Complexity results for independence and definability in propositional logic

Jérôme Lang
IRIT-UPS
118 route de Narbonne, 31062 Toulouse Cedex
France
e-mail: lang@irit.fr

Pierre Marquis
CRIL/Université d'Artois
rue de l'Université, S.P. 16, 62307 Lens Cedex
France
e-mail: marquis@cril.univ-artois.fr

Abstract

This paper focuses on three notions of independence/dependence that relate formulas and variables inside propositional logic: conditional (variable-variable) independence, formula-variable independence and a form of dependence between sets of variables w.r.t. a formula called definability. For each notion, definitions, various formal characterizations, complexity results and some tractable restrictions are exhibited. Close connections to some notions of relevance recently introduced in the literature are pointed out as well.

1 Introduction

When reasoning about knowledge represented in propositional logic, it is often useful to identify some particular relationships between several parts of the knowledge base, or between some of the variables. The first kind of relationship one may have in mind is *independence*, which is particularly useful when reasoning about belief change [11] [9], about action [19] or causality. At the opposite side, one may be interested in identifying dependence relations between variables or formulas.

This paper aims at studying several forms of independence/dependence that are *built-in inside propositional logic* (this contrasts with many works where independence is extra-logical); we especially consider these forms of independence from the point of view of computational complexity.

After some formal preliminaries (Section 2), we start with a simple form of formula-variable independence (Section 3). A knowledge base Σ is viewed independent of a set X of variables whenever giving truth values to variables of X cannot change the contents of Σ. Then we focus on conditional independence (Section 4). Conditional independence relates sets of variables w.r.t. a knowledge base and possible contexts (represented as assignments of variables). Intuitively, two variables x and y are considered independent iff for every possible context ω_Z, knowing the truth value of x does not enable deriving the truth value of y given Σ and *vice-versa*. We first consider conditional independence as introduced in [5] [8], and we extend this notion to two stronger forms of independence by enabling incomplete knowledge about contexts to be handled. We show how these different notions of independence are interrelated. Some characterization results are provided, the complexity of independence checking is analyzed and some tractable restrictions are exhibited. The computational complexity of closely related forms of (in)dependence, especially influenceability [4], relevance [14] [15], and causal independence [8] is also investigated.

Then we turn to the opposite side and focus on a form of dependence, called definability (Section 5). Intuitively, given a knowledge base Σ, a variable y can be defined from a set X of variables iff the knowledge of the truth values of the variables of X (whatever they are) enables us to conclude about the truth value of y. Several characterizations of defining and minimal defining families are provided; we also show how definability interacts with independence. Additionally, the complexity of basis checking is studied in the general case and some restricted cases where it is polynomial are pointed out.

Finally, some applications of independence / dependence are briefly sketched (Section 6), before concluding (Section 7).

2 Formal Preliminaries

Let PS be a finite 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 $X \subseteq PS$, $PROP_X$ denotes the sublanguage of $PROP_{PS}$ generated from the variables of X, only. A literal of $PROP_X$ is either a variable of X or the negation of a variable of X. A clause δ (resp. a term γ) of $PROP_X$ is a (possibly empty) disjunction (resp. conjunction) of literals of $PROP_X$. A CNF (resp. a DNF) formula of $PROP_X$ is a conjunction of clauses (resp. a disjunction of terms) of $PROP_X$.

From now on, Σ denotes a finite propositional knowledge base, i.e. a conjunctively interpreted finite set of propositional formulas from $PROP_{PS}$. $Var(\Sigma)$ is the set of propositional variables appearing in Σ. Elements of PS are denoted x, y etc. Subsets of PS are denoted X, Y etc. In order to simplify notations, we will assimilate every singleton $X = \{x\}$ with its unique element x. Full instantiations of variables of $X \subseteq PS$ are denoted by ω_X and their set is denoted Ω_X.

For every formula Φ and every variable x, $\Phi_{x\leftarrow 0}$ (resp. $\Phi_{x\leftarrow 1}$) is the formula obtained by replacing in Φ every occurence of x by the constant $false$ (resp. $true$). Two formulas Ψ and Φ are said *equivalent modulo* a formula Σ iff $\Sigma \wedge \Psi \equiv \Sigma \wedge \Phi$.

The set of prime implicants modulo Σ of a formula Φ over $PROP_X$, which will be denoted by $PI_\Sigma^X(\Phi)$, is defined by $PI_\Sigma^X(\Phi) = max(\{PI(\Sigma \Rightarrow \Phi) \cap PROP_X\}, \models)$ where $PI(\Phi)$ $(= PI_{true}^{PS}(\Phi))$ denotes the set of prime implicants of Φ.
Dually, $IP_\Sigma^X(\Phi)$ denotes the set of prime implicates of Φ modulo Σ over $PROP_X$, i.e.,
$IP_\Sigma^X(\Phi) = min(\{IP(\Sigma \Rightarrow \Phi) \cap PROP_X\}, \models)$
where $IP(\Phi)$ $(= IP_{true}^{PS}(\Phi))$ denotes the set of prime implicates of Φ. In both $PI_\Sigma^X(\Phi)$ and $IP_\Sigma^X(\Phi)$, only one representative per equivalence class is kept.

For instance, if $\Sigma = \{a \vee b, \neg a \wedge c \Rightarrow e, d \Leftrightarrow e\}$, then $PI_\Sigma^{\{a,b,c,d\}}(e) = \{d, \neg a \wedge c, \neg a \wedge \neg b\}$, $PI_\Sigma^{\{a,c\}}(e) = \{\neg a \wedge c\}$, $PI_\Sigma^{\{b,c\}}(e) = \emptyset$, $PI_\Sigma^{\{a,b,c,d\}}(a \vee b) = \{true\}$, $IP(\Sigma) = \{a \vee b, a \vee \neg c \vee e, \neg d \vee e, d \vee \neg e, a \vee \neg c \vee d\}$.

The complexity results we give in this paper refer to some complexity classes which deserve some recalls. More about them can be found in Papadimitriou's textbook [24]. We assume that the classes P, NP and coNP are known to the reader. The following classes will also be considered:
(i) BH_2 (also known as DP) is the class of all languages L such that $L = L_1 \cap L_2$, where L_1 is in NP and L_2 in coNP. The canonical BH_2-complete problem is SAT−UNSAT: a pair of formulas $\langle \varphi, \psi \rangle$ is in SAT−UNSAT iff φ is satisfiable and ψ is not.
(ii) $\Sigma_2^P = \mathsf{NP}^{\mathsf{NP}}$ is the class of all languages recognizable in polynomial time by a nondeterministic Turing machine using NP-oracles. Lastly, $\Pi_2^P = \mathsf{co}\Sigma_2^P$.

3 Formula-variable independence

In this section, we investigate several forms of formula-variable (in)dependence (FV (in)dependence for short), starting with the simplest one.

Let Σ be a KB and X a finite subset of PS. The most trivial form of formula-variable independence one can have in mind is the following syntactical one.

