4th International Symposium on Imprecise Probabilities and Their Applications, Pittsburgh, Pennsylvania, 2005

Possibilistic networks with locally weighted knowledge bases

Salem Benferhat
CRIL, Université d'Artois
benferhat@cril.univ-artois.fr

Salma Smaoui
CRIL, Université d'Artois
smaoui@cril.univ-artois.fr

Abstract

Possibilistic networks and possibilistic logic bases are important tools to deal with uncertain pieces of information. Both of them offer a compact representation of possibility distributions. This paper studies a new representation format, called hybrid possibilistic networks, which cover both standard possibilistic networks and possibilistic knowledge bases. An adaptation of propagation algorithm for singly (resp. multiply) connected hybrid possibilistic networks is provided.

Keywords. Possibilistic networks, possibilistic logic.

1 Introduction

Possibilistic and probabilistic networks [2, 17, 19, 11, 14] are important tools proposed for an efficient representation and analysis of uncertain information.
Their success is due to their simplicity and their capacity of representing and handling independence relationships which are important for an efficient management of uncertain pieces of information.
Possibilistic networks are directed acyclic graphs (DAG), where each node encodes a variable and every edge represents a "causal" relationship between two variables. Uncertainty is expressed by means of conditional possibility distributions for each node in the context of its parents.

The inference in possibilistic graphs depends on the structure of a DAG. For simply connected graphs, the inference process can be achieved in a polynomial time. However, for multiply connected graphs, the propagation algorithm is expensive and generally requires a graphical transformation from the initial graph to another tree structure such as a junction tree. Nodes in this tree are sets of variables called clusters. The propagation algorithm efficiency depends on clusters' size, and the space complexity is function of cartesian product of clusters variables' domains.

This paper proposes a new representation of uncertain information in possibilistic networks. Local uncertainty is no longer represented by conditional possibility distributions but by possibilistic knowledge bases. The main advantage of this representation concerns space complexity. For instance, in singly connected networks, it may happen that for a given variable the number of parents can be very high. In this case, it may be impossible to provide the conditional possibility distributions for this variable. In our framework, one can only provide a compact representation of this conditional possibility distributions by means of knowledge bases. A similar remark also holds for multiply-connected networks. Namely, during the junction tree construction, it may happen that the size of clusters can be very large. In this case, it can be impossible to construct possibility distributions associated with clusters. Our representation enables us to represent possibilistic knowledge bases associated with large clusters.

The rest of this paper is organized as follows : first, we give a brief background on possibilistic logic and propagation algorithms for possibilistic networks (Section 2). Then, we present our new representation and show how standard possibilistic networks and possibilistic bases can be encoded in the framework (Section 3). The adaptation of propagation algorithms are proposed in Section 4 for polytree networks and in Section 6 for multiply connected graphs.

2 Possibilistic logic

2.1 Possibility distributions

Let $V = \{A_1, A_2, ..., A_n\}$ be a set of variables. D_{A_i} denotes the finite domain associated with the variable A_i. For the sake of simplicity, and without lost of generality, variables considered here are assumed to be binary. a_i denotes one of the two instances of A_i and $\neg a_i$ represents the other instance of A_i. $\varphi, \psi, ..$ denote propositional formulas obtained from V and logical connectors $\wedge, \vee, \neg$. $\perp$ denotes contradiction.
$\Omega = \times_{A_i \in V} D_{A_i}$ represents the universe of discourse and ω, an element of Ω, is called an *interpretation*. It is either denoted by tuples $(a_1, ..., a_n)$ or by conjunctions $(a_1 \wedge ... \wedge a_n)$, where a_i's are respective instances of A_i's. In the following, $\models$ denotes the propositional logic satisfaction. $\omega \models \varphi$ means that ω is a model of φ.

