ELSEVIER International Journal of Approximate Reasoning 29 (2002) 135–173

INTERNATIONAL JOURNAL OF APPROXIMATE REASONING

www.elsevier.com/locate/ijar

On the transformation between possibilistic logic bases and possibilistic causal networks ☆

Salem Benferhat [a,*], Didier Dubois [a], Laurent Garcia [b], Henri Prade [a]

[a] *IRIT, Université Paul Sabatier, 118 Route de Narbonne, 31062 Toulouse Cedex 4, France*
[b] *LERIA, Université d'Angers, 2 Boulevard Lavoisier, 49045 Angers Cedex, France*

Received 1 August 2000; accepted 1 April 2001

Abstract

Possibilistic logic bases and possibilistic graphs are two different frameworks of interest for representing knowledge. The former ranks the pieces of knowledge (expressed by logical formulas) according to their level of certainty, while the latter exhibits relationships between variables. The two types of representation are semantically equivalent when they lead to the same possibility distribution (which rank-orders the possible interpretations). A possibility distribution can be decomposed using a chain rule which may be based on two different kinds of conditioning that exist in possibility theory (one based on the product in a numerical setting, one based on the minimum operation in a qualitative setting). These two types of conditioning induce two kinds of possibilistic graphs. This article deals with the links between the logical and the graphical frameworks in both numerical and quantitative settings. In both cases, a translation of these graphs into possibilistic bases is provided. The converse translation from a possibilistic knowledge base into a min-based graph is also described. © 2002 Elsevier Science Inc. All rights reserved.

☆ This paper is a fully revised and extended version of two conference papers [6,7].

* Corresponding author.

E-mail addresses: benferhat@irit.fr (S. Benferhat), dubois@irit.fr (D. Dubois), garcia@info.univ-angers.fr (L. Garcia), prade@irit.fr (H. Prade).

PII: S 0 8 8 8 - 6 1 3 X (0 1) 0 0 0 6 1 - 5

1. Introduction

Possibilistic logic is an offspring of Zadeh's possibility theory [31], which offers a framework for the representation of states of partial ignorance owing to the use of a dual pair of possibility and necessity measures [13]. Possibility theory may be quantitative or qualitative [14,19] according to the range of these measures which may be the real interval $[0, 1]$, or a finite linearly ordered scale as well. Possibilistic logic (e.g., [12]) has been developed for more than 10 years. It provides a sound and complete machinery for handling qualitative uncertainty with respect to a semantics expressed by means of possibility distributions which rank-order the possible interpretations. At the syntactic level, possibilistic logic handles pairs of the form (p, α), where p is a classical logic formula and α is an element of a totally ordered set. The pair (p, α) expresses that the formula p is certain at least to the level α, or more formally by $N(p) \geqslant \alpha$, where N is the necessity measure associated to the possibility distribution expressing the underlying semantics. Possibilistic logic is essentially qualitative since only the preordering induced on the formulas is important ($N(p) > N(q)$ means "p is more certain than q"). Possibilistic logic has a complexity slightly higher than classical logic since its complexity is about $\log_2 n * \mathrm{SAT}$ where n is the number of certainty levels used in the knowledge base and SAT is the complexity of the satisfaction problem in classical logic.

The notion of possibilistic graph for the representation of multidimensional possibility distributions is not new [17]. There are two kinds of representation: undirected graphs or hypergraphs ([17,21]) and directed acyclic graphs [20]. The latter are the counterparts of probabilistic Bayesian networks [26,23] in the framework of possibility theory. There are also recent techniques for learning possibilistic networks from imprecise data [21,27]. In probability theory, graphical structures are essential for efficient uncertainty propagation, because probabilistic logic is computationally very difficult to handle. However, because of the existence of the possibilistic logic machinery, there is not the same necessity to introduce graphical structures in possibility theory, for problems described by means of Boolean variables.

Yet Bayesian-like networks have a clear appeal for knowledge acquisition and could be useful in the case of possibilistic knowledge as much as the case of probabilistic knowledge.

The goal of this paper is to establish a bridge between directed possibilistic graphs and possibilistic logic. We wish to take advantage of the graphical representation, provided by directed graphs, while preserving the connection to a formal logical framework. Possibilistic logic has a kind of expressive power different from the one of directed possibilistic graphs, since, in the latter, knowledge must be structured as a directed acyclic graph (DAG). The basic information encoded in DAGs is a set of conditional possibility distributions whose aggregation defines a joint possibility distribution. However conditional possibility does not admit of a single definition. This article shows how to encode a directed possibilistic graph in possibilistic logic. We also give the converse transformation. The following section gives the necessary background on possibilistic logic. Section 3 discusses the notion of possibilistic conditioning. Section 4 introduces directed possibilistic graphs. Section 5 studies their encoding in possibilistic logic. Section 6 discusses the problem of recovering the initial conditional possibility distribution from the joint possibility distribution computed with the chain rule, and the discussion briefly refers to the idea of possibilistic independence. Section 7 proposes an encoding of a set of possibilistic logic formulas into a directed possibilistic graph. Proofs are provided in Appendix A.

2. Possibilistic logic

Let $\mathscr{L}$ be a finite propositional language. $p, q, r, \ldots$ denote propositional formulas. $\top$ and $\bot$, respectively, denote tautologies and contradictions. $\vdash$ denotes the classical syntactic inference relation. Ω is the set of classical interpretations ω of $\mathscr{L}$, and $[p]$ is the set of classical models of p (i.e, interpretations where p is true). In the following, we shall write $\omega \in [p]$ or $\omega \models p$ indifferently.

2.1. Possibility distributions and possibility measures

The basic element of possibility theory is the possibility distribution π which is a mapping from Ω to the interval $[0, 1]$. The degree $\pi(\omega)$ represents the compatibility of ω with the available information (or beliefs) about the real world. By convention, $\pi(\omega) = 0$ means that the interpretation ω is impossible, and $\pi(\omega) = 1$ means that nothing prevents ω from being the real world. When $\pi(\omega) > \pi(\omega')$, ω is a preferred candidate to ω' for being the real state of the world. A possibility distribution π is said to be normal if $\exists \omega \in \Omega$, such that $\pi(\omega) = 1$, namely there exists at least one interpretation which is consistent with all the available beliefs.

Given a possibility distribution π, we can define two different ways of rank-ordering formulas of the language from this possibility distribution. This is obtained using two mappings grading, respectively, the possibility and the certainty of a formula p:

- the possibility (or consistency) degree $\Pi(p) = \max\{\pi(\omega) : \omega \in [p]\}$ which evaluates the extent to which p is consistent with the available beliefs expressed by π [31]. It satisfies:

  $$\forall p, \ \forall q \quad \Pi(p \vee q) = \max(\Pi(p), \Pi(q));$$

- the necessity (or certainty, entailment) degree $N(p) = 1 - \Pi(\neg p)$ which evaluates the extent to which p is entailed by the available beliefs. We have [13]:

  $$\forall p, \ \forall q \quad N(p \wedge q) = \min(N(p), N(q)).$$

2.2. Possibilistic knowledge bases

A possibilistic knowledge base is a finite set of weighted formulas

$$\Sigma = \{(p_i, \alpha_i),\ i = 1, m\},$$

where $\alpha_i > 0$ is understood as a lower bound on the degree of necessity $N(p_i)$. Formulas with zero degree are not explicitly represented in the knowledge base (only beliefs which are somewhat accepted are explicitly represented). The higher the weight, the more certain the formula.

Definition 1. Let Σ be a possibilistic knowledge base, and $\alpha \in [0, 1]$. We call the α-cut (resp. strict α-cut) of Σ, denoted by $\Sigma_{\geqslant\alpha}$ (resp. by $\Sigma_{>\alpha}$), the set of classical formulas in Σ having a certainty degree at least equal to (resp. strictly greater than) α.

A possibilistic knowledge base Σ is said to be consistent if its classical support, obtained by forgetting the weights, is classically consistent. We define by

$$\mathrm{Inc}(\Sigma) = \max\{\alpha_i : \Sigma_{\geqslant\alpha_i} \vdash \bot\}$$

the inconsistency degree of Σ. $\mathrm{Inc}(\Sigma) = 0$ means that $\Sigma_{\geqslant\alpha_i}$ is consistent for all α_i.

From a possibilistic knowledge base, a syntactic possibilistic entailment has been defined as follows:

Definition 2. Let Σ be a consistent possibilistic knowledge base and (p, α) a possibilistic formula (with p a classical formula and $\alpha \in (0, 1]$), p is entailed from Σ to degree α denoted by $\Sigma \vdash (p, \alpha)$, iff $\Sigma_{\geqslant\alpha} \cup \{\neg p\}$ is inconsistent.

When Σ is inconsistent, then (p, α) is a non-trivial consequence of Σ, if and only if

(i) $\mathrm{Inc}(\Sigma \cup \{(\neg p, 1)\}) > \mathrm{Inc}(\Sigma)$ and

(ii) $\mathrm{Inc}(\Sigma \cup \{(\neg p, 1)\}) \geqslant \alpha$.

This consequence relation is closely related to the so-called rational entailment in the sense of Lehmann and Magidor [24]; see [13,18].

2.3. From possibilistic knowledge bases to possibility distributions

Given a possibilistic knowledges base Σ, we can generate a unique possibility distribution by associating to each interpretation, its level of compatibility with explicit beliefs, i.e., with Σ. When a possibilistic knowledge base only consists of one formula $\{(p, \alpha)\}$, then each interpretation ω which satisfies p should have possibility degree $\pi(\omega) = 1$, since it is consistent with p. Each interpretation ω which falsifies p should have a possibility degree $\pi(\omega)$ such that the higher α (i.e., the more certain p), the lower $\pi(\omega)$; in particular, if $\alpha = 1$ (i.e., p is completely certain), then $\pi(\omega) = 0$, namely ω is impossible. One way to realize this constraint is to assign the degree $1 - \alpha$ to $\pi(\omega)$ with a numerical encoding. Therefore, the possibility distribution associated with $\Sigma = \{(p, \alpha)\}$ is $\forall \omega \in \Omega$,

$$\pi_{\{(p,\alpha)\}}(\omega) = \begin{cases} 1 & \text{if } \omega \in [p] \text{ (i.e. } \omega \text{ satisfies } p), \\ 1 - \alpha & \text{otherwise (i.e. } \omega \text{ falsifies } p). \end{cases}$$

When $\Sigma = \{(p_i, \alpha_i),\ i = 1, m\}$ is a general possibilistic knowledge base, then all the interpretations satisfying all the beliefs in Σ have the highest possibility degree, namely 1, and the other interpretations will be ranked w.r.t. the most certain belief that they falsify, namely we get [12]:

Definition 3. The possibility distribution associated with a knowledge base Σ is defined, $\forall \omega \in \Omega$, by

$$\pi_{\{(p,\alpha)\}}(\omega) = \begin{cases} 1 & \text{if } \omega \in [p_i]\ \forall (p_i, \alpha_i) \in \Sigma, \\ 1 - \max\{\alpha_i : (p_i, \alpha_i) \in \Sigma;\ \omega \notin [p_i]\} & \text{otherwise.} \end{cases}$$

Thus, π_Σ can be viewed as the result of the combination of the $\pi_{\{(p_i,\alpha_i)\}}$'s using the minimum operator, that is

$$\pi_\Sigma(\omega) = \min\{\pi_{\{(p_i,\alpha_i)\}}(\omega) : (p_i, \alpha_i) \in \Sigma\}.$$

Example 1. Let $\Sigma = \{(q, 0.3), (q \vee r, 0.5)\}$. Then

$$\pi_\Sigma(qr) = \pi_\Sigma(q\neg r) = 1; \quad \pi_\Sigma(\neg qr) = 0.7; \quad \pi_\Sigma(\neg q \neg r) = 0.5.$$

The two interpretations qr and $q\neg r$ are the preferred ones since they are the only ones which are consistent with Σ, and $\neg qr$ is preferred to $\neg q \neg r$, since the highest belief falsified by $\neg qr$ (i.e., $(q, 0.3)$) is less certain than the highest belief falsified by $\neg q \neg r$ (i.e., $(q \vee r,\ 0.5)$).

Note that the possibility distribution π_Σ is not necessarily normalized. Namely, π_Σ is normalized if and only if Σ is consistent. Moreover, it can be verified that

$$\mathrm{Inc}(\Sigma) = 1 - \max_\omega \pi_\Sigma(\omega).$$

Definition 4. Two possibilistic knowledge bases Σ and Σ' are said to be semantically equivalent if and only if $\pi_\Sigma = \pi_{\Sigma'}$.

Finally, it can be checked that [12]

$$\Sigma \vdash (p, \alpha) \quad \text{iff } \pi_\Sigma \leqslant \pi_{\{(p,\alpha)\}}.$$

The following definition and lemmas are useful for the rest of the paper.

Definition 5. Let (p, α) be a belief in Σ. Then (p, α) is said to be subsumed by Σ if

$$(\Sigma - \{(p, \alpha)\})_{\geqslant \alpha} \vdash p.$$

Similarly, (p, α) is said to be strictly subsumed by Σ if $\Sigma_{>\alpha} \vdash p$.

Lemma 1. [1]*Let (p, α) be a subsumed belief of Σ. Then Σ and $\Sigma' = \Sigma - \{(p, \alpha)\}$ are equivalent.*

As a corollary of the previous lemma, we can add or remove subsumed beliefs without changing the possibility distribution. This means that several syntactically different possibilistic knowledge bases may have the same possibility distribution as their semantic counterpart. In such a case, it can be shown that their α-cuts, which are classical knowledge bases, are logically equivalent in the usual sense. The following lemma exhibits similar conclusions when we remove tautologies from knowledge bases.