Definition 1 (syntactical FV independence) *Σ is syntactically independent of X iff $Var(\Sigma) \cap X = \emptyset$. When Σ is not syntactically independent of X, it is said* syntactically dependent *of X.*

This basic form of independence suffers from two important drawbacks. On the one hand, it does not satisfy the principle of irrelevance of syntax: two equivalent formulas are not independent of the same variables. On the other hand, it is too much demanding: while it is the case that no variable Σ is independent of which has any influence on Σ, the converse does not hold in the general case. For instance, $\Sigma = \{a, b \Leftrightarrow b\}$ is syntactically dependent of b, but any information about b does not convey anything new about Σ. This calls for a weaker but more robust notion of FV independence.

Definition 2 ((semantical) FV independence) *Σ is* independent *of X, noted $X \not\mapsto \Sigma$, iff there exists a formula Σ' s.t. $\Sigma' \equiv \Sigma$ and Σ is syntactically independent of X. When Σ is not independent of X, it is said* dependent *of X.*

Thus, Σ is independent of X iff Σ can be made syntactically independent of X while preserving logical equivalence. Accordingly, syntactical independence implies independence, but the converse does not hold in the general case. Thus, $\Sigma = \{a, b \Leftrightarrow b\}$ is independent of b.

Clearly enough, FV independence exhibits the following metatheoretic properties:

Proposition 1
(1) If $\Sigma \equiv \Sigma'$, then ($X \not\mapsto \Sigma$ iff $X \not\mapsto \Sigma'$).
(2) If $X \not\mapsto \Sigma$ and $X' \subseteq X$, then $X' \not\mapsto \Sigma$.

(3) If $X \not\mapsto \Sigma$ and $X \not\mapsto \Sigma'$, then $X \not\mapsto \neg\Sigma$, $X \not\mapsto \Sigma \wedge \Sigma'$ and $X \not\mapsto \Sigma \vee \Sigma'$.

Nevertheless, FV independence neither is monotonic nor anti-monotonic w.r.t. expansion of Σ (strengthening or weakening Σ can easily make it no longer independent of X). It is also not monotonic w.r.t. expansion of X. Thus, while $\Sigma = \{a\}$ is independent of b, neither $\{a, b\}$ nor $\{a \vee b\}$ are independent of b. Moreover, Σ is not independent of $\{a, b\}$.

Finally:

Proposition 2 *$X \not\mapsto \Sigma$ iff $\forall x \in X$, $x \not\mapsto \Sigma$.*

Dually, when Σ is dependent of X, there exists an $x \in X$ Σ is dependent of which. The case where Σ depends on every variable of X calls for a stronger notion of FV dependence, called full FV dependence.

Definition 3 (full FV dependence) *Σ is* fully dependent *of X iff Σ is dependent of every variable $x \in X$.*

A particular case of full FV dependence that proves useful is reached when $X = Var(\Sigma)$. In this situation, Σ is said simplified.

Definition 4 (simplified KB) *Σ is* simplified *iff Σ is fully dependent of $Var(\Sigma)$.*

For instance, $\Sigma = \{a, b \Leftrightarrow b\}$ is not simplified while the equivalent KB $\Sigma' = \{a\}$ is simplified.

Simplified KBs do not incorporate any useless variable. Accordingly, simplifying a KB can prove helpful for various reasoning tasks since it enables improving its readibility while decreasing its size. As the next proposition shows it, the notion of simplified KB actually is the point where syntactical independence and (semantical) independence coincide.

Proposition 3 *Σ is simplified iff the following equivalence holds: for every $X \subseteq PS$, Σ is syntactically independent of X iff $X \not\mapsto \Sigma$ holds.*

Interestingly, every KB can be simplified in a truth-preserving way:

Proposition 4 *For every Σ, there exists a simplified KB Σ' s.t. $\Sigma \equiv \Sigma'$.*

The following result gives a constructive proof of the previous proposition. It tells us that Σ remains unchanged when a variable Σ is independent of which is assigned a truth value:

Proposition 5 *The next four statements are equivalent:*
(1) $x \not\mapsto \Sigma$;
(2) $\Sigma_{x \leftarrow 0} \equiv \Sigma_{x \leftarrow 1}$;
(3) $\Sigma \equiv \Sigma_{x \leftarrow 0}$;
(4) $\Sigma \equiv \Sigma_{x \leftarrow 1}$.

On the previous example, we have $\Sigma_{b \leftarrow 0} \equiv \Sigma_{b \leftarrow 1} \equiv a$. Accordingly, Σ is independent of b.

Thanks to the proposition above, we can easily design a greedy algorithm for simplifying KBs. This algorithm consists in considering every variable x of $Var(\Sigma)$ in a successive way, while replacing Σ by $\Sigma_{x \leftarrow 0}$ whenever Σ is independent of x. At the end, the resulting KB is simplified.

We now give several characterizations of FV independence, allowing us to relate it with some notions of independence pointed out so far. 2. and 3. of Proposition 6 below tell us that FV independence can be easily read off from Σ whenever Σ is given under the prime implicants or the prime implicates normal form. For KBs given under these normal forms, independence comes down to syntactical independence.

Proposition 6 *The next four statements are equivalent:*
(1) $X \not\mapsto \Sigma$;
(2) $PI(\Sigma) \subseteq PROP_{Var(\Sigma) \setminus X}$;
(3) $IP(\Sigma) \subseteq PROP_{Var(\Sigma) \setminus X}$;
(4) $\Sigma \equiv forget(\Sigma, X)$.

Stepping back to our running example, $PI(\Sigma) = IP(\Sigma) = \{a\}$ clearly is syntactically independent of variable b.

Interestingly, this proposition shows our notion of FV independence to coincide with Boutilier's notion of non-influenceability; indeed, Proposition 4 from [4] states that Σ is influenceable iff there exists a prime implicant of Σ which contains a variable from X, where X is the set of so-called controllable variables. It also shows FV independence to coincide with Lakemeyer's notion of irrelevance of a formula to a subject matter since Theorem 3 from [14] states that Σ is relevant to X iff there exists a prime implicate of Σ which contains a variable from X. Accordingly, Proposition 6 gives rise to a new model-theoretic characterization of influenceability and relevance. The last characterization result 4. of Proposition 6 relates FV independence to the $forget$ operator introduced in the more general framework of first-order logic by Lin and Reiter [17]. $forget(\Sigma, X)$ is defined inductively by:
$forget(\Sigma, \emptyset) = \Sigma$;

$forget(\Sigma, \{x\}) = (\Sigma_{x\leftarrow 0} \vee \Sigma_{x\leftarrow 1})$;
$forget(\Sigma, X \cup \{x\}) = forget(forget(\Sigma, X), \{x\})$.

$forget(\Sigma, X)$ is the most general consequence of Σ that is independent of X, up to logical equivalence. On our example, $forget(\{a, b \Leftrightarrow b\}, \{b\}) \equiv a$.

Point 3. of Proposition 6 also allows us to give an alternative characterization of the notion of *strict relevance* introduced by Lakemeyer in [15]. Indeed, a formula Σ is said strictly relevant[1] to a subject matter X iff there exists a prime implicate of Σ mentioning a variable from X, and every prime implicate of Σ mentions only atoms from X. Accordingly, 3. shows Σ strictly relevant to X iff Σ is dependent of X and independent of $Var(\Sigma) \setminus X$.