A possibility distribution π is a mapping $\Omega \to [0, 1]$. $\pi(\omega)$ denotes the compatibility degree of an interpretation ω with available pieces of information. By convention, $\pi(\omega) = 0$ means that the interpretation ω is impossible. $\pi(\omega) = 1$ means that ω is totally possible. $\pi(\omega) > \pi(\omega')$ means that ω is preferred to ω'. A possibility distribution is said to be normalized if there exists an interpretation ω such that $\pi(\omega) = 1$.
Given a possibility distribution π, two dual measures are defined:
- Possibility measure of a formula φ:

$$\Pi(\varphi) = \max\{\pi(\omega) : \omega \models \varphi\}$$

which represents the compatibility degree of φ with available pieces of information encoded by π.
- Necessity measure of a formula φ:

$$N(\varphi) = 1 - \Pi(\neg\varphi)$$

which corresponds to the certainty degree associated with φ from available pieces of information encoded by π.

Lastly, several definitions of possibilistic conditioning have been proposed in the literature [13, 9, 4, 10]. In this paper, we simply recall the conditioning definition, defined by :

$$\pi(\omega \mid \phi) = \begin{cases} 1 & if\ \pi(\omega) = \Pi(\phi) \text{ and } \omega \models \phi \\ \pi(\omega) & if\ \pi(\omega) < \Pi(\phi) \text{ and } \omega \models \phi \\ 0 & otherwise. \end{cases} \tag{1}$$

2.2 Possibilistic knowledge bases

A possibilistic knowledge base is a finite set of weighted formulas $\Sigma = \{(\varphi_i, \alpha_i), i = 1, ..., m\}$, where φ_i is a propositional formula and $\alpha_i \in [0, 1]$. (φ_i, α_i) can be viewed as a constraint stating that the certainty degree of φ_i is at least equal to α_i, namely $N(\varphi_i) \geq \alpha_i$.
Possibilistic knowledge bases are compact representations of possibility distributions. Namely, each possibilistic knowledge base induces a unique possibility distribution, defined by [8]:
$\forall \omega \in \Omega$,

$$\pi_\Sigma(\omega) = \begin{cases} 1 & if\ \forall(\varphi_i, \alpha_i) \in \Sigma,\ \omega \models \varphi_i, \\ 1 - max\{\alpha_i : \omega \not\models \varphi_i\} & otherwise. \end{cases} \tag{2}$$

Example 1 *Let $\Sigma = \{(a \vee \neg b, \frac{1}{4}), (b, \frac{1}{2}), (\neg a \vee \neg b, \frac{3}{4})\}$ be a possibilistic knowledge base. The possibility distribution π_Σ associated with Σ is given in Table 1.*

A	B	$\pi_\Sigma(AB)$
a	b	1/4
a	$\neg b$	1/2
$\neg a$	b	3/4
$\neg a$	$\neg b$	1/2

Table 1: Joint possibility distribution π_Σ

The following definitions are useful for the rest of the paper:

Definition 1 *Two possibilistic knowledge bases Σ_1 and Σ_2 are said to be equivalent if their associated possibility distributions are equal, namely :*

$$\forall \omega \in \Omega, \quad \pi_{\Sigma_1}(\omega) = \pi_{\Sigma_2}(\omega)$$

Subsumption definition is as follows :

Definition 2 *Let (φ, α) a formula in Σ. Then (φ, α) is said to be subsumed by Σ if Σ and $\Sigma \backslash \{(\varphi, \alpha)\}$ are equivalent knowledge bases.*

Namely, subsumed formulas are redundant formulas that can be removed (or added) without changing possibility distributions.

Example 2 *Let $\Sigma' = \{(a \vee \neg b, \frac{1}{4}), (b, \frac{1}{2}), (\neg a \vee \neg b, \frac{3}{4}), (a \vee b, \frac{1}{2})\}$ be a possibilistic knowledge base. The formula $(a \vee b, \frac{1}{2})$ is subsumed by Σ' since the possibility distribution $\pi_{\Sigma'}$ associated with Σ' (see Table 2) is equivalent to the one associated with $\Sigma = \Sigma' \backslash \{(a \vee b, \frac{1}{2})\}$ (see Table 1).*

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

A	B	$\pi_{\Sigma'}(AB)$
a	b	1/4
a	$\neg b$	1/2
$\neg a$	b	3/4
$\neg a$	$\neg b$	1/2

Table 2: Joint possibility distribution $\pi_{\Sigma'}$

Definition 3 *Let Σ be a possibilistic knowledge base. The inconsistency degree of Σ, denoted* $\mathrm{Inc}(\Sigma)$*, is defined by :*

$$Inc(\Sigma) = \max\{\alpha_i : \Sigma_{\geq \alpha_i} \models \bot\} \quad (3)$$

where $\Sigma_{\geq \alpha_i}$ is a set of possibilistic formulas in Σ having a weight greater or equal to α_i.