Lemma 2. *Let $(\top, \alpha)$ be a tautological belief of Σ. Then Σ and $\Sigma' = \Sigma - \{(\top, \alpha)\}$ are equivalent.*

The proof is obvious since tautologies are satisfied by each interpretation and using Definition 3 only formulas which are falsified by a given interpretation are taken into account when computing the possibility degree of this interpretation. In fact, since $\pi_{\{(\top,\alpha)\}} = 1 \ \forall \alpha$; $\pi_{\{(\top,\alpha)\}}$ has no effect on π_Σ.

The following proposition shows that if a knowledge base does not contain strictly subsumed formulas, then the weights associated with formulas in the knowledge base Σ is the same as the one obtained from the possibility distribution associated with Σ. Let N_{π_Σ} be the necessity measure induced by π_Σ.

Proposition 1. *Let Σ be such that it does not contain any strictly subsumed formulas. Then $N_{\pi_\Sigma}(p) = \alpha \ \forall (p, \alpha) \in \Sigma$.*

Knowledge bases which do not contain strictly subsumed formulas corresponds to the so-called partial epistemic entrenchments in [30].

3. Possibilistic conditioning

In this section, we discuss the basic notion used in directed possibilistic graphs, i.e. conditioning in the framework of possibility theory. In the

[1] All the proofs of lemmas and propositions of this paper are given in Appendix A.

remaining of this paper we consider only non-dogmatic possibility distributions π on Ω, such that $\pi(\omega) > 0$ for all $\omega \in \Omega$. They are possibility distributions which exclude no interpretation, be it very implausible. Since we assume only non-dogmatic possibility distributions, it implies that $\pi(p) > 0$ as soon as p is not a contradiction.

3.1. Definitions

The notion of conditioning is crucial in probability theory. It is usually expressed by the following Bayesian equation:

$$P(q|p) = \frac{P(p \wedge q)}{P(p)} \quad \text{for } P(p) > 0.$$

In possibility theory, there exist several definitions for conditioning, depending on whether the setting is qualitative or numerical.

Viewed as a revision process, conditioning in possibility theory transforms a possibility distribution π and a new and totally reliable information $p \neq \bot$ into a new possibility distribution denoted by $\pi' = \pi(\cdot|p)$. Natural properties for π' are:

(A_1) π' should be normalized;
(A_2) $\forall \omega \notin [p],\ \pi'(\omega) = 0$;
(A_3) $\forall \omega,\ \omega' \in [p],\ \pi(\omega) > \pi(\omega')$ if and only if $\pi'(\omega) > \pi'(\omega')$;
(A_4) if $\Pi(p) = 1$, then $\forall \omega \in [p]$, $\pi(\omega) = \pi'(\omega)$;
(A_5) $\forall \omega \in \Omega$ if $\pi(\omega) = 0$, then $\pi'(\omega) = 0$.

A_1 says that the new state of knowledge is consistent. A_2 confirms that p is a totally reliable piece of information. A_3 says that the new possibility distribution should not alter the previous relative order between models of p. A_4 says that if p is completely possible, then the revision does not affect π on the models of p. A_5 says that impossible worlds remain impossible after conditioning. A_5 does not apply to non-dogmatic possibility distributions.

The above properties do not guarantee a unique definition of conditioning. Indeed, the effect of the axiom A_2 may result in a sub-normalized possibility distribution, due to discarding the best models of π. Restoring the normalisation, so as to satisfy A_1, can be done in two different ways:

- In an ordinal setting, we assign the maximal possibility level to the best models of $p \neq \bot$, and we get [16]:

$$\pi(\omega|_m p) = \begin{cases} 1 & \text{if } \pi(\omega) = \Pi(p),\ \omega \in [p], \\ \pi(\omega) & \text{if } \pi(\omega) < \Pi(p),\ \omega \in [p], \\ 0 & \text{if } \omega \notin [p]. \end{cases}$$

This definition of conditioning will be called min-based conditioning. It satisfies the above five axioms.

- In a numerical setting, we proportionally shift up all *models of p*:

$$\pi(\omega|_* p) = \begin{cases} \frac{\pi(\omega)}{\Pi(p)} & \text{if } \omega \in [p], \\ 0 & \text{otherwise.} \end{cases}$$

This definition of conditioning will be called product-based conditioning. It also satisfies the above five axioms.

Both conditionings satisfy an equation of the form

$$\Pi(q) = \Box(\Pi(q|p), \Pi(p)), \tag{1}$$

which is similar to Bayesian conditioning, respectively, for $\Box = \min$ [22] and the product (denoted by $*$, or omitted, in the following). The rule based on the product is much closer to genuine Bayesian conditioning than the qualitative conditioning based on the minimum which is purely based on comparing levels; conditioning based on the product requires more of the structure of the unit interval and is a special case of Dempster rule of conditioning [29]. When conditioning is based on the minimum, the equation does not lead to a unique definition of conditioning. The solution given by "$|_m$" is the least specific (i.e., the greatest) solution satisfying (1).

3.2. Conditioning = combining + normalizing

Possibilistic definitions of conditioning can be retrieved from, first, a combination of possibility distributions with the minimum operator, followed by an operation of normalization. Let π be the initial possibility distribution and π_p the possibility distribution expressing that the piece of information p is sure (that is $\pi_p(\omega) = 1$ if $\omega \in [p]$ and $\pi_p(\omega) = 0$ otherwise). The conjunctive combination is defined by

$$\pi_{\text{conj}}(\omega) = \min(\pi(\omega), \pi_p(\omega)).$$

It is easy to verify that the following operation of normalization, denoted by N1, applied to π_{conj} gives exactly the definition of min-based conditioning:

$$\pi^{\text{N1}}(\omega) = \begin{cases} 1 & \text{if } \pi_{\text{conj}}(\omega) = h(\pi_{\text{conj}}), \\ \pi_{\text{conj}}(\omega) & \text{otherwise.} \end{cases}$$

with $h(\pi_{\text{conj}}) = \max_{\omega'} \pi_{\text{conj}}(\omega')$.

When conditioning is based on the product, the normalization, denoted by N2, is made with

$$\pi^{\text{N2}}(\omega) = \frac{\pi_{\text{conj}}(\omega)}{h(\pi_{\text{conj}})},$$

($h(\pi_{\text{conj}})$ is supposed to be positive since π is non-dogmatic).

Note that in general, the combination of two possibility distributions with the minimum operator, followed by a normalisation step N1 is not an associative process [15,28], while the combination with the product followed by the normalisation step N2 is an associative operator [15].

However, both modes of conditioning are associative as shown by the following proposition:

Proposition 2. *Let π be a (non-dogmatic) possibility distribution, let p and q be two non-mutually exclusive propositions. Let $\pi_1 = \pi(\cdot|p)$ and $\pi_2 = \pi_1(\cdot|q)$. Let $\pi_3 = \pi(\cdot|q)$ and $\pi_4 = \pi_3(\cdot|p)$.*

Then $\pi_2 = \pi_4$.

This proposition is not valid when conditioning on mutually exclusive propositions p and q. Indeed, then $\pi_2(\omega) = 1$ if $\omega \models q$ and 0 otherwise, while $\pi_4(\omega) = 1$ if $\omega \models p$ and 0 otherwise. It is not valid either for dogmatic possibilistic distributions such that $\Pi(r) = 0$ for some non-contradictory proposition r. Indeed suppose p and r, p and q, are two logically independent pairs, but, $q \models r$. Suppose also $\pi(\omega) > 0$ whenever $\omega \models \neg r$. Then $\pi_1(\omega) > 0$ for and only for $\omega \models p \wedge \neg r$. Hence $\pi_2(\omega) = 1$ if and only if $\omega \models q$ and 0 otherwise. However, $\pi_3(\omega) = \pi_2(\omega)$ since $q \wedge \neg r = \bot$. Hence $\pi_4(\omega) = 1$ if and only if $\omega \models p \wedge q$ and 0 otherwise. So, π_2 and π_4 differ.

Remark. Note that if we allow to deal with sub-normalized possibility distributions (namely A_1 is no longer required), then there exists an obvious definition of conditioning given by

$$\pi(\omega|p) = \begin{cases} \pi(\omega) & \text{if } \omega \in [p], \\ 0 & \text{otherwise.} \end{cases}$$

However, with this definition we no longer have $\Pi(p|p) = 1$, and conditioning would be nothing but a particular case of intersection.

We have defined conditioning upon formulas and interpretations. In the following, we will also use conditioning on variables. This corresponds to defining it for every possible instanciation of these variables. For example, let A and B be two Boolean variables with domains $D_A = \{a, \neg a\}$ and $D_B = \{b, \neg b\}$, the conditional possibility distribution $\Pi(B|A)$ corresponds to the four values $\Pi(b|a)$, $\Pi(\neg b|a)$, $\Pi(b|\neg a)$ and $\Pi(\neg b|\neg a)$. We will use the notation $\Pi(B|A)$ (instead of $\pi(B|A)$) for conditional possibility distributions, since the instanciations of B are literals, usually concerning more than one interpretation in Ω (the set of interpretations induced by the language used).

3.3. Conditioning and material implication

In probability theory, it is clear that conditioning is different from material implication. Namely, $P(\neg p \vee q) \neq P(q|p)$, except if $P(\neg q \wedge p) = 0$ for which the two probabilities are equal to 1. For instance, it is easy to show that the set of constraints $P(a) = r \in [0, 1]$, $P(b|a) = s \in [0, 1]$, $P(b|\neg a) = t \in [0, 1]$, always admits of a solution. The reason is straightforward: the first constraint defines a probability distribution on D_A while the others define a probability distribution on other domains (for each instance of the variable A, a probability distribution on a copy of D_B is defined). The set of three constraints is consistent since the a priori probability distribution $P(A)$ and the conditional probability distributions $P(B|a)$ and $P(B|\neg a)$ are defined, in some sense, on *independent languages*.

However, as pointed out by Pearl [26], the set of constraints $P(a) = s \in [0, 1]$ and $P(\neg a \vee b) = t \in [0, 1]$ does not always admit of a solution (for example: $P(a) = 0.01$ and $P(\neg a \vee b) = 0.9$). The reason is that, here, the two constraints are applied to the same probability distribution, that is on *the same domain* $D_A \times D_B$.

The links between min-based conditioning and material implication in possibility theory are summarized in the following proposition:

Proposition 3. *For every normalized possibility distribution* π, *we have* $\Pi(\neg p \vee q) \geqslant \Pi(q|_m p) \geqslant \Pi(p \wedge q)$.

Moreover, it can be easily checked that:
- $\Pi(q|_m p) = \Pi(p \wedge q)$ *if and only if* $\Pi(p \wedge q) < \Pi(p \wedge \neg q)$ *or* $\Pi(p \wedge q) = 1$,
- $\Pi(q|_m p) = \Pi(\neg p \vee q)$ *if and only if* $\Pi(p \wedge q) \geqslant \Pi(p \wedge \neg q)$ *or* $\Pi(p \wedge \neg q) > \Pi(p \wedge q) \geqslant \Pi(\neg p)$.

It is clear that $\Pi(q|_m p)$ and $\Pi(q)$ cannot be independently specified, since it is forbidden to have $\Pi(p) \leqslant \Pi(p|_m q) < 1$. However, if the product-based conditioning is used, $\Pi(q|_* p)$ and $\Pi(p)$ are again independent quantities, like in probability theory.

3.4. Decomposition based on the minimum

The decomposition of a possibility distribution consists in expressing a joint possibility distribution as a combination of conditional possibility distributions. For this purpose, we can follow the same procedure as in probability theory. Let $\{A_1, \ldots, A_n\}$ be the set of variables which is ordered arbitrarily. From the definition of min-based conditioning and for positive possibility distributions, we have

$$\pi(A_1 \cdots A_n) = \min[\Pi(A_1|A_2 \cdots A_n), \Pi(A_2 \cdots A_n)].$$

When applying this definition to $\Pi(A_2 \cdots A_n)$ repeatedly, then to $\Pi(A_3 \cdots A_n), \ldots, \Pi(A_{n-1}A_n)$ the joint possibility distribution is decomposed into

$$\pi(A_1 \cdots A_n) = \min[\Pi(A_1|A_2 \cdots A_n), \ldots, \Pi(A_{n-1}|A_n), \Pi(A_n)]. \tag{2}$$

This means that a possibility distribution $\pi(A_1 \cdots A_n)$ can be seen as the combination, by the minimum operator, of conditional possibility distributions $\Pi(A_i|A_{i+1} \cdots A_n)$.

It is obvious that, by construction, the conditional possibility distribution $\Pi(A_i|A_{i+1} \cdots A_n)$ associated with each variable A_i is normalized, that is

$$\forall (a_{i+1}, \ldots, a_n) \in D_{i+1} \times \cdots \times D_n, \quad \max_{a_k \in A_i} \Pi(a_k|a_{i+1}, \ldots, a_n) = 1.$$

The decomposition given by Eq. (2) can be simplified by assuming conditional independence between variables. For instance, if A_1 is independent of $A_{i+1} \cdots A_n$ in the context $A_2 \cdots A_i$, then the expression $\Pi(A_1|A_2 \cdots A_n)$ can be simplified into $\Pi(A_1|A_2 \cdots A_i)$. Some results on the possibilistic independence relation are given in Section 6.