We turn now into computational complexity issues. While syntactical FV dependence can be easily checked in linear time in the size of the input, this is far from being expected for FV dependence and full FV dependence in the general case:

Proposition 7 (complexity of FV dependence) FV DEPENDENCE *and* FULL FV DEPENDENCE *are* NP*-complete.*

Using then Proposition 6, we get the following corollary:

Proposition 8
(1) Boutilier's influenceability is NP*-complete.*
(2) Lakemeyer's relevance of a formula to a subject matter is NP*-complete.*

We have also derived the following:

Proposition 9
(1) Lakemeyer's [15] strict relevance of a formula to a subject matter is BH_2*-complete.*
(2) Lakemeyer's [14] strict relevance of a formula to a subject matter is Π_2^P*-complete.*

These complexity results improve Theorem 50 from [15] which only points out the NP-hardness of irrelevance and strict irrelevance (w.r.t. the definition given in [15]).

Interestingly, the complexity of both decision problems fall down to P whenever checking FV (in)dependence becomes tractable. Apart from the case of syntactical FV independence, some other restrictions on Σ makes FV (in)dependence testable in polynomial time. Especially, as a corollary of Proposition 5 we get:

Proposition 10 *Whenever Σ belongs to a class $\mathcal{C}$ of formulas that is tractable for clausal query answering and stable for variable instantiation (i.e. replacing in $\Sigma \in \mathcal{C}$ any variable by* true *or by* false *gives a formula that still belongs to $\mathcal{C}$) then* FV (IN)DEPENDENCE *is in* P.

In particular, when Σ is restricted to a renamable Horn CNF formula or to binary clauses, FV (IN)DEPENDENCE belongs to P. Theorem 51 from [15] can be viewed as a restriction of the latter to the case where Σ is Horn.

Additionally, when FV INDEPENDENCE is in P, the simplification algorithm given above runs in time polynomial in the size of Σ, i.e. the corresponding function problem is in FP: the simplification task can be achieved in a tractable way.

4 Conditional independence

In contrast to FV independence, conditional independence relates several sets of variables within a propositional KB. Conditional independence has been first introduced by Darwiche and Pearl [8] [6] as a logical counterpart of probabilistic independence.

Definition 5 (conditional independence) *[8] Let Σ be a KB and X, Y, Z disjoint subsets of PS. X and Y are independent given Z w.r.t. Σ (denoted by $X \sim_\Sigma^Z Y$) iff $\forall \omega_X \in \Omega_X, \forall \omega_Y \in \Omega_Y, \forall \omega_Z \in \Omega_Z$, the consistency of both $\omega_X \wedge \omega_Z \wedge \Sigma$ and $\omega_Y \wedge \omega_Z \wedge \Sigma$ implies the consistency of $\omega_X \wedge \omega_Y \wedge \omega_Z \wedge \Sigma$.*

Clearly enough, conditional independence given Z w.r.t. Σ satisfies the following properties [7]:

Proposition 11
(1) $X \sim_\Sigma^Z Y$ iff $Y \sim_\Sigma^Z X$.
(2) If $\Sigma \equiv \Sigma'$, then ($X \sim_\Sigma^Z Y$ iff $X \sim_{\Sigma'}^Z Y$).
(3) If $X' \subseteq X$ and $Y' \subseteq Y$ and $X \sim_\Sigma^Z Y$, then $X' \sim_\Sigma^Z Y'$.

However, conditional independence neither is not stable by contraction or by expansion of Z. For instance, let $\Sigma = \{\neg a \vee \neg b \vee c, \neg a \vee b \vee d, a \vee \neg c, \neg a \vee c \vee y, b \vee \neg c \vee d\}$. We have $c \sim_\Sigma^\emptyset d$ but $c \not\sim_\Sigma^{\{a\}} d$ (because of $\neg a \vee c \vee d$: when a is *true*, learning $\neg c$ tells that d is *true*), $c \not\sim_\Sigma^{\{b\}} d$ and however $c \sim_\Sigma^{\{a,b\}} d$ (because $a \wedge b \wedge \Sigma \models c$, $a \wedge \neg b \wedge \Sigma \models d$ and $\neg a \wedge \Sigma \models \neg c$, i.e., full knowledge about a and b breaks all links between c and d). Conditional independence is also not stable by weakening

[1] As pointed out in [15], this notion differs from the one given in [14], where Σ is said strictly relevant to X iff every prime implicate of Σ contains a variable from X.

or strengthening Σ in the general case. Thus, while we have $c \sim^{\emptyset}_{\Sigma} d$, we also have $c \not\sim^{\emptyset}_{\Sigma \cup \{c \Leftrightarrow d\}} d$; while we have $c \sim^{\{a,b\}}_{\Sigma} d$, we also have $c \not\sim^{\{a,b\}}_{\Sigma \setminus \{\neg a \vee \neg b \vee c\}} d$.

The two limit cases when Z is respectively empty or equal to $Var(\Sigma) \setminus (X \cup Y)$, are of particular interest, especially when investigating computational complexity.

Definition 6 (marginal independence) *X and Y are* marginally independent *w.r.t. Σ iff $X \sim^{\emptyset}_{\Sigma} Y$.*

Definition 7 (ceteris paribus independence) *X and Y are* ceteris paribus independent *w.r.t. Σ (denoted by $X \sim^{ceteris\ paribus}_{\Sigma} Y$) iff $X \sim^{Var(\Sigma) \setminus (X \cup Y)}_{\Sigma} Y$.*

As said in [8], $X \sim^{Z}_{\Sigma} Y$ holds iff for any possible *full information* about Z, the addition of information about Y does not tell us anything new about X. This definition does not apply to contexts where the new information that can be learned about Z may be incomplete., i.e. the truth value of some variables of Z is not available, or, more generally, many partial (and eventually mutually exclusive) assignments of variables of Z are possible. For instance, if Z represents a set of possibly measurable variables, associated to a set of sensors (one for each $z \in Z$), it can be the case that some measurements fail, i.e. the value of z is not always available.

We are now going to strengthen Darwiche and Pearl's conditional independence by taking account of incomplete information about Z. We propose two definitions, whether the incomplete information about Z is required to be conjunctive or not. Namely, X and Y are strongly (resp. perfectly) independent given Z w.r.t. Σ iff *whichever conjunctive information (i.e. a set of facts)* (resp. *whichever information*, i.e. any formula) we may learn about Z, then the addition of information about Y does not enable telling anything new about X.

Definition 8 (strong conditional independence) *X and Y are* strongly independent *given Z w.r.t. Σ (denoted $X \approx^{Z}_{\Sigma} Y$) iff for any term γ_Z of $PROP_Z$, the consistency of both $\omega_X \wedge \gamma_Z \wedge \Sigma$ and of $\omega_Y \wedge \gamma_Z \wedge \Sigma$ implies the consistency of $\omega_X \wedge \omega_Y \wedge \gamma_Z \wedge \Sigma$.*

Strong conditional independence has the same metatheoretic properties as conditional independence, plus the preservation by contraction of Z (which is a trivial consequence of the definition).

Marginal strong conditional independence obviously coincides with marginal conditional independence. *Ceteris paribus* strong conditional independence is defined by imposing $Z = Var(\Sigma) \setminus (X \cup Y)$ and denoted by $X \approx^{ceteris\ paribus}_{\Sigma} Y$.