$Inc(\Sigma) = 0$ means that Σ is consistent.

Example 3 *Let $\Sigma = \{(a \vee b, \frac{3}{4}), (\neg a \vee b, \frac{1}{2}), (\neg b, \frac{1}{4})\}$ be a possibilistic knowledge base. $\Sigma \backslash \{\neg b, \frac{1}{4}\}$ is consistent. By adding the formula $(\neg b, \frac{1}{4})$, Σ becomes inconsistent. Then, $Inc(\Sigma) = \frac{1}{4}$.*

2.3 Standard possibilistic networks

Possibilistic networks, denoted ΠG, are graphically represented by directed acyclic graphs (DAG). Nodes correspond to variables and edges encode "causal" relationships among variables. A node A_j is said to be a parent of A_i if there is an edge from the node A_j to the node A_i. Parents of A_i are denoted by U_{A_i}.

Uncertainty is represented at each node by local conditional possibility distributions. More precisely, for each variable A:
If A is a root, namely $U_A = \emptyset$, then
$$\max(\Pi(a), \Pi(\neg a)) = 1.$$
If A has parents, namely $U_A \neq \emptyset$, then
$$\max_a \Pi(a \mid u_A) = 1, \forall a \in D_A, u_A \in D_{U_A}.$$
where D_{U_A} is the cartesian product of domains of variables which are parents of A.

Possibilistic networks are also compact representation of possibility distributions. More precisely, joint possibility distributions associated with min-based possibilistic network is computed using a so-called "min-based chain rule" similar to the probabilistic "chain rule" :

$$\pi_{\Pi G}(a_1, ..., a_n) = \min_{i=1..n} \Pi(a_i \mid u_{A_i}), \quad (4)$$

where a_i is an instance of A_i and $u_{A_i} \subseteq \{a_1, ..., a_n\}$ is an element of the cartesian product of domains associated with variables U_{A_i} which are parents of A_i.

Example 4 *Figure 1 gives an example of possibilistic networks. Table 3 provides local conditional possibility distributions of each node given its parents.*

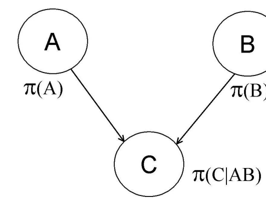

Figure 1: Example of possibilistic causal network ΠG

a	1
$\neg a$	$\frac{3}{4}$

b	$\frac{1}{2}$
$\neg b$	1

$C \mid AB$	ab	$\neg ab$	else
c	1	$\frac{1}{2}$	1
$\neg c$	$\frac{1}{4}$	1	$\frac{3}{4}$

Table 3: Local conditional possibility distributions associated with DAG of Figure 1

Using possibilistic chain rule, the joint possibility distribution $\Pi(ABC) = \Pi_{\Pi G}(ABC)$ (see Table 4) associated with the possibilistic network ΠG is computed using Equation 4.

A	B	C	$\min(\pi(A), \pi(B), \pi(C \mid AB))$
a	b	c	1/2
a	b	$\neg c$	1/4
a	$\neg b$	c	1
a	$\neg b$	$\neg c$	3/4
$\neg a$	b	c	1/2
$\neg a$	b	$\neg c$	1/2
$\neg a$	$\neg b$	c	3/4
$\neg a$	$\neg b$	$\neg c$	3/4

Table 4: Joint possibility distribution $\Pi(ABC)$

Propagation algorithms aim to establish a posteriori possibility distributions of each node A given some evidence on a set of variables E.
Propagation algorithms on polytree (singly connected networks) can be achieved in polynomial time, while propagation algorithms on multiply connected graphs are NP-complete [5].