There is not a unique decomposition of a possibility distribution since it depends on the initial ordering between variables. Indeed, let us consider the following example with π defined on $\{a, \neg a\} \times \{b, \neg b\}$ such that

$$\pi(\neg a, \neg b) = 1; \quad \pi(a, b) = \pi(a, \neg b) = 0.8; \quad \pi(\neg a, b) = 0.7.$$

There are two ways for decomposing this possibility distribution,

- either $\Pi(A, B) = \min(\Pi(B|A), \pi(A))$, and then $\Pi(B|A)$ is described by the matrix:

$B\|A$	a	$\neg a$
b	1	0.7
$\neg b$	1	1

  and $\pi(A)$ by the values $\pi(a) = 0.8$ and $\pi(\neg a) = 1$,
- or $\pi(A, B) = \min(\Pi(A|B), \pi(B))$, and then $\Pi(A|B)$ is described by the matrix:

$A\|B$	b	$\neg b$
a	1	0.8
$\neg a$	0.7	1

  and $\pi(B)$ by the values $\pi(b) = 0.8$ and $\pi(\neg b) = 1$.

It turns out that these two possible decompositions of the possibility distribution π correspond to the two following possibilistic knowledge bases (as it can be verified using the procedure presented in the following in Section 5.2):

$$\Sigma_1 = \{(\neg b \vee a, 0.3), (\neg a, 0.2)\},$$
$$\Sigma_2 = \{(b \vee \neg a, 0.2), (\neg b \vee a, 0.3), (\neg b, 0.2)\}.$$

These two possibilistic knowledge bases are semantically equivalent since they give the same possibility distribution π (obtained by using Definition 3), which can be also recovered by applying (2). We will see in the following section how to generate possibilistic knowledge bases from a set of conditional possibility distributions. It can be checked that $\pi(A|B) < \pi(B)$ is always obtained, unless $\Pi(A|B) = 1$.

Remark. When conditioning with the product, the decomposition follows the way used in probability

$$\pi(A_1 \cdots A_n) = \Pi(A_1|A_2 \cdots A_n) * \cdots * \Pi(A_{n-1}|A_n) * \Pi(A_n).$$

4. Directed possibilistic graphs

4.1. Background and notations on probabilistic causal networks

We first give some notations and definitions used in the remaining of this article. Let $V = \{A_1, A_2, \ldots, A_n\}$ be a set of variables. We denote by D_A the supposedly finite domain of the variable A and D_i the domain of the variable A_i, for short, instead of D_{A_i}. For Boolean variables, a_i denotes any of the two instances of A_i and $\neg a_i$ represents the other instance of A_i. $X, Y, Z, \ldots$ denote subsets of variables from V, and $D_X = \times_{A_i \in X} D_i = \{x_1, x_2, \ldots, x_m\}$ represents the Cartesian product of variable domains in X. By x we denote any instance of X. The set of interpretations $\Omega = \times_{A_i \in V} D_i$ is simply the Cartesian product of all variable domains in V. Depending on the context, interpretations are denoted either by tuples: $\omega = (a_1, \ldots, a_n)$ or by conjunctions: $\omega = a_1 \wedge \cdots \wedge a_n$. If X is a subset of variables, and x an instance of X which is the projection of ω on X, then we write $\omega \models x$, or $x = \omega_X$. In particular, $\omega \models a_i$, or $a_i = \omega_A$, means that a_i appears in ω. More generally, if $X \subseteq Y$ and the projection of instance y on X is x, we write $y \models x$.

This section briefly recalls the basic ideas of underlying probabilistic causal networks [9,10,23,26] which are DAGs. The nodes represent variables (for example, the temperature of a patient, the color of a car,...) and the edges encode causal links (or influences) between these variables. Uncertainty is represented on each node and expresses in a causal language the strength of the link between variables. When there is an edge from node A_i to node A_j, node A_i is said to be a parent of A_j. Parents of A_j are denoted by $\mathrm{Par}(A_j)$, and by u_j we denote any instance of $\mathrm{Par}(A_j)$, namely any element of the Cartesian product $\times_{A_j \in \mathrm{Par}(A_i)} D_j$. A probability measure P is associated with a graph G in the following way:

- For the nodes A_i which are roots of the graph (i.e., $\mathrm{Par}(A_i) = \emptyset$), we specify the a priori probability degrees associated with each instance of A_i, namely we supply every $P(a_i)$ where $a_i \in D_i$. The probabilities must satisfy the normalization condition, i.e. $\sum_{a_i \in D_i} P(a_i) = 1$.
- For every other node A_i of the graph, we prescribe every conditional probability $P(a_i|u_i)$ where $a_i \in D_i$ and u_i range on all instances of $\mathrm{Par}(A_i)$. Conditional probabilities should satisfy the following normalization condition:

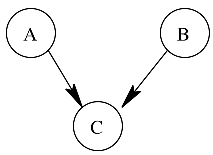
$$\forall u_i \in \times_{A_j \in \mathrm{Par}(A_i)} D_j, \quad \sum_{a_i \in D_i} P(a_i | u_i) = 1.$$

From these (a priori and conditional) probabilities associated with the DAG, the definition of the joint probability distribution p_{BN} (defined on every elementary event) is given by the following expression, called probabilistic chain rule

$$p_{BN}(a_1, \ldots, a_n) = \prod_{i=1,n} P(a_i | \omega_{\mathrm{Par}(i)}),$$

where a_i is a possible instance of A_i, $\prod$ denotes the product, $\omega_{\mathrm{Par}(i)}$ is the projection of $(a_1, \ldots, a_n)$ on $\times_{A_j \in \mathrm{Par}(A_i)} D_j$.

Example 2. Consider the following DAG:



The set of Boolean variables is $V = \{A, B, C\}$. We need the values of the following a priori and conditional probabilities: $P(a)$, $P(b)$ and four values $P(c|a,b)$ for $(a,b) \in D_A \times D_B$. Then, the joint probability distribution is computed by: $p_{BN}(a_i, b_j, c_k) = P(c_k | a_i, b_j) * P(a_i) * P(b_j)$, with $a_i \in D_A$, $bj \in D_B$ and $c_k \in D_C$. For example, $p_{BN}(a, b, \neg c) = P(\neg c | a, b) * P(a) * P(b)$.

The set of probabilistic constraints is always consistent and admits of a solution. The joint probability distribution p_{BN} given by the previous equation allows to recover a priori and conditional probabilities given by the expert. The consistency of constraints requires the use of DAGs [26]. Nevertheless, if we had interpreted conditioning by material implication, the existence of a solution would not be guaranteed even for a DAG structure, as seen in Section 3.3.

4.2. Definition of directed possibilistic graphs

We now define the directed possibilistic graphs, denoted by ΠG, which are the possibilistic counterparts of previous causal networks. We also use a DAG structure. The difference with probabilistic causal networks comes from the uncertainty measure which is used: a priori (resp. conditional) probabilities are replaced by a priori (resp. conditional) possibilities. Let us notice that it is possible to express ignorance, i.e. it is not necessary to give the a priori possibilities on a variable A if they are unknown. In such a case, they are equal to 1 for all possible instances of the variable). In the rest of the paper variables are restricted to binary ones since we are concerned with the bridge between possibilistic nets and possibilistic logic.

Remark. The restriction to binary variables is made for the purpose of relating possibilistic graphs and possibilistic logic. The results of this section can be extended to non-binary variables. Clearly, one of the expected computational advantages of developing possibilistic causal networks concerns the easier handling of non-binary variables.

The normalization conditions on each variable $A_i = \{a_i, \neg a_i\}$ of the graph are:

- $\max(\Pi(a_i), \Pi(\neg a_i)) = 1$ (for a priori possibilities concerning root nodes A_i), and
- for each $u_i \in \times_{A_j \in \mathrm{Par}(A_i)} D_j$, $\max(\Pi(a_i|u_i), \Pi(\neg a_i|u_i)) = 1$ (for conditional possibilities concerning non-rooted nodes A_i).

In practice, data are not given in terms of conditional possibilities but in terms of conditional necessities. Indeed, it is more convenient for experts to express what they know for sure, instead of what is possible and consistent with their knowledge. However, conditional possibilities are immediately derived from conditional necessities using the duality property $\Pi(q|p) = 1 - N(\neg q|p)$.

The two kinds of conditioning used in possibility theory lead to two different ways of computing the joint possibility distributions.

Definition 6. Let ΠG be a directed possibilistic acyclic graph. Let $\omega = (a_1, \ldots, a_n)$ be a given interpretation. Then, the joint possibility distribution associated with a ΠG is computed with the following equation, called *chain rule*:

$$\pi(\omega) = \square_{i=1,n} \Pi(a_i|\omega_{\mathrm{Par}(i)}),$$

where $\square$ represents either the minimum or the product operator, and $\Pi(a_i|\omega_{\mathrm{Par}(i)})$ are conditional possibilities in ΠG.

When the joint possibility distribution is computed with the minimum (resp. product), the ΠG will be denoted by ΠG_m (resp. ΠG^*).

When conditioning in possibility theory is defined via the product, the computation of the joint possibility distribution associated with ΠG is the same as in probability theory. The reason is straightforward since the basic tools used in the two theories for computing the joint distribution are the same.

Directed possibilistic graphs have been defined. The following section studies their encoding in possibilistic logic.

5. Encoding possibilistic graphs in possibilistic logic

The goal of this section is to translate a directed possibilistic graph into a possibilistic logic base. We start by general considerations and new notations. Then we provide the translation for the two kinds of possibilistic graphs.

5.1. The basic methodology

The starting point is that the directed possibilistic base associated to a directed possibilistic graph is the result of the fusion of elementary bases. These elementary bases are composed of formulas associated to the prior and conditional possibilities attached to nodes of the directed possibilistic graph.

Let $\{A_1, \ldots, A_n\}$ be the set of all binary variables of ΠG. For the sake of simplicity, a possibilistic causal network will be represented by a set of triples, $\Pi G = \{(a_i, u_i, \alpha_i) : \alpha_i = \Pi(a_i|u_i) \neq 1$ is an element of the graph$\}$, [2] where a_i is an instance of the variable A_i and u_i is an element of the Cartesian product of the domains D_j of the variables $A_j \in \mathrm{Par}(A_i)$.

With each isolated triple (a, u, α) of the directed possibilistic graph, where a is an instance of variable A, is associated the single possibilistic formula $(\neg a \vee \neg u, 1 - \alpha)$. Clearly, its associated possibility distribution $\pi_{a,u}$ is

$$\pi_{a,u}(\omega) = \begin{cases} 1 & \text{if } \omega \models \neg a \vee \neg u, \\ \alpha & \text{otherwise (that is if } \omega \models a \wedge u). \end{cases}$$

It is easy to check that the conditional possibilities are recovered from $\pi_{a,u}$, independently of the definition of conditioning. Indeed, $\Pi_{a,u}(a \wedge u) = \alpha$ holds since every interpretation which satisfies $a \wedge u$ falsifies $(\neg a \vee \neg u, 1 - \alpha)$. Moreover, $\Pi_{a,u}(u) = \max(\Pi_{a,u}(a \wedge u), \Pi_{a,u}(\neg a \wedge u))$. As every interpretation satisfying $\neg a \wedge u$ satisfies $(\neg a \vee \neg u, 1 - \alpha)$, we get $\Pi_{a,u}(u) = 1$. Note that $\Pi_{a,u}(u) = 1$ only because we consider an isolated triple (a, u, α), where the parents of the parents of the variable A are not considered.

Therefore, if we locally apply min-based conditioning we get: $\Pi_{a,u}(a|u) = \alpha$ since $\Pi_{a,u}(a \wedge u) < \Pi_{a,u}(u)$. Now if we apply the product-based conditioning, we have $\Pi_{a,u}(a|u) = \Pi_{a,u}(a \wedge u)/\Pi_{a,u}(u) = \alpha$.

---

[2] The restriction to the conditional possibilities different from 1 is done since only these conditional possibilities are used for the computation of joint possibility distributions (since 1 is a neutral element w.r.t. both minimum and product operator).

5.2. Logical encoding of ΠG_m

First we study the case where the fusion of possibility distributions associated with knowledge bases is made by means of the minimum operation.