Since the set of all possible choices for γ_Z corresponds to the set of all partial assignments of the variables of Z, we get:

Proposition 12
$X \approx^{Z}_{\Sigma} Y$ iff $X \sim^{Z'}_{\Sigma} Y$ for every $Z' \subseteq Z$.

Obviously, $X \approx^{Z}_{\Sigma} Y$ entails $X \sim^{Z}_{\Sigma} Y$. The converse generally does not hold since conditional independence is not stable by contraction of Z. Indeed, for the previous example, we have $c \sim^{\{a,b\}}_{\Sigma} d$ but $c \not\approx^{\{a,b\}}_{\Sigma} d$ since $c \sim^{\{b\}}_{\Sigma} d$ does not hold.

As a corollary of Proposition 12, *ceteris paribus* strong independence coincides with Lakemeyer's definition [14] of irrelevance between two subject matters relative to a knowledge base[2].

The following results enable characterizing strongly conditionally independent sets of variables. They both express that $X \approx^{Z}_{\Sigma} Y$ iff any set of simple facts (i.e. literals) we may learn about Z never enables deducing a nontrivial *disjunctive information* involving both X and Y.

Proposition 13 (consequence decomposability) *$X \approx^{Z}_{\Sigma} Y$ iff for any term γ_Z of $PROP_Z$, and $\forall \varphi_X \in PROP_X$, $\forall \varphi_Y \in PROP_Y$, $\gamma_Z \wedge \Sigma \models \varphi_X \vee \varphi_Y$ implies $\gamma_Z \wedge \Sigma \models \varphi_X$ or $\gamma_Z \wedge \Sigma \models \varphi_Y$.*

Proposition 14 *$X \approx^{Z}_{\Sigma} Y$ iff $\forall \delta \in IP^{X \cup Y \cup Z}_{true}(\Sigma)$, δ never includes both a variable of X and a variable of Y.*

Proposition 14 slightly generalizes Lakemeyer's Theorem 5 from [14] (we recover it when $Z = V \setminus (X \cup Y)$). Note that there is no such nice characterization in terms of prime implicates for (standard) conditional independence.

As a consequence of Proposition 14, it is enough in practice to consider strong conditional independence of single variables:

Proposition 15
$X \approx^{Z}_{\Sigma} Y$ iff $\forall x \in X$ $\forall y \in Y$, $x \approx^{Z \cup (X \setminus \{x\}) \cup (Y \setminus \{y\})}_{\Sigma} y$.

This result is useful for the practical computation of strong conditional independence relations. Note that

[2] We omit Lakemeyer's original definition of relevance for the sake of brevity; it is enough to refer to Theorem 31 in [15] which states that X is relevant to Y w.r.t. Σ iff there is a Z such as $X \not\sim^{Z}_{\Sigma} Y$.

there is no similar result for (standard) conditional independence. Things become even simpler with *ceteris paribus* strong independence, since Proposition 15 becomes: $X \approx_{\Sigma}^{ceteris\ paribus} Y$ iff $\forall x \in X\ \forall y \in Y$, $x \approx_{\Sigma}^{ceteris\ paribus} y$ (cf. Lemma 15 in [15]).

Interestingly, FV independence can be recovered from strong conditional independence:

Proposition 16
Let new be a variable of $PS \setminus Var(\Sigma)$.
$X \not\leftrightarrow \Sigma$ *iff* $X \approx_{\Sigma \Leftrightarrow new}^{Var(\Sigma)\setminus X}$ *new* .

This means that in any state of knowledge regarding $Var(\Sigma) \setminus X$, knowing the truth values of variables in X cannot help us knowing the truth value of *new* and hence of Σ. The converse, i.e., expressing strong conditional independence from FV independence, is possible as well (see Proposition 17). However, the exhibited transformation is not polynomial and thus will not be helpful when investigating computational complexity.

According to Proposition 14, $x \not\approx_{\Sigma}^{Z} y$ iff there is a prime implicate δ in $IP_{true}^{Z\cup\{x,y\}}(\Sigma)$ mentioning both x and y. This is equivalent to say that there is a prime *implicant* γ in $PI_{\Sigma}^{Z\cup\{x\}}(y)$ or in $PI_{\Sigma}^{Z\cup\{x\}}(\neg y)$, *consistent with* Σ and mentioning x. The consistency condition is necessary; indeed, let us consider $\Sigma = \{c \Rightarrow a, d \Rightarrow b\}$ and $Z = \{c, d\}$; $PI_{\Sigma}^{Z\cup\{a\}}(b) = \{c \wedge \neg a, d\}$ mentions a but nevertheless $a \approx_{\Sigma}^{\{c,d\}} b$ holds; this is because $c \wedge \neg a$ is not consistent with Σ, or in other words, $c \wedge \neg a$ is a prime implicant of $\Sigma \Rightarrow b$ only because it is a prime implicant of $\neg\Sigma$. Thus, the set of prime implicants of interest is $PI_{\Sigma}^{Z\cup\{x\}}(y) \setminus PI^{Z\cup\{x\}}(\neg\Sigma)$, which corresponds exactly to set of minimal abductive explanations of y w.r.t. Σ, where the set of possible individual hypotheses is the set of literals built up from $Z \cup \{x\}$ [10]. Equivalently, this set is the *label* of y according to the ATMS literature [25]. This leads to the following characterization:

Proposition 17 *Let $Label_{\Sigma}^{Z\cup\{x\}}(y)$ be the disjunction of all prime implicants γ in $PI_{\Sigma}^{Z\cup\{x\}}(y)$ s.t. $\gamma \wedge \Sigma$ is consistent. Then $x \approx_{\Sigma}^{Z} y$ iff both $Label_{\Sigma}^{Z\cup\{x\}}(y)$ and $Label_{\Sigma}^{Z\cup\{x\}}(\neg y)$ are independent of x.*

In other words, $x \approx_{\Sigma}^{Z} y$ iff both x and $\neg x$ are irrelevant hypotheses for (minimally) explaining y and $\neg y$ (i.e. neither x nor $\neg x$ does participate to any minimal explanation of y and neither x nor $\neg x$ does participate to any minimal explanation of $\neg y$) [10].

This gives us an algorithm for computing strong independence relations using a basic ATMS. Let $SIV_{\Sigma}^{Z}(y) = \{x \in Var(\Sigma) \setminus \{y\} \mid x \approx_{\Sigma}^{Z} y\}$. A list of variables `L` is initialized to $Var(\Sigma) \setminus \{y\}$, and each time a new environment of y (i.e. one of the disjuncts of $Label_{\Sigma}^{Z\cup\{x\}}(y)$) or of $\neg y$ is computed, then all variables appearing in it are removed from `L`. Then, at any step, `L` contains $SIV_{\Sigma}^{Z}(y)$ and the algorithm reaches $SIV_{\Sigma}^{Z}(y)$ when it ends up (this shows a possible "anytime" use of this algorithm).

We now define our last notion of conditional independence, stronger than the two previous ones.

