neg O$ is an implicate of $\tau(W)$, and if it were not a prime implicate of it then O would not be minimally inconsistent with $\tau(KB)$ and thus would not be a nogood. Therefore, $\neg O$ is a prime implicate of $\tau(W)$. Reciprocally, if $\neg O$ is a prime implicate of $\tau(W)$ then $\tau(W) \models \neg O$, i.e., $\tau(W) \wedge \neg O$ is inconsistent. The fact that $\neg O$ is a prime implicate implies that $\neg O$ is minimally inconsistent with $\tau(W)$, hence a nogood of $\tau(KB)$. ■

PROOF OF PROPOSITION 15. 1. We have $\tau(KB * ok(i)) = \langle \tau(W \cup \{\phi_i\}), \tau(\Delta) \setminus \{ok(i)\}\rangle$. (1) Clearly, this default theory has exactly the same extensions as $\langle \tau(W \cup \{\phi_i\}), \tau(\Delta)\rangle$, because $ok(i)$, being a consequence of $\tau(W \cup \{\phi_i\})$, is in all extensions, independently of the fact that it is in the set of defaults or not. (2) It also has the same extensions as $\langle \tau(W) \cup \{ok(i)\}, \tau(\Delta)\rangle$, because $\tau(W) \cup \{ok(i)\}$ and $\tau(W \cup \{\phi_i\})$ are logically equivalent. (3) Now, from Proposition 13, we have that $KB * ok(i)$ is V-equivalent to $\tau(KB * ok(i))$, and, since V-equivalence holds trivially between two default theories whose extensions are identical, it is V-equivalent to $\langle \tau(W) \cup \{ok(i)\}, \tau(\Delta)\rangle$.

2. The proof is similar, except for point 1.: $\tau(KB * bad(i)) = \langle \tau(W \cup \{neg\phi_i\}), \tau(\Delta) \setminus \{ok(i)\}\rangle\rangle$ has the same extensions as $\langle \tau(W \cup \{\neg\phi_i\}), \tau(\Delta)\rangle$, because $ok(i)$, being inconsistent with $\tau(W \cup \{\neg\phi_i\})$, is in no extension, independently of the fact that it is in the set of defaults or not.

3. The extensions of $\langle \tau(W) \cup \{\mathit{forget}(i)\}, \tau(\Delta)\rangle$ are identical to the extensions of $\langle \tau(W) \cup \{\mathit{forget}(i)\}, \tau(\Delta) \setminus \{ok(i)\}\rangle$ because $\tau(W) \cup \{\mathit{forget}(i)\} \models \neg ok(i)$. The V-equivalence between $KB * \mathit{forget}(i)$ and $\langle \tau(W) \cup \{\mathit{forget}(i)\}, \tau(\Delta)\}\rangle$ follows from the latter fact and Proposition 13. ■

PROOF OF PROPOSITION 16. 1. VALIDABILITY. Membership is shown by the following nondeterministic algorithm for VALIDABILITY: *guess an outcome*

R; check that $W \wedge R$ *is consistent; check that* $W \wedge R \models \phi_i$. This algorithm works in nondeterministic polynomial time using an NP-oracles, hence the membership to Σ_2^p.

Hardness is shown by the following polynomial reduction from 2-QBF to VALIDABILITY: to an instance $\exists a_1 \ldots \exists a_m \forall b_1 \ldots \forall b_n F$ of 2-QBF we associate the instance of VALIDABILITY defined by $T = \{\{a_i, \neg a_i\} \mid i = 1 \ldots m\}$, $W = \top$, $\phi_i = F$. It is easy to verify that $\exists a_1 \ldots \exists a_m \forall b_1 \ldots \forall b_n F$ is a positive instance of 2-QBF if and only if T potentially validates ϕ_i.

2. DISCRIMINABILITY. Membership of the complementary problem UNDISCRIMINABILITY) to NP is shown by the following nondeterministic algorithm: *guess an outcome* R *of* T *and two interpretations* M_1 *and* M_2*; check that* $M_1 \models W \wedge R \wedge \phi_i$ *and that* $M_2 \models W \wedge R \wedge \neg\phi_i$.