The next proposition shows that the computation (given by Definition 6) of the joint possibility distribution obtained from a directed possibilistic graph is equivalent to the one obtained by combination, with the minimum, of the joint possibility distributions associated with the formulas encoding the different triples of the directed graph. More formally:

Proposition 4. *Let π_i be the possibility distribution associated with the possibilistic formula (a_i, u_i, α_i). Then the joint possibility distribution computed from the directed graph ΠG_m is the same as the one obtained by combining the possibility distributions π_i's with the minimum operator.*

This proposition is easy to prove. Indeed, in the previous subsection it has been noticed that the value of the local possibility distribution $\pi_i(a_i \wedge u_i)$ induced by (a_i, u_i, α_i) coincides with the (local) conditional possibility value $\Pi(a_i|u_i)$. Let $\omega = \{a_1, \ldots, a_n\}$ be a possible interpretation. The possibility degree of the interpretation ω, obtained by combining local conditional possibility distributions by the chain rule is thus equal to

$$\begin{aligned}\pi(\omega) &= \min\{\alpha_i : (a_i, u_i, \alpha_i) \in \Pi G_m, \omega \models \neg a_i \vee \neg u_i\}.\\ &= \min_{i=1,n} \pi_i.\end{aligned}$$

The following lemma describes the possibilistic knowledge base associated with the combination of two possibility distributions by the minimum:

Lemma 3. *Let Σ_1 and Σ_2 be two possibilistic knowledge bases. Let π_1 and π_2 be the two possibility distributions associated with Σ_1 and Σ_2, respectively. Let $\pi_{\min}$ be the combination with the minimum operator of π_1 and π_2, that is $\forall\omega$, $\pi_{\min}(\omega) = \min(\pi_1(\omega), \pi_2(\omega))$. The resulting knowledge base corresponding to $\pi_{\min}$ is simply: $\Sigma_{\min} = \Sigma_1 \cup \Sigma_2$.*

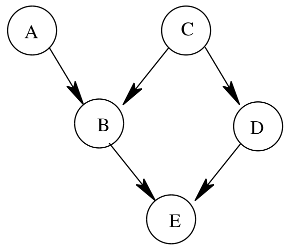
The proof is easy and can be found for instance in [4,5]. Using the previous proposition and lemma, the following corollary states what is exactly the knowledge base associated with a directed possibilistic graph:

Corollary 1. *The possibilistic knowledge base associated with $\Pi G_m = \{(a_i, u_i, \alpha_i) : \Pi(a_i|u_i) = \alpha_i \neq 1\}$ is*

$$\Sigma = \{(\neg a_i \vee \neg u_i, 1 - \alpha_i)) : (a_i, u_i, \alpha_i) \in \Pi G_m\}.$$

This result is important since it implies that the possibilistic logic reasoning machinery can be applied to directed possibilistic graphs.

Example 3. Consider the following DAG:



Assume that the conditional possibility degrees are given by the following tables:

$\Pi(A)$

a	1
$\neg a$	3/4

$\Pi(C)$

c	1
$\neg c$	1

$\Pi(B|AC)$

$B|AC$	ac	$\neg ac$	elsewhere
b	1/2	1	1
$\neg b$	1	1/4	1

$\Pi(D|C)$

$D|C$	c	$\neg c$
d	1	1/4
$\neg d$	1	1

$\Pi(E|BD)$

$E|B$	$b\neg d$	$\neg bd$	elsewhere
e	1	1/2	1
$\neg e$	3/4	1	1

So, $\Pi G_m = \{(\neg a, \emptyset, 3/4), \quad (b, ac, 1/2), (\neg b, \neg ac, 1/4), (d, \neg c, 1/4), \quad (\neg e, b\neg d, 3/4), (e, \neg bd, 1/2)\}$.

The knowledge base associated with this DAG when conditioning is based on the minimum operator is simply : $\Sigma = \{(a, 1/4), (\neg b \vee \neg a \vee \neg c, 1/2), (b \vee a \vee \neg c, 3/4), (\neg d \vee c, 3/4), (e \vee \neg b \vee d, 1/4), (\neg e \vee b \vee \neg d, 1/2)\}$.

5.3. Logical encoding of product-based possibilistic networks

In this section, we show that it is also possible to encode directed possibilistic graphs based on the product into possibilistic logic. We follow the same steps as in the previous section.

Proposition 5. *Let* π_i *be the possibility distribution associated with the triple* (a_i, u_i, α_i). *Then the joint possibility distribution computed from the directed graph* ΠG^* *is the same as the one obtained by combining the possibility distributions* π_i *by the product operator.*

It remains to find what is the possibilistic base associated with the combination of two possibility distributions by the product operator. This is summarized in the following proposition [5]:

Proposition 6. *Let* $\Sigma_1 = \{(p_i, \alpha_i) : i \in I\}$ *and* $\Sigma_2 = \{(q_j, \beta_j) : j \in J\}$. *Let* π_1 *and* π_2 *be the two possibility distributions associated with* Σ_1 *and* Σ_2, *respectively. Let* π^* *be the combination with the product of* π_1 *and* π_2. *The resulting base associated with* π^* *is*

$$\mathscr{C}^*(\Sigma_1, \Sigma_2) = \Sigma_1 \cup \Sigma_2 \cup \{(p_i \vee q_j, \alpha_i + \beta_j - \alpha_i \times \beta_j) : \\ i \in I, j \in J, p_i \vee q_j \neq \top\}.$$

NB: The combination operator $\mathscr{C}^*$ is commutative and associative, hence the combination can be applied to m elements without taking care of the ordering in which the elements are considered.

The following proposition shows that, for a given node A, the combination of two knowledge bases induced by two triples dealing with this node is simply the union of the two bases.

Proposition 7. *Let* A *be a given node (or variable) of the* ΠG, *and let* $u_1 \neq u_2$ *be two instances of* $Par(A)$, *the set of parents of* A. *Suppose* ΠG *contains* (a_1, u_1, α_1) *and* (a_2, u_2, α_2), *where* a_1 *and* a_2 *are possibly equal. Let* $\Sigma_1 = \{(\neg a_1 \vee \neg u_1, \alpha_1)\}$ *and* $\Sigma_2 = \{(\neg a_2 \vee \neg u_2, \alpha_2)\}$. *Then* $\mathscr{C}^*(\Sigma_1, \Sigma_2) = \Sigma_1 \cup \Sigma_2$.

The proof is obvious since by construction $u_1 \wedge u_2 = \bot$, hence $\neg u_1 \vee \neg u_2 \equiv \top \equiv \neg u_1 \vee \neg u_2 \vee \neg a_1 \vee \neg a_2$ hence the clause $(\neg u_1 \vee \neg u_2 \vee \neg a_1 \vee \neg a_2, \alpha_1 + \alpha_2 - \alpha_1 * \alpha_2)$ is a tautology that does not need to be added to $\mathscr{C}^*(\Sigma_1, \Sigma_2)$.

As a corollary of Proposition 7, it follows that the knowledge base resulting from combining (with the product) formulas associated to the tuples of a given node A, is simply the union of these formulas, i.e.,

$$\Sigma_A = \{(\neg a_i \vee \neg u_i, 1 - \alpha_i) : (a_i, u_i, \alpha_i) \in \Pi G\}.$$

Another corollary of the above propositions is:

Corollary 2. *From* Propositions 5–7, *it follows that the knowledge base associated with the graph* ΠG^* *is the combination by the product operator of the knowledge bases associated with each node of the graph.*

Example 3 (*continued*). : Recall that $\Pi G^* = \{(\neg a, \emptyset, 3/4), (b, ac, 1/2), (\neg b, \neg ac, 1/4), (d, \neg c, 1/4), (\neg e, b\neg d, 3/4), (e, \neg bd, 1/2)\}$. The knowledge bases associated to the different nodes of the DAG are (see Corollary 2):

- $\Sigma_A = \{(a, 1/4)\}$;
- $\Sigma_B = \{(\neg a \vee \neg b \vee \neg c, 1/2), (a \vee b \vee \neg c, 3/4)\}$;
- $\Sigma_C = \emptyset$;
- $\Sigma_D = \{(c \vee \neg d, 3/4)\}$;
- $\Sigma_E = \{(\neg b \vee d \vee e, 1/4), (b \vee \neg d \vee \neg e, 1/2)\}$.

  Now let us combine them using $\mathscr{C}^*$:
- the combination of Σ_A and Σ_B leads to:

$$\begin{aligned}\Sigma_{AB} &= \mathscr{C}^*(\Sigma_A, \Sigma_B) = \Sigma_A \cup \Sigma_B \cup \{(a \vee b \vee \neg c, 13/16)\}\\ &= \{(a, 1/4), (\neg a \vee \neg b \vee \neg c, 1/2), (a \vee b \vee \neg c, 13/16)\}\end{aligned}$$

- (the formula $(a \vee b \vee \neg c, 3/4)$ is removed since it is subsumed by $(a \vee b \vee \neg c, 13/16)$);
- since $\Sigma_C = \emptyset$, we have $\Sigma_{ABC} = \mathscr{C}^*(\Sigma_{AB}, \Sigma_C) = \Sigma_{AB}$;
- combining the result with Σ_D *gives*

$$\begin{aligned}\Sigma_{ABCD} &= \mathscr{C}^*(\Sigma_{ABC}, \Sigma_D) = \Sigma_{ABC} \cup \Sigma_D \cup \{(a \vee c \vee \neg d, 13/16)\}\\ &= \{(a, 1/4), (\neg \vee \neg b \vee \neg c, 1/2), (a \vee b \vee \neg c, 13/16), (c \vee \neg d, 3/4),\\ &\quad (a \vee c \vee \neg d, 13/16)\};\end{aligned}$$

- combining the result with Σ_E leads to

$$\begin{aligned}\Sigma_{ABCDE} &= \mathscr{C}^*(\Sigma_{ABCD}, \Sigma_E)\\ &= \Sigma_{ABCD} \cup \Sigma_E \cup \{(a \vee \neg b \vee d \vee e, 7/16), (a \vee b \vee \neg d \vee \neg e, 5/8),\\ &\quad (\neg a \vee \neg b \vee \neg c \vee d \vee e, 5/8), (a \vee b \vee \neg c \vee \neg d \vee \neg e, 29/32),\\ &\quad (b \vee c \vee \neg d \vee \neg e, 7/8), (a \vee b \vee c \vee \neg d \vee \neg e, 29/32)\}\\ &= \{(a, 1/4), (\neg a \vee \neg b \vee \neg c, 1/2), (a \vee b \vee \neg c, 13/16), (c \vee \neg d, 3/4),\\ &\quad (a \vee c \vee \neg d, 13/16), (\neg b \vee d \vee e, 1/4), (b \vee \neg d \vee \neg e, 1/2),\\ &\quad (a \vee \neg b \vee d \vee e, 7/16), (a \vee b \vee \neg d \vee \neg e, 5/8),\\ &\quad (\neg a \vee \neg b \vee \neg c \vee d \vee e, 5/8), (a \vee b \vee \neg d \vee \neg e, 29/32),\\ &\quad (b \vee c \vee \neg d \vee \neg e, 7/8)\}.\end{aligned}$$

This knowledge base contains 12 clauses while in the case of min we only have 6. This clearly illustrates that the combination with the product leads to a larger knowledge base than if we combine with the minimum, due to Proposition 6. This comes from the fact that the product is not compatible with a finite scale since it always adds formulas with intermediary levels, not already present among the levels of the original knowledge bases.

6. Recovering the initial causal data and independencies

6.1. Recovering the initial data

A natural question, when we compute a joint possibility distribution using the chain rule, is to see if we can recover the a priori and conditional possibilities one starts from. In probability theory the answer is always yes. The following proposition shows that this is also the case if the chain rule is based on the product:

Proposition 8. *Let $\Pi(a|u)$ be the conditional possibility distributions attached to the variable A in the ΠG^*. Let π^* be the joint possibility distribution obtained using the chain rule with the product, and Π^* its associated possibility measure. Then, for conditional possibility distributions we have*

$$\Pi^*(a|u) = \Pi(a|u).$$

The proof is given in Appendix A and uses the two following technical lemmas:

Lemma 4. *Let V be the set of all variables of a DAG. Let Y be a strict subset of V, and y be a fixed instantiation of Y. Let $Z = V - Y$. Then there exists z, instance of Z, such that*

$$\square_{A,a,z,u:\, A\in Z, z\models a, z\wedge y\models u}\Pi(a|u) = 1,$$

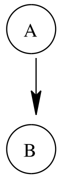
where $\square$ is either min *or product and u is the instanciation of* $\mathrm{Par}(A)$ *in* $\omega = z \wedge y$.

When $Y = \emptyset$ Lemma 4 simply means that the joint distribution associated with a possibilistic graph is normalized, independently of the definition of conditioning. More generally, it says that marginalized possibility distributions for any fixed y is also normalized.

Lemma 5. *Let $\Pi(A_i|\mathrm{Par}(A_i))$ be the conditional possibility distributions attached to variables A_i in $\Pi G_\square$, where $\square$ is either the minimum operator or the product operator. Let $\pi_\square$ be the joint possibility distribution obtained using the chain rule. Then for each value a of each variable A and each instance u of* $\mathrm{Par}(A)$, *we have* $\Pi_\square(a \wedge u) = \Pi(a|u) \,\square\, \Pi_\square(u)$.

Obviously, Lemma 5 implies Proposition 8 for $\square$ = product. However it suggests that Proposition 8 does not hold when the minimum-based conditioning is used, as illustrated by the following small example:

Example 4. Let us consider the following graph:



with $\Pi(a) = 1$; $\Pi(\neg a) = 1/4$; $\Pi(b|a) = 1/3$; $\Pi(\neg b|a) = 1$ and $\Pi(b|\neg a) = 1/3$; $\Pi(\neg b|\neg a) = 1$.

The joint possibility distribution is

$$\pi_m(ab) = 1/3; \quad \pi_m(a\neg b) = 1,$$
$$\pi_m(\neg ab) = 1/4; \quad \pi_m(\neg a\neg b) = 1/4.$$

Clearly, we have $\Pi_m(b|\neg a) = 1 \neq 1/3$ since $\pi_m(\neg ab) = \pi_m(\neg a) = 1/4$.

The reason for not recovering the original values in the example is that the conditional possibility distributions specified by the user are not coherent with the properties of (ordinal) conditional possibility. Indeed, using the definition of conditional possibility measure, recall that we always have:

$$\text{If } \Pi(p|q) \neq 1, \quad \text{then } \Pi(p|q) = \Pi(p \wedge q) < \Pi(q).$$

We see clearly, from the previous example that this constraint is violated since $\Pi(b|\neg a) = 1/3 \neq 1$ and $\Pi(b|\neg a) > \Pi(\neg a) = 1/4$. Therefore, it is not surprising if the above value 1/3 is not recovered.