Definition 9 (perfect conditional independence) *X and Y are perfectly independent for Z w.r.t. Σ (denoted $X \asymp_{\Sigma}^{Z} Y$) iff $\forall \varphi_Z \in PROP_Z$, $\forall \omega_X \in \Omega_X$, $\forall \omega_Y \in \Omega_Y$, the consistency of both $\omega_X \wedge \Sigma \wedge \varphi_Z$ and of $\omega_Y \wedge \Sigma \wedge \varphi_Z$ implies the consistency of $\omega_X \wedge \omega_Y \wedge \Sigma \wedge \varphi_Z$.*

$X \asymp_{\Sigma}^{Z} Y$ means that no significant relationship between X and Y can be inferred when learning *any* information, including disjunctive information, about Z. This is expressed by the following result, similar to Proposition 13:

Proposition 18
$X \asymp_{\Sigma}^{Z} Y$ iff $\forall \varphi_Z \in PROP_Z$, $\forall \varphi_X \in PROP_X$, $\forall \varphi_Y \in PROP_Y$, $\varphi_Z \wedge \Sigma \models \varphi_X \vee \varphi_Y$ implies $\varphi_Z \wedge \Sigma \models \varphi_X$ or $\varphi_Z \wedge \Sigma \models \varphi_Y$.

Clearly, $X \asymp_{\Sigma}^{Z} Y$ implies $X \approx_{\Sigma}^{Z} Y$ (the latter is recovered when φ_Z is a term). The converse is generally false, as shown by the following example: $\Sigma = \{c \Rightarrow a, d \Rightarrow b\}$, $X = \{a\}$, $Y = \{b\}$, $Z = \{c, d\}$, $\varphi_X = a$, $\varphi_Y = b$, and $\varphi_Z = c \vee d$. We have $a \approx_{\Sigma}^{Z} b$; nevertheless, we have $\varphi_Z \wedge \Sigma \not\models a$, $\varphi_Z \wedge \Sigma \not\models b$ and $\varphi_Z \wedge \Sigma \models a \vee b$, which means that $a \not\asymp_{\Sigma}^{Z} b$.

In contrast to the two weaker forms of independence discussed before, perfect independence is insensitive to the granularity of the representation, in the following sense: if Z is replaced by another set of variables Z' s.t. Z and Z' define each other in Σ (see Section 5), then perfect independence is preserved:

Proposition 19 *If Z and Z' are subsets of PS s.t. every z of Z (resp. Z') is equivalent modulo Σ to a formula of $PROP_{Z'}$ (resp. $PROP_Z$), then ($X \asymp_{\Sigma}^{Z} Y$ iff $X \asymp_{\Sigma}^{Z'} Y$).*

As an illustration, let $Z = \{n(orth), s(outh), e(ast), w(est)\}$ and $Z' = \{ne, nw, se, sw\}$ where Σ

contains $s \Leftrightarrow (se \vee sw)$, $e \Leftrightarrow (ne \vee se)$, etc. and mutual exclusivity statements between ne,nw,se and sw (such as $sw \Rightarrow \neg se$, etc.). Z and Z' define each other, because Σ entails $ne \Leftrightarrow (n \wedge e)$, etc. Let us now add to Σ the two formulas $se \Rightarrow rain$ and $sw \Rightarrow wind$, which imply $s \Rightarrow (rain \vee wind)$. Then *rain* and *wind* are strongly independent given Z' while they are not given Z. In both cases, perfect independence between *rain* and *wind* does not hold, because we may later learn something about the theme "location" implying a link between *wind* and *rain*. This example shows that the lack of perfect independence between X and Y corresponds intuitively to a *potential dependence* given the theme corresponding to Z. Unfortunately, it seems that perfect conditional independence does not have any simple characterization in terms of prime implicants/implicates.

We end up this section with computational complexity issues. We start by investigating in depth the complexity of standard conditional independence. We distinguish a number of restrictions on X, Y, Z and Σ which may lower the complexity level, namely: $|X| = 1$ and/or $|Y| = 1$ (checking whether a variable is independent from a variable / a set of variables), $X \cup Y = Var(\Sigma)$ (*twofold partition independence*), $Z = \emptyset$ (*marginal independence*) and $Z = Var(\Sigma) \setminus (X \cup Y)$ (*ceteris paribus independence*)[3].

Proposition 20 (complexity of independence) *The results are synthesized on Table 1 (where V stands for $Var(\Sigma)$).*

The numerous results contained in Proposition 20 are proved in the following order, which tries to minimize the number of proofs: 1. CONDITIONAL INDEPENDENCE is in Π_2^{P}; 2. MARGINAL INDEPENDENCE OF A VARIABLE FROM A SET OF VARIABLES is Π_2^{P}-hard; 3. CONDITIONAL INDEPENDENCE OF SINGLE VARIABLES is Π_2^{P}-hard; 4. CETERIS PARIBUS INDEPENDENCE is in coNP; 5. TWOFOLD PARTITION INDEPENDENCE is coNP-hard; 6. MARGINAL VARIABLE INDEPENDENCE is coBH_2-complete.

Proposition 21 (strong independence) STRONG CONDITIONAL INDEPENDENCE *is Π_2^{P}-complete.*

Strong conditional independence remains Π_2^{P}-complete when X or Y is a singleton and when both are singletons. As to *ceteris paribus* strong independence, it is in Π_2^{P} but we do do not have a proof of completeness.

[3] Note that, for twofold partition independence, the distinctions on Z are irrelevant; therefore, all three problems of the last row of Table 1 are identical.

Note that the Π_2^{P}-completeness of STRONG CONDITIONAL INDEPENDENCE coheres with the Σ_2^{P}-completeness of checking whether an individual hypothesis is relevant (for minimal explanation), as shown in [10]. More interestingly, the abductive characterization[4] (Proposition 15) of strong conditional independence enables taking advantage of some restrictions (especially restricting Σ to a set of Horn clauses) for which the computational complexity of checking irrelevance for minimal explanation falls down to the first level of the polynomial hierarchy, carrying with it the complexity of strong conditional independence. Considering DNF KBs is another restriction that makes the complexity of STRONG CONDITIONAL INDEPENDENCE falling down to the first level of the polynomial hierarchy. To be more precise, when Σ is in DNF, we can easily prove that STRONG CONDITIONAL INDEPENDENCE becomes coNP-complete.

Things are not very different with perfect conditional independence.

Proposition 22 (perfect independence) PERFECT CONDITIONAL INDEPENDENCE *is Π_2^{P}-complete.*

Focusing on the aforementioned syntactic restrictions leads exactly to the same results as for strong conditional independence, and the same lack of proof of Π_2^{P}-completeness for the *ceteris paribus* case.

These results are significant for a number of related problems such as relevance or causality (with applications to model-based diagnosis [6]). In particular, we identify the complexity of causal independence in symbolic causal networks (Darwiche and Pearl [8]). We omit the full definition of causal independence because it would need too much space (see [8]). In short, a propositional knowledge base Σ is causally independent w.r.t. a causal structure $\mathcal{G}$ (being a directed acyclic graph on a subset of $Var(\Sigma)$ where edges represent direct causal relations) iff (i) Σ is satisfiable and (ii) for any state S of the variables not concerned by the causal structure, and for each variable v of $\mathcal{G}$, then v is independent of its non-effects given its direct causes w.r.t. $\Sigma \wedge S$, where the non-effects of v are the

[4] By the way, let us mention that [15] introduces a notion of relevance $RX_\Sigma(X, \Phi)$ of a formula Φ to a subject matter X w.r.t. a KB Σ that can be characterized as well using abductive terms. Indeed, $RX_\Sigma(X, \Phi)$ holds iff there exists a minimal abductive explanation of Φ w.r.t. Σ that contains some variable from X. Accordingly, $RX_\Sigma(X, \Phi)$ holds iff at least one literal built up from X is relevant for minimally explaining Φ. The Σ_2^{P}-completeness of this additional form of relevance then comes from [10]. This completes the complexity results given in [15].