Hardness is proved by the following polynomial reduction from UNSAT to DISCRIMINABILITY: to any instance F of UNSAT we associate the instance of DISCRIMINABILITY defined by $T = \emptyset$, $W = \{F \vee l\}$ where l is a new literal, not appearing in F, and $\phi = l$. F is unsatisfiable if and only if T discriminates φ. ■

References

[1] ANTONIOU, G., 'On the dynamics of default reasoning', *Proc. of ECSQARU'99*, 1–10.

[2] BREWKA, G. 'Preferred subtheories: an extended logical framework for default reasoning', *Proc. of IJCAI'89*, 1043–1048.

[3] BENFERHAT, S., C. CAYROL, D. DUBOIS, J. LANG and H. PRADE, 'Inconsistency management and prioritized syntax-based entailment', *Proc. of IJCAI'93*, 640–645.

[4] DE KLEER, J., and B. WILLIAMS, 'Diagnosing multiple faults', *Artificial Intelligence* 32 (1987), 97–130.

[5] DE KLEER, J., A. MACKWORTH and R. REITER, 'Characterizing diagnoses and systems', *Artificial Intelligence* 56 (1992), 197–222.

[6] EITER, Th., and G. GOTTLOB, 'The complexity of logic-based abduction', *JACM* 42(1) (1995), 3–42.

[7] GOTTLOB, G., 'Complexity results for nonmonotonic logics', *Journal of Logic and Computation* 2 (1992), 397–425.

[8] KARP, R., 'Complexity of computer computations', in *Reducibility among combinatorial problems*, Plenum Press, New York, pp. 85–103, 1977.

[9] KEAN, A.,' A formal characterization of a domain independent abductive reasoning system', Ph.D. thesis, University of British Columbia, 1992.

[10] LANG, L., and P. MARQUIS, 'Complexity results for independence and definability in propositional logic', *Proc. of KR'98*, 356–367.

[11] Marquis, P., 'Consequence finding algorithms', in vol. 5, *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Kluwer Academic, 2000, 41–145.

[12] McIlraith, S., 'Generating tests using abduction', *Proc. of KR'94*, 449–460.

[13] McIlraith, S., and R. Reiter, 'On tests for hypothetical reasoning', *Readings in model-based diagnosis*, Morgan Kaufman, 89–96, 1992.

[14] Moore, R. C., 'A formal theory of knowledge and action', *Readings in Planning*, Morgan Kaufmann, 480–519, 1990.

[15] Nebel, B., 'Belief revision and default reasoning: syntax-based approaches', *Proc. of KR'91*, 417–428.

[16] Papadimitriou, Ch., *Computational Complexity*, Addison-Wesley, 1994.

[17] Poole, D., 'A logical framework for default reasoning', *Artificial Intelligence* 36 (1987), 27–47.

[18] Poole, D., 'Normality and faults in logic-based diagnosis', *Readings in model-based diagnosis*, Morgan Kaufman, pp. 71–77, 1992.

[19] Reiter, R., 'A theory of diagnosis from first principles', *Artificial Intelligence* 32(1) (1987), 57–96.

[20] Rymon, R., 'Search through systematic set enumeration', *Proc. of KR'92*, 539–550.

[21] Scherl R. B., and H. J. Levesque, 'The frame problem and knowledge-producing actions', *Proc. of AAAI'93*, 689–695.

[22] Stillman, J., 'The complexity of propositional default logics', *Proc. of AAAI'92*, 794–799.

[23] Stockmeyer, L. J., 'The polynomial hierarchy', *TCS* 3 (1977), 1–22.

[24] van Linder, B., W. van der Hoek and J.-J. Meyer, 'Tests as epistemic updates', *Proc. of ECAI'94*, 331–335.

[25] Williams M. A., and G. Antoniou, 'A strategy for revising default theory extensions', *Proc. of KR'98*, 24–33.

Jérôme Lang
IRIT-UPS
31062 Toulouse Cedex, France
lang@irit.fr

Pierre Marquis
CRIL/Université d'Artois
62307 Lens Cedex, France
marquis@cril.univ-artois.fr