This anomalous behaviour also exists in possibilistic logic, namely a possibility distribution associated with a possibilistic base may fail to preserve the original weights attached to formulas in the base. To be convinced, it is enough to consider a small example where $\Sigma = \{(a, 0.8), (a \vee b, 0.4)\}$. We can easily check that $N_{\pi_\Sigma}(a \vee b) = 0.8$. This is due to the fact that $(a \vee b, 0.4)$ is strictly subsumed by $(a, 0.8)$.

Back to the causal network, the following proposition compares conditional possibilities computed by the chain rule with the ones specified by the user.

Proposition 9. *Let $\Pi(a|u)$ be the conditional possibility distributions over the variables A in the DAG. Let π_m be the joint possibility distribution obtained using the chain rule with the minimum-based conditioning. Then, either $\pi_m(a|u) = \Pi(a|u)$ or $\pi_m(a|u) = 1$. And if $\pi_m(a|u) = 1 \neq \Pi(a|u)$, then $\Pi(a|u) > \pi_m(u)$.*

This means that the computed joint possibility distribution either preserves the initial values or moves them up to 1 (this is observed in Example 4).

A ΠG_m is said to be *coherent* if all of its initial data are recovered after applying the chain rule.

6.2. Recovering independencies

In possibility theory, the definition of independence between variables is not unique [2,8,11]. The two following definitions are the most usual ones:

Definition 7 (*Non-interactivity*). A variable A and a set of variables Y are independent in the context of a set of variables Z, if and only if for each instance (a, y, z) of (A, Y, Z) we have

$$\Pi(a, y|z) = \min(\Pi(a|z), \Pi(y|z)).$$

Definition 8 (*Causal independence*). A variable A and a set of variables Y are independent in the context of a set of variables Z, if and only if for each instance (a, y, z) of (A, Y, Z) we have

$$\Pi(a|z) = \Pi(a|yz).$$

The following proposition shows that the joint possibility distribution guarantees the "non-interactivity" independence relations from the structure of the DAG, as in a probabilistic network:

Proposition 10. *Let A be a given variable, and Y be a set of variables that contain neither a parent of A nor any of its descendants. Let π_m be the joint possibility distribution computed from a DAG ΠG_m using the min-based chain rule* (*in the sense of Definition* 6). *Then A and Y are independent in the context of* $\mathrm{Par}(A)$ *in the sense of the non-interactivity definition.*

Note that there is no need to require that the DAG be coherent in order to recover the independencies in the graph. The question of whether a joint possibility distribution can be decomposed using a stronger definition of independence is under study [2].

Note that the above proposition does not hold if causal independence is used instead of non-interactivity. Indeed, consider a simple DAG which contains two unrelated nodes A and B. Assume that a priori possibilities are: $\Pi(a) = 0.8, \Pi(\neg a) = 1,\ \Pi(b) = 0.5, \Pi(\neg b) = 1$.

It can be easily checked that $\Pi(a) = 0.8$ but $\Pi(a|b) = 1$.

6.3. Structure of knowledge bases induced by a ΠG_m

The knowledge bases constructed from $\Pi G_m s$ have a particular form.

Let $\{A_1, \ldots, A_n\}$ be the ordering of variables prescribed by the DAG, such that parents of each variable A_i are in $\{A_{i+1}, \ldots, A_n\}$ in the DAG. Then it can be checked that the knowledge base associated with the graph is of the form

$$\Sigma = \Sigma_1 \cup \cdots \cup \Sigma_n,$$

where Σ_i is a knowledge base involving variable A_i and its parents only. It can be checked that for each Σ_i we have $(\neg a_i \vee \neg u_i, \alpha_i) \in \Sigma_i$ iff $\Pi(a_i|u_i) = 1 - \alpha_i \in \Pi G_m$. Especially, each clause in Σ_i is a disjunction of an instance of A_i and instances of all of its parents. We call such a set of clauses a clausal completion of the variable A_i in the context of its parents.

As a preliminary step to the following section let us show that a clausal possibilistic knowledge base where a variable appears in all clauses can be put in a form similar to Σ_i. To this aim, let us define the notion of complete extension of a knowledge base with respect to a variable.

Definition 9. Let Σ be a possibilistic knowledge base in a clausal form, where all clauses involve an instance of a variable A. Let X be the set of other variables appearing in the clauses of Σ. A clausal completion of Σ with respect to variable A denoted by $E(\Sigma)$, is the set of all clauses of the form $(a \vee \neg x, \alpha)$ where a is an instance of A, x is an instance of all variables in X, and $\alpha = \max\{\alpha_i : (a \vee p_i, \alpha_i) \in \Sigma, x \models \neg p_i\}$, with $\max\{\emptyset\} = 0$.

Note that x is a conjunction of literals while p_i is a disjunction thereof. The idea is to change each disjunction into a set of maximal disjunctions involving all variables.

Example 5. Let $\Sigma = \{(a \vee b, 0.5), (a \vee c, 0.7)\}$. Let us find the clausal completion of A. Then we can check that $E(\Sigma) = \{(a \vee b \vee \neg c, 0.5), (a \vee b \vee c, 0.5), (a \vee c \vee \neg b, 0.7), (a \vee c \vee b, 0.7)\}$, which is semantically equivalent to Σ and to $E(\Sigma) = \{(a \vee b \vee \neg c, 0.5), (a \vee c \vee \neg b, 0.7), (a \vee c \vee b, 0.7)\}$.

Now we can prove:

Proposition 11. *The two bases Σ and $E(\Sigma)$ are equivalent.*

The notion of clausal completion is instrumental for turning a possibilistic knowledge base into a possibilistic network of the ΠG_m kind, which is the topic of the following section.

7. Encoding possibilistic bases into a ΠG_m

In this section, we present the transformation of possibilistic knowledge bases into directed possibilistic graphs ΠG_m. One way to do it is to use possibility distributions as intermediary step. Indeed, a knowledge base leads to a possibility distribution, from which it is possible to build a ΠG_m (see Section 3.4). This would apply as well to ΠG^*. However, this way is computationally expensive. So, we want to find the ΠG_m directly from the knowledge base.

The encoding of a possibilistic knowledge into a ΠG_m is less straightforward than the previous transformation. Indeed, we cannot directly view each formula as a triple and then build the graph, but we need some pre-processing steps, because as seen above the possibilistic base constructed from a ΠG_m has a special form. The constructed possibilistic graph ΠG_m should be such that:

- the joint possibility distribution computed from the ΠG_m using the minimum operator should be the same as the one computed from the knowledge base and
- the ΠG_m constructed from the knowledge bases should be coherent.

As stated in the last sub-section, the knowledge base associated with a possibilistic net has a special form. Therefore, we need to put the possibilistic knowledge base in this special form.

To reach this aim, the construction of the causal network is obtained in three steps: the first step simply consists in putting the knowledge base into a clausal form and in removing tautologies. The second step consists in constructing the graph associated to an arbitrary ordering of variables, and the last step computes the conditional possibilities associated to the constructed graph.

7.1. Putting bases in a clausal form and removing tautologies

In this step, a base Σ is rewritten into a semantically equivalent base Σ'. Getting Σ' consists in putting the knowledge base into a clausal form and in removing tautologies. The following proposition shows how to put the base in a clausal form first [12]:

Proposition 12. *Let $(p, \alpha) \in \Sigma$. Let $\{c_1, \ldots, c_n\}$ be the set of clauses encoding p in classical logic. Let Σ' be a new knowledge base obtained from Σ by replacing (p, α) by $\{(c_1, \alpha), \ldots, (c_n, \alpha)\}$. Then the two knowledge bases Σ and Σ' are semantically equivalent.*

Then removing tautologies still leads to an equivalent possibilistic base (see Lemma 2). The removing of tautologies is an important point since this will avoid having links in the graph which do not make sense. For example, the tautological formula $(\neg a \vee \neg y \vee a, 1)$ might induce a spurrious link between A and Y.

7.2. Constructing the graph

The second step consists in constructing the graph, namely the determination of the vertices (variables) of the graph and the parents of each vertex. The set of variables is simply the set of propositional symbols which appear in the knowledge base. Moreover, since possibilistic logic is based on propositional logic, then all variables are binary. To construct the graph, we first rank the variables, according to an arbitrary numbering $\{A_1, A_2, \ldots, A_n\}$ of the variables. This ranking intends to mean that parents of each variable A_i can only be in $\{A_{i+1}, \ldots, A_n\}$ (but they may fail to exist).

We first give some intuitive examples before presenting the technical construction of the graph.

Example 6. Let $\Sigma = \{(t, 0.6), (t \vee v, 0.4)\}$. From this knowledge base one may think that the variable T depends on the variable V. However, we can easily check that Σ is equivalent to the following one: $\Sigma' = \{(t, 0.6)\}$, where clearly, V has no influence on T. The formula $(t \vee v, 0.4)$ is simply subsumed by $\Sigma - \{(t \vee v, 0.4)\}$.

Subsumed beliefs are not the only ones which may induce fictitious dependencies:

Example 7. Let $\Sigma = \{(a, 0.5), (\neg a \vee b, 0.5)\}$. In this base, neither $(a, 0.5)$ nor $(\neg a \vee b, 0.5)$ is subsumed, and one can think that there is a relationship between the variables A and B. However, we can easily check that this base is equivalent to $\Sigma' = \{(a, 0.5), (b, 0.5)\}$, where clearly A and B are unrelated.

So we are led to state:

Proposition 13. *Let A be a variable, and $(a \vee p, \alpha)$ be a clause of Σ containing the instance a of A. If $\Sigma \vdash (p, \alpha)$, then the base Σ and the knowledge base Σ' obtained from Σ by replacing $(a \vee p, \alpha)$ by (p, α) are equivalent.*

Intuitively, one could say that two variables are related if there is a clause containing an instance of these two variables, and they are unrelated otherwise. Example 8 shows that two variables can be related even if there is no clause in the base containing an instance of each variable:

Example 8. Let $\Sigma = \{(c \vee a, 0.5), (\neg c \vee b, 0.5)\}$. In this base, if we could establish the dependence between variables only if there exists a clause containing an instance of each of them, then clearly A and B would be unrelated. However, we can check that $\Sigma \vdash (a \vee b, 0.5)$.

All these examples recall that logic is syntax independent, and graphs are more suitable to exhibit independence and relevance relations.

Recall that our aim is to get a coherent ΠG_m (where local conditional possibility distributions can be retrieved by conditioning their min-based combination). The following example shows that it is enough to look only for clauses of the base containing instances of the variable A:

Example 9. Let $\Sigma = \{(a \vee b, 0.6), (c \vee b, 0.5), (c \vee a, 0.5)\}$. Assume that the parents of C are A and B. Clearly, if we compute the conditional possibilities $\Pi(C|AB)$ only from clauses containing C, namely $\Sigma_C = \{(c \vee b, 0.5), (c \vee a, 0.5)\}$, then it is not guaranteed to get a coherent ΠG_m. Indeed, in this example, if the computation of $\Pi(\neg c|\neg a \neg b)$ is simply based on Σ_C, we get $\Pi(\neg c|\neg a \neg b) = 0.5$ (since $\{(c \vee b, 0.5), (c \vee a, 0.5)\} \vdash (c \vee a \vee b, 0.5)$) but we can check that after computing the joint possibility distribution $\pi_\Sigma : \Pi_\Sigma(\neg c|\neg a \neg b) = 1$.

This is due to the fact that we have both $\Sigma \vdash (c \vee a \vee b, 0.5)$ and $\Sigma \vdash (a \vee b, 0.6)$, hence: $\Pi_\Sigma(\neg c \neg a \neg b) = \Pi_\Sigma(\neg a \neg b)$.

Indeed, $(c \vee b, 0.5)$ and $(c \vee a, 0.5)$ are not subsumed but $(c \vee a \vee b, 0.6)$ is subsumed due to the clause $(a \vee b, 0.6)$ in Σ.

Based on these intuitive examples, we are now able to give the algorithm which transforms a base into a DAG, given an arbitrary numbering of the variables $\{A_1, \ldots, A_n\}$. It is composed of four iteratively repeated steps.

For each variable A_i, determine parents of A_i, compute the corresponding clausal completion, remove redundant data and lastly produce a local base Σ_i that will correspond to the part of the graph relating A_i to its parents. More precisely:

For $i = 1, \ldots, n$ do
Begin
/* *Determination of the local base for* A_i */
1. Let $(a_i \vee p, \alpha)$ be a clause of Σ s.t. a_i is an instance of A_i, and p is only built from $\{A_{i+1}, \ldots, A_n\}$
1.1. If $(a_i \vee p, \alpha)$ is subsumed, then remove it from Σ (due to Lemma 1)
1.2. If $\Sigma \vdash (p, \alpha)$, then replace $(a_i \vee p, \alpha)$ by (p, α) (due to Proposition 12)
2. Let K_i be the set of clauses $(a_i \vee p, \alpha)$ in Σ s.t. p is only built from $\{A_{i+1}, \ldots, A_n\}$
3. The parents of the variable A_i are :
$\text{Par}(A_i) = \{A_j : \exists c \in K_i$ such that $c$contains an instance of $A_j\}$
/* *Compute the clausal completion of* K_i */
4. Replace in Σ, K_i by its clausal completion $E(K_i)$ (due to Proposition 13)
/* *Remove incoherent data* */
5. For each $(a_i \vee p, \alpha)$ of Σ (where p is built from $\{A_{i+1}, \ldots, A_n\}$) such that $\Sigma \vdash (p, \alpha)$ replace $(a_i \vee p, \alpha)$ by (p, α)
/* *Produce* Σ_i */
6. Let Σ_i be the set of clauses $(a_i \vee p, \alpha)$ in Σ s.t. p is only built from $\{A_{i+1}, \ldots, A_n\}$
End

The algorithm starts by determining the parents of each variable (steps 1–3), and then proceeds (steps 4–6) in rewriting the knowledge base such that:

(i) it immediately gives the conditional possibility distributions attached to each variable, and

(ii) ensures the recovery of original values when using the chain rule for computing the joint possibility distributions.