$X \sim_{\Sigma}^{Z} Y$	any Z	$Z = \emptyset$	$Z = V \setminus (X \cup Y)$
any X,Y	Π_2^{P}-complete	Π_2^{P}-complete	coNP-complete
$X = \{x\}$ or $Y = \{y\}$	Π_2^{P}-complete	Π_2^{P}-complete	coNP-complete
$X = \{x\}$ and $Y = \{y\}$	Π_2^{P}-complete	coBH_2-complete	coNP-complete
$X \cup Y = V$	coNP-complete	coNP-complete	coNP-complete

Table 1: complexity of conditional independence.

vertices in $\mathcal{G}$ which are neither parents nor descendents of v.

Proposition 23
Darwiche and Pearl's CAUSAL INDEPENDENCE *is* Π_2^{P}-*complete.*

5 Definability

Let us now focus on a form of dependence between sets of variables w.r.t. a KB, called *definability*. Definability is a strong form of dependence: while (conditionally) dependent variables interact *in some situations* ω_Z, definability imposes that some variables are fixed *whenever* some other variables are fixed as well.

Definition 10 (defining families, bases)
Let $X, Y \subseteq PS$ *and* $y \in PS$.

- X **defines** y *w.r.t.* Σ, *denoted* $X \sqsubseteq_\Sigma y$, *iff* $\forall \omega_X \in \Omega_X, \omega_X \wedge \Sigma \models y$ *or* $\omega_X \wedge \Sigma \models \neg y$.
- X **defines** Y *w.r.t.* Σ *(denoted* $X \sqsubseteq_\Sigma Y$*) iff* $\forall y \in Y, X \sqsubseteq_\Sigma y$.
- X **defines minimally** Y *w.r.t.* Σ *iff* $X \sqsubseteq_\Sigma Y$ *and no proper subset of* X *does it.*
- X *is a* **basis** *for* Y *w.r.t.* Σ *iff* X *defines minimally* Y *w.r.t.* Σ *and* Σ *is consistent.*

For instance, let b stand for "bissextile year", and $d4$ (resp. $d25$, $d100$, $d400$) for "divisible by 4" (resp. by 25, 100, by 400). The knowledge base is
$\Sigma = \{d400 \Rightarrow b, d100 \wedge \neg d400 \Rightarrow \neg b, d4 \wedge \neg d100 \Rightarrow b, \neg d4 \Rightarrow \neg b, d100 \Leftrightarrow (d4 \wedge d25)\}$.
$\{d4, d100, d400\}$ and $\{d4, d25, d400\}$ are bases for b;
$\{d25, d100, d400\}$ does not define b, because the joint falsity of these three variables does not enable telling whether b is true or false, since we do not know whether $d4$ holds or not.
$\{b, d100, d400\}$ defines $d4$, but not minimally, since $\{b, d100\}$ is a basis for $d4$; the latter also is a basis for $\{d4, d100, d400\}$.

Clearly enough, definability satisfies the following properties:

Proposition 24
(1) $\sqsubseteq_\Sigma$ *is transitive.*
(2) If $X' \subseteq X$*, then* $X \sqsubseteq_\Sigma X'$.
(3) If $X \sqsubseteq_\Sigma Y$ *and* $X \sqsubseteq_\Sigma Y'$*, then* $X \sqsubseteq_\Sigma Y \cup Y'$.
(4) If $X \sqsubseteq_\Sigma Y$ *and* $\Sigma' \models \Sigma$*, then* $X \sqsubseteq_{\Sigma'} Y$.

Easy consequences of such properties are:
(5) $X \sqsubseteq_\Sigma X$.
(6) If $X \sqsubseteq_\Sigma Y$ and $X' \sqsubseteq_\Sigma Y'$, then $X \cup X' \sqsubseteq_\Sigma Y \cup Y'$.

As a corollary of Beth's theorem [3] (stated in the more general framework of first-order logic), we get the equivalence between this *implicit* form of definability and the following *explicit* form: X (explicitly) defines y w.r.t. Σ iff there exists a formula Φ_y s.t. $Var(\Phi_y) \subseteq X$ and $\Sigma \models (\Phi_y \Leftrightarrow y)$; in this case, Φ_y is clearly unique up to equivalence modulo Σ.

Going back to our example, we get the following explicit definitions:
$\Sigma \models (d4 \wedge (\neg d100 \vee d400)) \Leftrightarrow b$;
$\Sigma \models (d4 \wedge (\neg d25 \vee d400)) \Leftrightarrow b$;
$\Sigma \models (b \vee d100) \Leftrightarrow d4$;
$\Sigma \models (d100 \wedge b) \Leftrightarrow d400$;

The following equivalent formulation is due to Padoa[23]. It is useful for looking for tractable restrictions of definability.

Proposition 25 (Padoa's method) *[23]*
For any Σ *and any* $X \subseteq PS$*, let* $rename(\Sigma, X)$ *be the formula obtained by replacing in* Σ *in a uniform way every variable from* $Var(\Sigma) \setminus X$ *by a new variable. If* $y \notin X$*, then (*$X \sqsubseteq_\Sigma y$ *iff* $(\Sigma \wedge rename(\Sigma, X)) \models (y \Leftrightarrow rename(y, \emptyset))$*).*

Clearly enough, whenever y does not belong to X, checking definability comes down to a standard deduction check. Since $X \sqsubseteq_\Sigma y$ trivially holds in the remaining case (i.e. $y \in X$), we can conclude that a set membership test plus a deduction check are always sufficient to decide definability.

Interestingly, the previous notion of variable definability can be easily turned into a notion of formula definability. Formally:

Definition 11 X defines Ψ *w.r.t.* Σ *(noted* $X \sqsubseteq_\Sigma \Psi$*) iff* $\forall \omega_X \in \Omega_X, \omega_X \models_\Sigma \Psi$ *or* $\omega_X \models_\Sigma \neg\Psi$. $Defin(X, \Sigma) = \{\Psi \mid X$ *defines* Ψ *w.r.t.* $\Sigma\}$ *is the sublanguage implicitely defined by* X *w.r.t.* Σ.

$Defin(X, \Sigma)$ is the set of formulas Ψ the truth values of which are known whenever the truth values of variables of X are known. $Defin$ exhibits the following metatheoretic properties:

Proposition 26
(1) If $X \sqsubseteq_\Sigma Y$*, then* $Defin(Y, \Sigma) \subseteq Defin(X, \Sigma)$.
(2) If $\phi, \psi \in Defin(X, \Sigma)$*, then* $\phi \wedge \psi \in Defin(X, \Sigma)$, $\phi \vee \psi \in Defin(X, \Sigma)$, $\neg\phi \in Defin(X, \Sigma)$.