3 Possibilistic networks with local knowledge bases

3.1 Definition of hybrid graphs

Pieces of information can be provided either in terms of possibilistic knowledge bases or in terms of conditional possibilities (if the size of universe of discourse is reasonable). They can also be represented either using graphical structures or logic-based structures. The aim of the new representation is to take advantage of these two possible representation formats. Graphical representation is used to take advantage of independence relations, and logic-based representation is used to have compact representation of possibility distributions.

There have been some works which exploit complementarity between classic logic and local propagation. An example of these works is Wilson and Mengin's work [21] which uses local computation for reasoning on classic logic.

This paper deals with hybrid possibilistic graphs (see also [3]). More precisely, hybrid possibilistic causal networks, denoted HG, are characterized by :

- *A graphical component* which is represented by a DAG (like standard possibilistic causal networks) that allows to represent independence relationships.
- *A quantitative component* which encodes uncertainties. It associates to each node a local knowledge bases instead of a conditional possibility distribution. Namely, at each node A_i, one provides a possibilistic knowledge base Σ_{A_i} which represents local knowledge base on A and its parents. Figure 2 provides an example of hybrid possibilistic graphs.

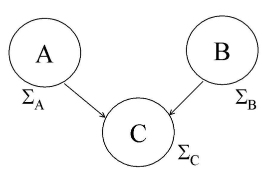

Figure 2: Hybrid graph HG with local knowledge bases

Hybrid graphs are also compact representations of joint possibility distributions. Namely, a possibility distribution associated with a hybrid possibilistic network HG is defined by :

$$\forall\omega,\ \pi_{HG}(\omega) = \min_{A_i \in V} \pi_{\Sigma_{A_i}}(\omega) \qquad (5)$$

where $\pi_{\Sigma_{A_i}}$ is the possibility distributions associated with Σ_{A_i} obtained using equation 6.

Next section shows that any possibilistic network ΠG (where local uncertainty is represented by a possibility distribution), can be represented by hybrid networks HG.

3.2 From ΠG to HG

We start by considering standard possibilistic causal networks ΠG where uncertainty is represented at the level of nodes by conditional possibility and a priori distributions.

Let A be a binary variable and a_i be an instance of this variable. Let $\pi(a_i|u_i)$ be a local possibility degree associated with A where u_i is an element of cartesian product of its parents (U_A) domains. Let us associate the following possibilistic knowledge base with the node A :

$$\Sigma_A = \{(\neg a_i \vee \neg u_i, \alpha_i) : \alpha_i = 1 - \pi(a_i|u_i) \neq 0\}. \quad (6)$$

It's easy to check that the conditional possibilities are recovered from Σ_A using Equation 2. $\pi_{\Sigma_{A_i}}(a_i \wedge u_i) = 1 - \alpha_i$ holds since interpretations that satisfy $a_i \wedge u_i$ falsify $(\neg a_i \vee \neg u_i, \alpha_i)$. Normalization is also assured since $\pi_{\Sigma_{A_i}}(u_i) = \max(\pi_{\Sigma_{A_i}}(a_i \wedge u_i), \pi_{\Sigma_{A_i}}(\neg a_i \wedge u_i)) = 1$ (i.e. $\neg a_i \wedge u_i$ satisfies $(\neg a_i \vee \neg u_i, \alpha_i)$).

Then, it can be easily proved that $\forall\omega$,

$$\pi_{\Pi G}(\omega) = \pi_{HG}(\omega) \qquad (7)$$

where $\pi_{\Pi G}$ and π_{HG} are obtained by using equations 4 and 5.

Example 5 *Let us build a hybrid possibilistic causal networks HG from standard possibilistic causal network ΠG of example 4 by associating knowledge bases to each node using 6. Uncertainty at the level of nodes A, B and C (binary variables) is represented by possibilistic knowledge bases Σ_A, Σ_B and Σ_C as follows:*

$\Sigma_A = \{(a, \frac{1}{4})\}$

$\Sigma_B = \{(\neg b, \frac{1}{2})\}$

$\Sigma_C = \{(\neg a \vee \neg b \vee c, \frac{3}{4}), (a \vee \neg b \vee \neg c, \frac{1}{2}), (\neg a \vee b \vee c, \frac{1}{4}), (a \vee b \vee c, \frac{1}{4})\}$

We can observe that the joint possibility distribution $\pi_{HG}(ABC)$ (given in Table 5 using Equation 5) is the same as the one given in Example 4. We can check that, $\forall\omega$, $\pi_{\Pi G}(\omega) = \pi_{HG}(\omega)$. For instance, $\pi_{HG}(\neg ab\neg c) = \pi_{\Pi G}(\neg ab\neg c) = \frac{1}{2}$.