Indeed, once $E(K_i)$ is computed, in order to evaluate $\Pi(a_i|u_i)$ then

- either $(\neg a_i \vee \neg u_i, \alpha) \notin E(K_i)$ then $\Pi(a_i|u_i) = 1$, or
- $(\neg a_i \vee \neg u_i, \alpha) \in E(K_i)$, then,
  - if $\Sigma \vdash (\neg u_i, \alpha)$, then $\Pi(a_i|u_i) = 1$ (since $\Pi(a_i \wedge u_i) = \Pi(a_i)$. Hence $(\neg a_i \vee \neg p, \alpha)$ can be removed from Σ),
  - otherwise $\Pi(a_i|u_i) = 1 - \alpha$.

Step 5 is not redundant with 1.1., since after computing the extension by step 4, it may happen that some $(a_i \vee p, \alpha)$ will belong to Σ and $\Sigma \vdash (p, \alpha)$.

The result of the algorithm is a partition $\{\Sigma_1, \ldots, \Sigma_n\}$ such that $\Sigma_1 \cup \cdots \cup \Sigma_n$ is semantically equivalent to Σ. Clearly, for $i > 1$, Σ_i does not contain any variable from $\{A_1, \ldots, A_{i-1}\}$. Moreover, Σ_{A_i} can be empty. In this case A_i has no parents: it corresponds to the roots of the graph, and the a priori possibility degrees on the domain of A_i are equal to 1. The subbases Σ_{A_i}'s give a direct computation of conditional possibility degrees as explained later. A graph associated to Σ is such that its vertices are the variables in Σ, and a link is drawn from A_j to A_i iff $A_j \in \mathrm{Par}(A_i)$, where $\mathrm{Par}(A_i)$ is given by step 3 in the algorithm. This graph is of course a DAG due to the a prori ordering of the variables.

Example 10. Let us consider the following base: $\Sigma = \{(a \vee b, 0.7), (\neg a \vee c \vee \neg d, 0.7), (a \vee c \vee d, 0.9), (b \vee c, 0.8), (\neg b \vee e, 0.2), (\neg d \vee f, 0.5)\}$. Σ contains six variables arbitrarily numbered in the following way: $A_1 = A, A_2 = B, A_3 = C, A_4 = D, A_5 = E$ and $A_6 = F$. The previous algorithm is applied to determine both the graph and the conditional possibilities associated with it.

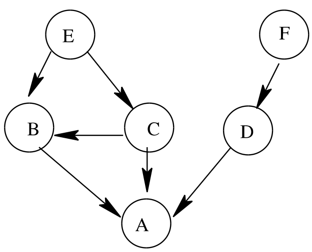
- Treatment of node A.
  1. For the node A, the formulas $(a \vee b, 0.7), (\neg a \vee c \vee \neg d, 0.7)$ and $(a \vee c \vee d, 0.9)$ are considered. They are not subsumed and b, $c \vee \neg d$ and $c \vee d$ are not inferred.
  2. $K_A = \{(a \vee b, 0.7), (\neg a \vee c \vee \neg d, 0.7), (a \vee c \vee d, 0.9)\}$.
  3. $\mathrm{Par}(A) = \{B, C, D\}$.
  4. $E(K_A) = \{(a \vee b \vee c \vee \neg d, 0.7), (a \vee b \vee \neg c \vee d, 0.7), (a \vee b \vee \neg c \vee \neg d, 0.7), (\neg a \vee b \vee c \vee \neg d, 0.7), (\neg a \vee \neg b \vee c \vee \neg d, 0.7)(a \vee b \vee c \vee d, 0.9), (a \vee \neg b \vee c \vee d, 0.9)\}$.
  5. $\Sigma = \{(a \vee b \vee \neg c \vee d, 0.7), (a \vee b \vee \neg c \vee \neg d, 0.7), (\neg a \vee \neg b \vee c \vee \neg d, 0.7), (a \vee b \vee c \vee d, 0.9), (a \vee \neg b \vee c \vee d, 0.9), (b \vee c \vee \neg d, 0.7), (b \vee c, 0.8), (\neg b \vee e, 0.2), (\neg d \vee f, 0.5)\}$. (The formulas $(a \vee c \vee \neg d, 0.7)$ and $(\neg a \vee b \vee c \vee \neg d, 0.7)$ are entailed by $(b \vee c \vee \neg d, 0.7)$ which is entailed by Σ.)

6. $\Sigma_A = \{(a \vee b \vee \neg c \vee d, 0.7), \quad (a \vee b \vee \neg c \vee \neg d, 0.7), (\neg a \vee \neg b \vee c \vee \neg d, 0.7), (a \vee b \vee c \vee d, 0.9), (a \vee \neg b \vee c \vee d, 0.9)\}$.

- Treatment of node B.
  1. For the node B, the formulas $(b \vee c \vee \neg d, 0.7), (b \vee c, 0.8), (\neg b \vee e, 0.2)$ are considered. $(b \vee c \vee \neg d, 0.7)$ is subsumed by $(b \vee c, 0.8)$. The formulas $(b \vee c, 0.8)$, $(\neg b \vee e, 0.2)$ are not subsumed. c and e are not inferred.
  2. $K_B = \{(b \vee c, 0.8), (\neg b \vee e, 0.2)\}$.
  9. $\mathrm{Par}(B) = \{C, E\}$.
  3. $E(K_B) = \{(b \vee c \vee e, 0.8), \ (b \vee c \vee \neg e, 0.8), \ (\neg b \vee c \vee e, 0.2), \ (\neg b \vee \neg c \vee e, 0.2)\}$.
  4. $\Sigma = \{(a \vee b \vee \neg c \vee d, 0.7), \ (a \vee b \vee \neg c \vee \neg d, 0.7), \ (\neg a \vee \neg b \vee c \vee \neg d, 0.7), \ (a \vee b \vee c \vee d, 0.9), (a \vee \neg b \vee c \vee d, 0.9), (b \vee c \vee e, 0.8), (b \vee c \vee \neg e, 0.8), (\neg b \vee \neg c \vee e, 0.2), (c \vee e, 0.2), (\neg d \vee f, 0.5)\}$.
  6. $\Sigma_B = \{(b \vee c \vee e, 0.8), (b \vee c \vee \neg e, 0.8), (\neg b \vee \neg c \vee e, 0.2)\}$.
- The same treatment is repeated for the nodes C, D, E and F.
  The final base is

$$\Sigma = \{(a \vee b \vee \neg c \vee d, 0.7), (a \vee b \vee \neg c \vee \neg d, 0.7), (\neg a \vee \neg b \vee c \vee \neg d, 0.7), (a \vee b \vee c \vee d, 0.9), (a \vee \neg b \vee c \vee d, 0.9), (b \vee c \vee e, 0.8), (b \vee c \vee \neg e, 0.8), (\neg b \vee \neg c \vee e, 0.2), (c \vee e, 0.2), (\neg d \vee f, 0.5)\}.$$

The final graph is



7.3. Determining the local conditional distributions

Once the graph is constructed we need to compute the conditional possibilities attached to nodes. The computation of $\Pi(A_i|\mathrm{Par}(A_i))$ is immediately obtained from Σ_i: Let $\{\Sigma_1, \ldots, \Sigma_n\}$ be the result of step 6 of the previous algorithm. Let A_i be a variable and $\mathrm{Par}(A) = \{B_1, \ldots, B_m\}$ be the set of its parents. Let a be an instance of A_i and $u = b_1 \wedge \cdots \wedge b_m$ be an instance of $\mathrm{Par}(A_i)$. Let:

$$\Pi(a|u) = \begin{cases} 1-\alpha_i & \text{if } (\neg a \vee \neg u, \alpha_i) \in \Sigma_i, \\ 1 & \text{otherwise.} \end{cases}$$

Then we have:

Proposition 14. *The possibility distribution associated to Σ and the possibility distribution obtained from the graph using the minimum operator are equal, namely*: $\pi_\Sigma = \pi_m$.

The proof is immediate using Proposition 3.

Example 11. The partition of the initial base is:

$$\Sigma_A = \{(a \vee b \vee \neg c \vee d, 0.7), (a \vee b \vee \neg c \vee \neg d, 0.7), (\neg a \vee \neg b \vee c \vee \neg d, 0.7), (a \vee b \vee c \vee d, 0.9), (a \vee \neg b \vee c \vee d, 0.9)\},$$
$$\Sigma_B = \{(b \vee c \vee e, 0.8), (b \vee c \vee \neg e, 0.8), (\neg b \vee \neg c \vee e, 0.2)\},$$
$$\Sigma_C = \{(c \vee e, 0.2)\},$$
$$\Sigma_D = \{(\neg d \vee f, 0.5)\}, \Sigma_E = \Sigma_F = \emptyset.$$

Then, we find the conditional possibility distributions given by the following table:

$\pi(A|BCD)$

$A\|BCD$	$\neg b \neg c \neg d$	$\neg b \neg c d$	$\neg b c \neg d$	$\neg b c d$
a	1	1	1	1
$\neg a$	0.1	1	0.3	0.3

$A\|BCD$	$b \neg c \neg d$	$b \neg c d$	$b c \neg d$	bcd
a	1	0.3	1	1
$\neg a$	0.1	1	1	1

$\pi(B|CE)$

$B\|CE$	$\neg c \neg e$	$\neg c e$	$c \neg e$	ce
b	1	1	0.8	1
$\neg b$	0.2	0.2	1	1

$\pi(C|E)$

$C\|E$	e	$\neg e$
c	1	1
$\neg c$	1	0.8

$\pi(D|F)$

$D\|F$	f	$\neg f$
d	1	0.5
$\neg d$	1	1

$\pi(E)$

e	1
$\neg e$	1

$\pi(F)$

f	1
$\neg f$	1

Proposition 15. *Let $\Pi(A_i|\mathrm{Par}(A_i))$ be the min-based conditional possibility distributions attached to variables A_i in a causal network ΠG_m obtained from a possibilistic knowledge base Σ. Let π_m be the joint possibility distribution obtained by the chain rule with minimum from the causal network. Then the constructed ΠG_m is coherent, namely*: $\Pi_m(A_i|\mathrm{Par}(A_i)) = \Pi(A_i|\mathrm{Par}(A_i))$.

The proof follows immediately from the algorithm. Suppose $(\neg a \vee \neg u, \alpha) \in \Sigma$. In step 5, when $\Sigma \vdash (\neg u, \alpha)$, then $(\neg a \vee \neg u, \alpha)$ is removed from Σ. Therefore, if $(\neg a \vee \neg u, \alpha) \in \Sigma$ (hence $\Pi(a|u) = 1 - \alpha$ in the net), then $\Sigma \nvdash (\neg u, \alpha)$ hence $N(\neg a \vee \neg u) > 0$ and $N(\neg u) = 0$, where N is the necessity measure induced from π_Σ (which is equal to π_m). Hence $\Pi_m(a \wedge u) < 1$, and $\Pi_m(u) = 1$, therefore using the definition of conditioning we have: $\pi_m(a|u) = 1 - \alpha = \Pi(a|u)$.

8. Conclusion

This paper has bridged the gap between possibilistic logic and directed possibilistic graphs. In the whole paper, a possibilistic logic formula is viewed as a piece of uncertain knowledge whose certainty level is lower-bounded in terms of a necessity measure. There exists another understanding of possibilistic formulas as expressing preferences under the form of goals with their priority levels. Thus the graph associated with a possibilistic base can be also used for figuring out how preferences interact.

We have shown that directed possibilistic graphs can be encoded into possibilistic logic, for the two possible definitions of conditioning. It allows the expert to express his knowledge using "causality" relations between variables, and then the possibilistic logic machinery can be applied after the computation of the corresponding possibilistic logic base. A future work would be the study of the complexity of the conversion of a directed causal network based on the product into possibilistic logic (with the min, the conversion is of linear complexity) and the comparison of the cost of the inference using the network directly with the one using the corresponding possibilistic knowledge base.

The converse translation from a possibilistic logic base to a min-based possibilistic graph has also been provided, given a prescribed ranking of the variables. In this case, once the base is put under a canonical form, each possibilistic logic formula is translated into one conditional possibility degree in the possibilistic graph. An open problem, which is the counterpart of the learning problem for Bayesian nets, is to optimize the total ordering of the variables so as to simplify the structure of the possibilistic network induced by a given possibilistic knowledge base. The syntactic translation of a possibilistic logic base to a product-based possibilistic graph is achieved in [1].

Apart from its conceptual appeal, the established equivalence between possibilistic logic and min-based possibilistic DAGs may have some impact on computational issues. Namely, local propagation algorithms in possibilistic DAGs can be envisaged and a systematic comparison between the efficiency of these algorithms and automated theorem proving methods in possibilistic logic [25] is worthwhile investigating.