While formula definability extends variable definability (since every variable y can be also viewed as the formula y), it can be recovered from it easily:

Proposition 27 *Let* new *be a variable of* $PS \setminus Var(\Psi)$. $X \sqsubseteq_\Sigma \Psi$ *iff* $X \sqsubseteq_{\Sigma \wedge (\Psi \Leftrightarrow new)} new$.

This proposition shows how an algorithm for checking variable definability can be used to decide formula definability.

Definability can be used to discriminate variables within a knowledge base in many ways, both as defining as well as defined variables. The most important ones are the variables which are necessary for defining Y, i.e., those which belong to all bases for Y. Necessary variables will be called *key variables* by analogy with the terminology used in the database community. When designing minimal tests, key variables correspond to necessary tests, which will have to be performed anyway (and thus, when building up conditional test plans, these necessary tests can be performed at once and in parallel without needing for any outcome). When a key variable cannot be tested, the set of hypotheses (diagnoses, etc.) cannot be fully discriminated. Computing key variables in a preliminary step can also prove valuable for improving the computation of the set of all bases for Y w.r.t. Σ.

Definition 12
x *is a* key variable *for* Y *w.r.t.* Σ *if and only if* x *belongs to all bases of* Y *w.r.t.* Σ.

Going back to our example, given that $Var(\Sigma) = \{b, d4, d25, d100, d400\}$, $d4$ and $d400$ are the only key variables for b; $d100$ is not a key variable for b (because $\{d4, d25, d400\}$ defines b and does not contain $d100$). The key variables for $Var(\Sigma)$ are $d4$ and $d400$. Lastly, $d25$ is the unique key variable for $d25$, since $d25$ is the unique basis for $d25$ w.r.t. Σ. As this example illustrates it, all the variables y of Y that cannot be defined (except in a trivial way) in Σ are key variables for Y.

Definition 13 y *is undefinable in* Σ *iff for all* $X \subseteq Var(\Sigma)$*, if* $X \sqsubseteq_\Sigma y$*, then* $y \in X$.

Now, since $Y \sqsubseteq_\Sigma Y$ trivially holds, it is easy to show that:

Proposition 28 x *is a key variable for* Y *w.r.t.* Σ *iff* $x \in Y$ *and* x *is undefinable in* Σ.

Moreover, since $\sqsubseteq_\Sigma$ is a transitive relation, undefinability in Σ can be considered as a specific case of non-definability.

Proposition 29 y *is undefinable in* Σ *iff* $Var(\Sigma) \setminus \{y\} \not\sqsubseteq_\Sigma y$.

Undefinability in Σ also has the following characterizations that prove helpful for deriving algorithms for deciding it.

Proposition 30 *The next three statements are equivalent:*
(1) y *is undefinable in* Σ;
(2) $\Sigma_{y \leftarrow 0} \wedge \Sigma_{y \leftarrow 1}$ *is satisfiable.*
(3) There exists a prime implicant of Σ *that does not contain* y *nor* $\neg y$.

Another way to simplify the search for defining families for a given variable y is to take account of some prior knowledge concerning independence. Intuitively, a basis for y is expected not to contain variables independent of y (the notion of independence we need here is the *ceteris paribus* one).

Proposition 31 *If* $x \sim_\Sigma^{Var(\Sigma) \setminus \{x,y\}} y$ *and* X *is a basis for* y*, then* $y \notin X$.

The following remark is not especially helpful from a computational perspective but shows a close connection between the definitions of conditional independence and definability. Notice that, although the original definition of conditional independence between X and Y w.r.t. Z requires X and Y to be disjoint, we may wish to apply the definition in the case where X and Y intersect. For $X = Y$, it leads to $X \sim_\Sigma^Z X$ if and only if $Z \sqsubseteq_\Sigma X$, which proves that definability is covered by (unrestricted) conditional independence. More generally, $X \sim_\Sigma^Z Y$ iff $Z \sqsubseteq_\Sigma X \cap Y$ and $X \setminus Y \sim_\Sigma^Z Y \setminus X$.

We now give the complexity of definability in the general case, as well as in some restricted cases.

Proposition 32 (complexity of definability) *The results are synthesized on Table 2.*

Since the transformation from formula definability to variable definability given by Proposition 27 can be achieved in polynomial time and since variable definability is a restriction of formula definability, these complexity results apply as well to formula definability.

Some tractable restrictions can be derived from Proposition 25.

Proposition 33 *Whenever Σ belongs to a class $\mathcal{C}$ of formulas that is tractable for clausal query answering and stable for expansion with partial renaming (i.e. if Σ belongs to $\mathcal{C}$ and X is a set of variables, then $\Sigma \wedge rename(\Sigma, X)$ belongs to $\mathcal{C}$ as well), then the four above problems are in* P.

Accordingly, when Σ is restricted to a renamable Horn formula or a set of binary clauses, then BASIS CHECKING becomes tractable. This is also the case when Σ is in Disjunctive Normal Form.

Proposition 34 *Whenever Σ is a DNF formula, then all four definability problems are in* P.

More interestingly, when Σ belongs to one of the above classes, the *function problem* associated with BASIS CHECKING is in FP, i.e. computing a basis for Y w.r.t. Σ can be achieved in time polynomial in $|\Sigma| + |Y|$. Additionally, when such tractable restrictions are considered and X is a basis for Y w.r.t. Σ, the truth value of every variable y of Y can be computed in time polynomial in $|\Sigma| + |X|$ for every X-world ω_X. Indeed, when Σ is tractable for clausal query answering, cheking whether $\Sigma \models (\omega_X \Rightarrow y)$ holds can be done in polynomial time and the truth value of this test gives the truth value of y.

We have also derived the following related complexity results:

Proposition 35
(1) Deciding whether y is undefinable in Σ is NP*-complete.*
(2) Deciding whether x is a key variable for Y w.r.t. Σ is NP*-complete.*

6 Taking advantage of independence/dependence

The main reason why the results presented in this paper are significant is that independence/dependence (under its various forms) is central in knowledge representation and more generally in artificial intelligence [1]. Due to space limitations, we present in the following only some of the possible uses of the notions of independence/dependence discussed in the paper.

First, it is noticeable that in many works, the independence relation between propositional variables is explicit (part of the input), while ours is implicit (derived from the input). Thus, computing independence relations from a knowledge base can be seen as an upstream task enabling us specifying the "core" (minimal) independence relation; this core independence relation can then be completed by specifying explicitely some additional dependencies using knowledge about the domain. Such an approach has been proposed in [19], where the well-known *possible models approach* update operator is recovered by using the above principle with a specific implicit dependence relation (*novelty*, see [13] [18] for details). Using in a similar way the various implicit dependence relations given in Sections 3 and 4 should lead to a variety of belief update operators. This topic is left for further research.

Definability also has many applications to knowledge representation and reasoning. In a companion paper [16], the practical computation of defining families is investigated; we also introduce a closely related concept (however different), namely *hypothesis discrimination*, and discuss its role in various fields such as model-based diagnosis or decision under uncertainty.

Beyond independence and definability, there are other types of relations between propositional variables which are worth investigating. For instance, Boutilier [4] defines the notions of *controllability* and *influenceability* which are related to the notions of independence and definability (characterizations by means of prime implicates hold as well [4]). Influenceability, which appears to be strongly connected to formula-variable independence, is briefly evoked in Section 3. Complexity results for controllability are given in [16].

7 Conclusion

This paper has brought some formal contributions to the notion of logical independence, and has then defined and studied a new form of dependence, namely, definability. As to independence, several notions have been investigated. Some of which were already known from the works of Lakemeyer, Darwiche and Pearl, Boutilier, and others are new proposals. For each notion, characterizations results have been given – many of which make use of prime implicants / implicates – that helped us pointing out some connections with ab-

definability	without minimality	with minimality
without Σ satisfiable	coNP-complete	BH_2-complete
with Σ satisfiable	BH_2-complete	BH_2-complete

Table 2: complexity of definability.

duction, especially with the works of Eiter and Gottlob [10]. Moreover, the complexity of the different notions encountered has been analyzed. As to definability, several characterization results have been provided and complexity issues have been investigated.

Our results call for several remarks and further questions. First, the complexity results established tend to confirm that checking whether some (in)dependence relation holds is generally very hard, except in some restricted cases which are tractable. Formula-variable independence, which could appear at a first glance as an easy problem, is already coNP-complete. Conditional independence is even higher, since most problems are located at the second level of the polynomial hierarchy.

In this paper independence and definability notions were induced directly from the (purely propositional) knowledge base. Further work will explore the connections not only with probabilistic independence but also with notions based on qualitative uncertainty calculi [26] [9] or on utility-theoretic notions, and also with Bacchus and Grove's [2] notions of preferential independence. Roughly speaking, in [2] a set of variables X is preferentially independent of Y when one's preference order among X-worlds does not depend on the assignments of the variables of Y, which seems to have some similarities with some of the notions studied in this paper. Knowing that there is a growing interest for the logical representation of preferences, it would be worth studying whether logical independence relations can be satisfactorily extended in order to take account of such qualitative preferences.

Acknowledgements

The publication of this paper was strongly dependent of Andreas Herzig since he pointed out a flaw in a preliminary version. We thank him a lot. The second author has been supported in part by the IUT de Lens and a "Contrat d'objectifs de la région Nord/Pas-de-Calais".

References

[1] *Artificial Intelligence* 97(1&2), special issue on relevance, December 1997.

[2] Fahiem Bacchus and Adam J. Grove. Utility independence in a qualitative decision theory. In *Proceedings of the* 5^{th} *International Conference on Principles of Knowledge Representation and Reasoning (KR'96)*, Boston, 1996, 542-552.

[3] Evert W. Beth. On Padoa's method in the theory of definition. *Indagationes mathematicae* 15, 1953, 330-339.

[4] Craig Boutilier. Toward a logic for qualitative decision theory. In *Proceedings of the* 4^{th} *International Conference on Principles of Knowledge Representation and Reasoning (KR'94)*, Bonn, 1994, 75-86.

[5] Adnan Darwiche. A logical notion of conditional independence. *Proceedings of the AAAI Fall Symposium on Relevance*, New Orleans, 1994, 36-40

[6] Adnan Darwiche. Model-based diagnosis using causal networks. In *Proceedings of the* 14^{th} *International Joint Conference on Artificial Intelligence (IJCAI'95)*, Montreal, 1995, 211-217.

[7] Adnan Darwiche. A logical notion of conditional independence: properties and applications. *Artificial Intelligence* 97(1&2), 1997, 45–82.

[8] Adnan Darwiche and Judea Pearl. Symbolic causal networks. In *Proceedings of the* 12^{th} *National Conference on Artificial Intelligence (AAAI'94)*, Seattle (WA), 1994, 238-244.

[9] Didier Dubois, Luis Fariñas del Cerro, Andreas Herzig and Henri Prade. Qualitative relevance and independence: a roadmap. In *Proceedings of the* 15^{th} *International Joint Conference on Artificial Intelligence (IJCAI'97)*, Nagoya, 1997, 62-67.

[10] Thomas Eiter and Georg Gottlob. The complexity of logic-based abduction. *Journal of the ACM*, 42 (1), 1995, 3-42.

[11] Luis Fariñas del Cerro and Andreas Herzig. Belief change and dependence. In *Proceedings of the* 6^{th} *Conference on Theoretical Aspects of Reasoning about Knowledge (TARK'96)*, De Zeeuwse Stromen, 1996, 147-161.

[12] Peter Gärdenfors. Belief revision and irrelevance. *PSA*, 2:349-356.

[13] Russell Greiner and Michael R. Genesereth. What's new? A semantic definition of novelty. In *Proceedings of the* 8^{th} *International Joint Conference on Artificial Intelligence (IJCAI'83)*, Karlsruhe, 1983, 450–454.

[14] Gerhard Lakemeyer. A logical account of relevance. In *Proceedings of the* 14^{th} *International Joint Conference on Artificial Intelligence (IJCAI'95)*, Montreal, 1995, 853-859.

[15] Gerhard Lakemeyer. Relevance from an epistemic perspective. *Artificial Intelligence* 97(1–2), 1997, 137–167.

[16] Jérôme Lang and Pierre Marquis. Two notions of dependence in propositional logic: controllability and definability. *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI'98)*, Madison (WI), 1998.

[17] Fangzhen Lin and Ray Reiter. Forget it! In *Proceedings of the AAAI Fall Symposium on Relevance*, New Orleans, 1994, 154-159.

[18] Pierre Marquis. Novelty revisited. In *Proceedings of the* 6^{th} *International Symposium on Methodologies for Intelligent Systems (ISMIS'91)*, Charlotte(NC), Lecture Notes in Artificial Intelligence 542, Springer-Verlag, 1991, pp. 550-559.

[19] Pierre Marquis. Possible models approach via independency. In *Proceedings of the* 11^{th} *European Conference on Artificial Intelligence (ECAI'94)*, Amsterdam, 1994, 336-340.

[20] A. Padoa. Essai d'une théorie algébrique des nombres entiers, précédé d'une introduction logique à une théorie déductive quelconque. In *Bibliothèque du Congrès International de Philosophie*, Paris, 1903, 309-365.

[21] Christos H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[22] Ray Reiter and Johan de Kleer. Foundations of assumption-based truth maintenance systems: preliminary report. In *Proceedings of the* 6^{th} *National Conference on Artificial Intelligence (AAAI'87)*, Seattle (WA), 1987, 183–188.

[23] A. Padoa. Essai d'une théorie algébrique des nombres entiers, précédé d'une introduction logique à une théorie déductive quelconque. In *Bibliothèque du Congrès International de Philosophie*, Paris, 1903, 309-365.

[24] Christos H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[25] Ray Reiter and Johan de Kleer. Foundations of assumption-based truth maintenance systems: preliminary report. In *Proceedings of the* 6^{th} *National Conference on Artificial Intelligence (AAAI'87)*, Seattle (WA), 1987, 183–188.

[26] Milan Studený. Formal properties of conditional independence in different calculi of artificial intelligence. In *Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, Granada, Lecture Notes in Computer Science 747, Springer-Verlag, 1993, 341-348.