Appendix A. Proofs of technical propositions

Lemma 1. *Let (p, α) be a subsumed belief of Σ. Then Σ and $\Sigma' = \Sigma - \{(p, \alpha)\}$ are equivalent, namely $\pi_\Sigma = \pi_{\Sigma'}$.*

Proof. Indeed, (p, α) is a subsumed formula in $\Sigma \Longleftrightarrow \Sigma - \{(p, \alpha)\} \vdash (p, \alpha) \Longleftrightarrow \pi_{\Sigma-\{(p,\alpha)\}} \leqslant \pi_{\{(p,\alpha)\}}$ (soundness and completeness of possibilistic logic). Hence, $\pi_\Sigma = \min(\pi_{\Sigma-\{(p,\alpha)\}}, \pi_{\{(p,\alpha)\}}) = \pi_{\Sigma-\{(p,\alpha)\}}$. □

Proposition 2. *Let π be a non-dogmatic possibility distribution, let p and q two non-mutually exclusive propositions. Then if $\pi_1 = \pi(\cdot|p)$ and $\pi_2 = \pi_1(\cdot|q)$, $\pi_3 = \pi(\cdot|q)$ and $\pi_4 = \pi_3(\cdot|p)$. Then, $\pi_2 = \pi_4$.*

Proof. Let us prove that $\pi_2(\omega) = \pi(\omega|p \wedge q)$.

$$\pi_2(\omega) = \begin{cases} 1 & \text{if } \pi(\omega|p) = \Pi(q|p) \text{ and } \omega \models q, \\ \pi(\omega|p) & \text{if } \pi(\omega|p) < \Pi(q|p) \text{ and } \omega \models q, \\ 0 & \text{if } \omega \not\models q. \end{cases}$$

But, if $\pi(\omega|p) < \Pi(q|p)$ and $\omega \models p$, then $\pi(\omega|p) = \pi(\omega)$ and $\pi(\omega) < \Pi(p)$ (since $\pi(\omega) < 1$). If $\omega \not\models p$ $\pi_2(\omega) = 0$ in any case. So

$$\pi_2(\omega) = \begin{cases} 1 & \text{if } \pi(\omega|p) = \Pi(q|p) \text{ and } \omega \models p \wedge q, \\ \pi(\omega) < 1 & \text{if } \pi(\omega) < \Pi(q|p) \text{ and } \pi(\omega) < \Pi(p) \\ & \quad \text{and } \omega \models p \wedge q, \\ 0 & \text{if } \omega \models \neg q \vee \neg p. \end{cases}$$

Note that $\pi(\omega) < \Pi(p)$ and $\pi(\omega) < \Pi(q|p)$, is equivalent to $\pi(\omega) < \min(\Pi\ (q|p), \Pi(p)) = \Pi(p \wedge q)$.

Hence, $\pi_2(\omega) = \pi(\omega) < 1$ if and only if $\pi(\omega) < \Pi(p \wedge q)$ for $\omega \models p \wedge q$.

So $\pi_2(\omega) = 1$ if and only if $\pi(\omega) = \min(\Pi(q|p), \Pi(p)) = \Pi(p \wedge q)$ for $\omega \models p \wedge q$. We conclude that $\pi_2(\omega) = \pi(\omega|p \wedge q)$. Hence $\pi_2(\omega) = \pi_4(\omega)$ by symmetry. □

Proposition 3. *For every normalized possibility distribution π, we have $\Pi(\neg p \vee q) \geqslant \Pi(q|_m p) \geqslant \Pi(p \wedge q)$. Moreover,*

- $\Pi(q|_m p) = \Pi(p \wedge q)$ if and only if $\Pi(p \wedge q) < \Pi(p \wedge \neg q)$ or $\Pi(p \wedge q) = 1$,
- $\Pi(q|_m p) = \Pi(\neg p \vee q)$ if and only if $\Pi(p \wedge q) \geqslant \Pi(p \wedge \neg q)$ or $\Pi(p \wedge \neg q) > \Pi(p \wedge q) \geqslant \Pi(\neg p)$.

Proof. $\Pi(q|_m p) \geqslant \Pi(p \wedge q)$ is obvious since $\Pi(q|_m p)$ is either equal to 1, or to $\Pi(p \wedge q)$. Now, if $\Pi(q|_m p) = 1$, this means that $\Pi(q \wedge p) = \Pi(p) \geqslant \Pi(p \wedge \neg q)$. Hence since $\Pi(\neg p \vee q) \geqslant \Pi(p \wedge q)$, it follows that $\Pi(\neg p \vee q) \geqslant \Pi(p) \geqslant \Pi(p \wedge \neg q)$. Hence (by normalisation of π, $\max(\Pi(\neg p \vee q), \Pi(p \wedge q)) = 1$), $\Pi(\neg p \vee q) = 1$. If $\Pi(q|_m p) \neq 1$, this means that $\Pi(q|_m p) = \Pi(p \wedge q)$ and the equality $\Pi(\neg p \vee q) \geqslant \Pi(p \wedge q)$ obviously holds.

The second item is obvious by definition. As for the third, either $\Pi(q|_m p) = 1$; clearly, this is true only when $\Pi(q \wedge p) \geqslant \Pi(\neg q \wedge p)$. Or $\Pi(q|_m p) = \Pi(q \vee \neg p) < 1$, and then $\Pi(q|_m p) = \Pi(q \wedge p) = \Pi(q \vee \neg p)$, which is equivalent to $\Pi(p \wedge \neg q) = 1 > \Pi(p \wedge q) \geqslant \Pi(\neg p)$. $\square$

Lemma 4. *Let V be the set of all variables of a DAG. Let Y be a strict subset of V, and y be a fixed instantiation of Y. Let $Z = V - Y$. Then there exists z, instance of Z, such that*

$$\square_{A,a,z,u:\, A \in Z, z \models a, z \wedge y \models u} \Pi(a|u) = 1,$$

where $\square$ is either min or product and u is the instanciation of Par(A) *in $\omega = z \wedge y$.*

Proof. Note first that from the normalisation conditions on conditional possibility distributions we always have, for any fixed values a of A and u of $\mathrm{Par}(A)\max(\Pi(a|u), \Pi(\neg a|u)) = 1$.

To prove the result, it is enough to exhibit a particular instance z of $V - Y$ such that $\Pi(a|u) = 1$ for any a, u such that $z \models a$ and $z \wedge y \models u$.

The particular instance z is obtained in a constructive way by the following algorithm:

Let $Z = V - Y$, $X = Y$, $x = y$.
While $Z \neq \emptyset$ do
Begin
  Select A in Z such that A has no parents in Z.
  Select a an instance of A such that $\Pi(a|u) = 1$ where u is the instance of $\mathrm{Par}(A)$ prescribed by x that is, $x \models u$ (since $\mathrm{Par}(A) \cap Z = \emptyset$).
  $x = x \wedge a$;
  $Z = Z - \{A\}$; $X = X \cup \{A\}$
End.

The algorithm first selects a variable A which has no parent in Z. Such a variable always exists, otherwise there is a cycle in the possibilistic network. Then we choose an instance a of A such that $\Pi(a|u) = 1$ and $x \models u$ where u is the instance of $\mathrm{Par}(A)$ prescribed by x. x is a global instantiation of the set of variables X. At the beginning it coincides with y (which is fixed).

When the value a of A has been chosen, we add A to X and we remove A from Z.

We repeat the previous step until we instantiate all the variables of Z, thus building an interpretation $\omega = z \wedge y$ which completes y.

At the end, we have for each value a of A such that $z \models a$:

$$\Pi(a|u) = 1 \quad \text{and} \quad z \wedge y \models u.$$

Therefore

$$\max_{z \in D_Z} \square_{A \in Z, z \models a, z \wedge y \models u} \Pi(a|u) = 1. \qquad \square$$

Lemma 5. *Let $\Pi(A_i|\mathrm{Par}(A_i))$ be the conditional possibility distributions attached to variables A_i in $\Pi G_\square$, where $\square$ is either the minimum operator or the product operator. Let $\pi_\square$ be the joint possibility distribution obtained using the chain rule. Then for each value a of each variable A and each instance u of $\mathrm{Par}(A)$, we have $\Pi_\square(a \wedge u) = \Pi(a|u) \square \Pi_\square(u)$.*

Proof. Let
- $D =$ the set of all descendants of A, and,
- $Z = V - (D \cup \{A\} \cup \mathrm{Par}(A))$ the set of variables not related to A.

In the following, z is an instance of Z and d an instance of D, and u an instance of $\mathrm{Par}(A)$. Note that for each $B \in V - (D \cup \{A\})$, $\mathrm{Par}(B) \cap (D \cup \{A\}) = \emptyset$. Indeed, if there is $C \in \mathrm{Par}(B) \cap (D \cup \{A\})$, then this simply means that the variable B is also a descendant of A and hence contradicts the fact that $B \in V - D \cup \{A\}$.

Then by definition, decomposing interpretations ω as $a \wedge u \wedge z \wedge d$, and fixing a and u, and, for any variable B denoting P_b the instance of $\mathrm{Par}(B)$ induced by ω

$$\begin{aligned}
\Pi_\square(a \wedge u) &= \max_{z,d}\{\pi_\square(a \wedge u \wedge z \wedge d)\} \\
&= \max_{z,d}\{\Pi(a|u) \square \{\Pi(b|P_b) : B \in \mathrm{Par}(A), \mathrm{Par}(A) = u\} \\
&\quad \square \{\Pi(c|P_c) : C \in Z, \mathrm{Par}(A) = u\} \\
&\quad \square \{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, A = a\} \\
&= \Pi(a|u) \square \max_{z}\{\square\{\Pi(b|P_b) : B \in \mathrm{Par}(A)\} \square \{\Pi(c|P_c) : C \in Z\} \\
&\quad \square \max_{d}\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, A = a\}
\end{aligned}$$

(since $\mathrm{Par}(B) \subseteq \mathrm{Par}(A) \cup Z, \mathrm{Par}(C) \subseteq \mathrm{Par}(A) \cup Z$ for variables $B \in \mathrm{Par}(A)$, $C \in Z$ in the above equations. However $\mathrm{Par}(R)$ can be any subset of V if $R \in D$.)

Note that $\max_d\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, A = a\} = 1$ due to Lemma 4 (letting $y = a \wedge u \wedge z$ and varying d). Hence

$$\Pi_\square(a \wedge u) = \Pi(a|u) \square \max_{z}\{\square\{\Pi(b|P_b) : B \in \mathrm{Par}(A)\} \square \{\Pi(c|P_c) : C \in Z\}.$$

Using Lemma 4 again, we also have changing a into $\neg a$:

$$\max_d\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, A = \neg a\} = 1.$$

Moreover, since

$$\max(\Pi(a|u), \Pi(\neg a|u)) = 1.$$

Let $\alpha_Z = \max_z\{\Box\{\Pi(b|P_b) : B \in \mathrm{Par}(A)\}\Box\{\Pi(c|P_c) : C \in Z\}$, We then have

$$\begin{aligned}
&\Pi_\Box(a \wedge u)\\
&\quad = \Pi(a|u) \Box \max\{\Pi(a|u) \Box \alpha_Z, \Pi(\neg a|u) \Box \alpha_Z\}\\
&\quad = \Pi(a|u) \Box\\
&\quad \max\{\Pi(a|u) \Box \alpha_Z \Box \max_d\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, A = a\},\\
&\quad \Pi(\neg a|u) \Box \alpha_Z \Box \max_d\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, A = \neg a\}\}\\
&\quad = \Pi(a|u) \Box \max\{\Pi_\Box(a \wedge u), \Pi_\Box(\neg a \wedge u)\}
\end{aligned}$$

(since $\Pi_\Box(a \wedge u)$ is developed with respect to parents of$A$; of *parents of A*, and then to the remainder of variables)

$$= \Pi(a|u) \Box \Pi_\Box(u). \qquad \Box$$

Proposition 8. *Let $\Pi(a|u)$ be the conditional possibility distributions over the variables A in the ΠG^*. Let π^* be the joint possibility distribution obtained using the chaining rule with the product, and Π^* its associated possibility measure. Then, for any conditional possibility distribution, we have*

$$\Pi^*(a|u) = \Pi(a|u).$$

Proof. The proof is immediate. Indeed, by definition,

$$\Pi^*(a|u) = \frac{\Pi^*(a \wedge u)}{\Pi^*(u)}.$$

Moreover, from Lemma 5 we have $\Pi^*(a \wedge u) = \Pi(a|u) * \Pi^*(u)$, therefore $\Pi^*(a|u) = \Pi(a|u)$. $\Box$

Proposition 9. *Let $\Pi(a|u)$ be the conditional possibility distributions over the variables A in the DAG. Let π_m be the joint possibility distribution obtained using the chain rule with the minimum-based conditioning. Then either $\pi_m(a|u) = \Pi(a|u)$ or $\pi_m(a|u) = 1$. And if $\pi_m(a|u) = 1 \neq \Pi(a|u)$, then $\Pi(a|u) > \pi_m(u)$.*

Proof. From Lemma 5, we have

$$\Pi_m(a \wedge u) = \min(\Pi(a|u), \Pi_m(u)).$$

Then, we distinguish two cases:

- $\Pi(a|u) = 1$, then $\Pi_m(a \wedge u) = \Pi_m(u)$, hence using the definition of conditioning, we get

$$\Pi_m(a|u) = 1.$$

- $\Pi(a|u) = \alpha < 1$, then $\Pi_m(a \wedge u) = \min(\alpha, \Pi_m(u))$.

If $\alpha \leqslant \Pi_m(u)$, then $\Pi_m(a|u) = \alpha$. If $\alpha > \Pi_m(u)$, then $\Pi_m(a \wedge u) = \Pi_m(u)$, hence using the definition of conditioning, we get

$$\Pi_m(a|u) = 1.$$

And it is clear that the only case where the conditional value is not recovered is when $\alpha > \Pi_m(u)$.

Proposition 10. *Let A be a given variable, and Y be a set of variables that contain neither a parent of A nor any of its descendants. Let π_m be the joint possibility distribution computed from a DAG ΠG_m using the min-based chain rule* (*in the sense of Definition* 5). *Then A and Y are independent in the context of* Par(A) *in the sense of the non-interactivity definition.*

Proof. Let

$$D = \text{the set of all descendants of } A \text{ and}$$
$$Z = V - (D \cup \{A\} \cup Y \cup \mathrm{Par}(A)) \text{ the rest of remaining variables.}$$

We first show that $\Pi_m(a \wedge y \wedge u) = \min(\Pi(a|u), \Pi_m(y \wedge u))$.

Using similar computation than in Lemma 5, we have

$$\begin{aligned}\Pi_m(a \wedge y \wedge u) &= \max_{z,d}\{\pi_m(a \wedge y \wedge u \wedge z \wedge d)\} \\ &= \max_{z}\{\min\{\Pi(a|u), \\ &\quad \min\{\Pi(e|P_e), E \in Y, Y = y, Z = z\}, \\ &\quad \min\{\Pi(b|P_b) : B \in \mathrm{Par}(A), \mathrm{Par}(A) = u, Z = z\}, \\ &\quad \min\{\Pi(c|P_c) : C \in Z, \mathrm{Par}(A) = u, Y = y, Z = z\}, \\ &\quad \max_{d}\{\min\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, Y = y, A = a, Z = z\}\}\}\end{aligned}$$

(note that $\forall E \in Y, \mathrm{Par}(E) \subseteq Z$).

Let us denote:

- $\alpha_e = \min\{\Pi(e|P_e), E \in Y, Y = y, Z = z\}$,
- $\alpha_b = \min\{\Pi(b|P_b) : B \in \mathrm{Par}(A), \mathrm{Par}(A) = u, Z = z\}$,
- $\alpha_c = \min\{\Pi(c|P_c) : C \in Z, \mathrm{Par}(A) = u, Y = y, Z = z\}$,
- $\alpha_{r,a} = \max_d \{\min\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, Y = y, A = a, Z = z\}\} = 1$ from Lemma 4,
- $\alpha_{r,\neg a} = \max_d \{\min\{\Pi(r|P_r) : R \in D, \mathrm{Par}(A) = u, Y = y, A = \neg a, Z = z\}\}$.

Then

$$\begin{aligned}\Pi_m(a \wedge y \wedge u) &= \max_z\{\min\{\Pi(a|u), \alpha_e, \alpha_b, \alpha_c, \alpha_{r,a}\}\}\\ &= \min(\Pi(a|u), \max_z\{\min\{\alpha_e, \alpha_b, \alpha_c, \alpha_{r,a}\}\})\\ &= \min(\Pi(a|u), \max\{\max_z\{\min\{\alpha_e, \alpha_b, \alpha_c, \alpha_{r,a}\}\},\\ &\quad \max_z\{\min\{\alpha_e, \alpha_b, \alpha_c, \alpha_{r,\neg a}\}\}\})\\ &= \min(\Pi(a|u), \max\{\min\{\Pi(a|u), \max_z\{\min\{\alpha_e, \alpha_b, \alpha_c, \alpha_{r,a}\}\},\\ &\quad \min\{\Pi(\neg a|u), \max_z\{\min\{\alpha_e, \alpha_b, \alpha_c, \alpha_{r,\neg a}\}\}\}\})\\ &= \min\{\Pi(a|u), \max\{\Pi_m(a \wedge y \wedge u), \Pi_m(\neg a \wedge y \wedge u)\}\}\\ &= \min\{\Pi(a|u), \Pi_m(y \wedge u)\}.\end{aligned}$$

Our aim is to show that

$$\Pi_m(a \wedge y|u) = \min\{\Pi_m(a|u), \Pi_m(y|u)\}$$

We distinguish two cases:

- $\Pi_m(a \wedge y|u) = 1$, this means, by using the definition of conditioning, that $\Pi_m(a \wedge y \wedge u) = \Pi_m(u)$, which implies that $\Pi_m(y \wedge u) = \Pi_m(u)$ and $\Pi_m(a \wedge u) = \Pi_m(u)$, which again implies $\Pi_m(a|u) = \Pi_m(y|u) = 1$.
- $\Pi_m(a \wedge y|u) \neq 1$, this means that $\Pi_m(a \wedge y|u) = \Pi_m(a \wedge y \wedge u) < \Pi_m(u)$. The aim is to show that $\min\{\Pi_m(a|u), \Pi_m(y|u)\} = \min \Pi(a|u), \Pi_m(y \wedge u)\}$.

We consider two cases:

- $\Pi_m(y|u) \neq 1$. Then $\Pi_m(y \wedge u) = \Pi_m(y|u)$. Again, we have two cases to consider:
  - $\Pi_m(a|u) = \Pi(a|u)$, hence the equality trivially holds.
  - $\Pi_m(a|u) \neq \Pi(a|u)$. This implies using Proposition 9 that $\Pi_m(a|u) = 1$ and $\Pi(a|u) > \Pi_m(u) \geqslant \Pi_m(y \wedge u)$, hence the equality also holds.
- $\Pi_m(y|u) = 1$ which implies $\Pi_m(y \wedge u) = \Pi_m(u)$. In this case, it remains to show that

$$\Pi_m(a|u) = \min\{\Pi(a|u), \Pi_m(u)\}.$$

But this is Lemma 5. □

Proposition 11. *The two bases Σ and $E(\Sigma)$ are equivalent.*

Proof. Variables involved in Σ and $E(\Sigma)$ are A and Par(A). Hence, we can restrict to interpretations built on these variables without any loss of generality. Let a be an instance of A, u is an instance of Par(A). We have:

$$\begin{aligned}\pi_{E(\Sigma)}(a \wedge u) &= 1 - \alpha \text{ if } (\neg a \vee \neg u, \alpha) \in E(\Sigma)\\ &= 1 \text{ otherwise}\end{aligned}$$

$$\begin{aligned}
\iff \quad \pi_{E(\Sigma)}(a \wedge u) &= 1 - \max\{\alpha_i : (\neg a \vee p_i, \alpha_i) \in \Sigma \text{ and } p_i \models \neg u\} \\
&= 1 \text{ otherwise} \\
\iff \quad \pi_{E(\Sigma)}(a \wedge u) &= 1 - \max\{\alpha_i : (\neg a \vee p_i, \alpha_i) \in \Sigma \text{ and } u \models \neg p_i\} \\
&= 1 \text{ otherwise} \\
\iff \quad \pi_{E(\Sigma)}(a \wedge u) &= 1 - \max\{\alpha_i : (\neg a \vee p_i, \alpha_i) \in \Sigma \text{ and } a \wedge u \models a \wedge \neg p_i\} \\
&= 1 \text{ otherwise} \\
\iff \quad & = \pi_{\Sigma}(a \wedge u). \qquad \square
\end{aligned}$$

Proposition 12. *Let A be a variable, and $(a \vee p, \alpha)$ be a clause of Σ containing the instance a of A s.t. $\Sigma \vdash (p, \alpha)$. Then the base Σ and the knowledge base Σ' obtained from Σ by replacing $(a \vee p, \alpha)$ by (p, α) are equivalent.*

Proof. The proof is immediate. Indeed:
- Σ' implies all formulas in Σ, since the only formula which is in Σ but not in Σ', namely $(a \vee p, \alpha)$, is entailed by (p, α).
- Σ implies all formulas in Σ', since the only formula which is in Σ' but not in Σ, namely (p, α), is entailed by Σ by hypothesis. □

References

[1] S. Benferhat, D. Dubois, S. Kaci, H. Prade, A graphical reading of possibilistic knowledge bases, in: 17th Conference on Uncertainty in Artificial Intelligence, Seattle, August 1–5, 2001.

[2] N. Ben Amor, S. Benferhat, D. Dubois, H. Geffner, H. Prade, Independence in qualitative uncertainty frameworks, in: Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning (KR 2000), Breckenbridge, CO, April 12–14, 2000, pp. 235 246.

[3] S. Benferhat, D. Dubois, H. Prade, Nonmonotonic reasoning, conditional objects and possibility theory, Art. Int. 92 (1–2) (1997) 259–276.

[4] S. Benferhat, D. Dubois, H. Prade, Syntactic combination of uncertain information: a possibilistic approach, in: Proceedings of Qualitative and Quantitative Practical Reasoning (ECSQARU-FAPR'97), Lecture Notes in Artificial Intelligence, vol. 1244, Springer, Berlin, 1997, pp. 30–42.

[5] S. Benferhat, D. Dubois, H. Prade, Some syntactic approaches to the handling of inconsistent knowledge bases: a comparative study. Part II: the prioritized case, in: Orlowska, Ewa (Eds.), Logic at Work, Physica, Heidelberg, 1998, pp. 473–511.

[6] S. Benferhat, D. Dubois, L. Garcia, H. Prade, Directed possibilistic graphs and possibilistic logic, in: B. Bouchon-Meunier, R.R. Yage, L.A. Zadeh, Information Uncertainty and Fusion, Kluwer Academic Publishers, Boston, MA, 1999, pp. 365–379.

[7] S. Benferhat, D. Dubois, L.Garcia, H. Prade, Possibilistic logic bases and possibilistic graphs, in: Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence UAI'99, Stockholm, 1999, pp. 57–64.

[8] L.M. de Campos, J.F. Huete, Independence concepts in possibility theory Parts I, II, Fuzzy Sets Syst. 103 (1999) 127–152; 487–505.
[9] R.G. Cowell, A.P. Dawid, S.L. Lauritzen, D.J. Spiegelhalter, Probabilistic Networks and Expert Systems, Springer, Berlin, 1999.
[10] A.P. Dawid, Conditional independence in statistical theory, J. Roy. Statist. Soc. A 41 (1979) 1–31.
[11] D. Dubois, L. Fariñas del Cerro, A. Herzig, H. Prade, An ordinal view of independence with application to plausible reasoning, in: R. Lopezde Mantaras, D. Poole (Eds.), Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence, 1994, pp. 195–203.
[12] D. Dubois, J. Lang, H. Prade, Possibilistic logic, in: D.M. Gabbay, et al. (Eds.), Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 3, Oxford University Press, Oxford, 1994b, pp. 439–513.
[13] D. Dubois, H. Prade, Fuzzy Sets and Systems: Theory and applications, Academic Press, New York, 1980.
[14] D. Dubois, H. Prade (with the collaboration of H. Farreny, R. Martin-Clouaire, C. Testemale), Possibility Theory – An Approach to Computerized Processing of Uncertainty, Plenum Press, New York, 1988.
[15] D. Dubois, H. Prade, Representation and combination of uncertainty with belief functions and possibility measures, Comput. Intell. 4 (4) (1988) 244–264.
[16] D. Dubois, H. Prade, The logical view of conditioning and its application to possibility and evidence theories, Int. J. Approx. Reason. 4 (1) (1990) 23–46.
[17] D. Dubois, H. Prade, Inference in possibilistic hypergraphs, in: International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU'90), Lecture Notes in Computer Science, vol. 521, Springer, Berlin, 1990, pp. 250–259.
[18] D. Dubois, H. Prade, Possibilistic logic, preferential models, non-monotonicity and related issues, in: Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'91), 1991, pp. 419–424.
[19] D. Dubois, H. Prade, Possibility theory: qualitative and quantitative aspects, in: Ph. Smets (Ed.), Handbook of Defeasible Reasoning and Uncertainty Management Systems, vol. 1, Kluwer Academic Press, Dordrecht, 1998, pp. 169–226.
[20] P. Fonck, Réseaux d'inférence pour le traitement possibiliste, Thèse de doctorat, Univ. de Liège, 1993.
[21] J. Gebhardt, Learning from data: possilistic graphical models, in: D.M. Gabbay, R. Kruse (Eds.), Handbook of Defeasible Reasoning and Uncertainty Management Systems, Kluwer, Dordrecht, 2000, pp. 315–390.
[22] E. Hisdal, Conditional possibilities independence and noninteraction, Fuzzy Sets and Systems 1 (1978) 283–297.
[23] F.V. Jensen, An Introduction to Bayesian Networks, UCL Press, University College, London, 1996.
[24] D. Lehmann, M. Magidor, What does a conditional knowledge base entail? Art. Int. 55 (1) (1992) 1–60.
[25] J. Lang, Possibilistic logic: complexity and algorithms, in: Moral, et al. (Eds.), Handbook of Defeasible Reasoning and Uncertainty Management Systems, vol. 5, 2000, pp. 179–220.
[26] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, Los Altos, CA, 1988.
[27] R. Sangüesa, U. Cortés, A. Gisolfi, A parallel algorithm for building possibilistic causal networks, Int. J. Approx. Reason. 18 (1998) 251–270.
[28] P. Shenoy, Valuation-based systems for discrete optimization, in: P.P. Bonissone, M. Henrion, L.N. Kanal, J.F. Lemmer (Eds.), Uncertainty in Artificial Intelligence, vol. 6, North-Holland, Amsterdam, 1991, pp. 385–400.

[29] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, NJ, 1976.
[30] M.A. Williams, Iterated theory base change: a computational Model, in: Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95), Montreal, 1995, pp. 1541–1550.
[31] L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets and Systems 1 (1978) 3–28.