3.3 From HG to ΠG

Each hybrid possibilistic network can also be expressed by a standard possibilistic network, provided

ω	π_{Σ_A}	π_{Σ_B}	π_{Σ_C}	π_{HG}
abc	1	1/2	1	1/2
$ab\neg c$	1	1/2	1/4	1/4
$a\neg bc$	1	1	1	1
$a\neg b\neg c$	1	1	3/4	3/4
$\neg abc$	3/4	1/2	1/2	1/2
$\neg ab\neg c$	3/4	1/2	1	1/2
$\neg a\neg bc$	3/4	1	1	3/4
$\neg a\neg b\neg c$	3/4	1	3/4	3/4

Table 5: Joint possibility distribution $\Pi_{HG}(ABC)$

that the number of parents of each node is not very high. The graphical component is the same. The conditional possibility distributions are simply the ones associated with the knowledge bases. More precisely, let A_i be variable and u_i be an element of the cartesian product of domains associated with variables U_{A_i} which are parents of A_i. Let Σ_{A_i} be the local knowledge base associated with the node A_i. Then, the conditional possibility degree $\pi(a_i|u_i)$ is defined by $\pi(a_i|u_i) = \pi(a_i \wedge u_i) = \pi_{\Sigma_{A_i}}(a_i \wedge u_i)$ and $\pi_{\Sigma_{A_i}}(a_i \wedge u_i)$ is defined using Equation 2

Example 6 Let us consider the hybrid possibilistic network built in Example 5. The standard possibilistic network is obtained by computing, for each variable in the network, the local conditional possibility degree from the local possibility knowledge base associated with the node using Equation 2. Uncertainty at the level of nodes A, B and C (binary variables) is represented by possibiliy distributions:

a	1
$\neg a$	$\frac{3}{4}$

b	$\frac{1}{2}$
$\neg b$	1

$C\|AB$	ab	$\neg ab$	else
c	1	$\frac{1}{2}$	1
$\neg c$	$\frac{1}{4}$	1	$\frac{3}{4}$

which are the same that in Table 3. Then, we can check that, $\forall\omega$, $\pi_{\Pi G}(\omega) = \pi_{HG}(\omega)$. For instance,

$$\begin{aligned}\pi_{\Pi G}(\neg ab\neg c) &= \min(\pi(\neg a), \pi(b), \pi(\neg c|\neg ab)) \\ &= \min(\tfrac{3}{4}, \tfrac{1}{2}, 1) = \tfrac{1}{2}.\end{aligned}$$

which is the same as $\pi_{HG}(\neg ab\neg c)$ given in Example 5.

3.4 From HG to Σ

The knowledge base Σ built from a hybrid possibilistic network HG is the result of the fusion of elementary bases. These bases correspond exactly to the local possibilistic knowledge bases $\{\Sigma_{A_i} : i = 1, ..., n\}$ provided at the level of each node $\{A_i : i = 1, ..., n\}$ in the network.

Proposition 1 The possibilistic knowledge base Σ associated with a hybrid possibilistic network HG is:

$$\Sigma = \bigcup_{i=1,\dots,n} \Sigma_{A_i} \tag{8}$$

where $\{\Sigma_{A_i} : i = 1, ..., n\}$ are local possibilistic knowledge bases associated with nodes $\{A_i : i = 1, ..., n\}$ in HG.

Then,

$$\pi_{HG}(\omega) = \pi_{\Sigma}(\omega) \tag{9}$$

Example 7 *Let us consider the hybrid possibilistic graph HG in Figure 2 and its local possibilistic knowledge bases built in Example 5. The knowledge base Σ built from this hybrid possibilistic network is:*

$$\begin{aligned}\Sigma &= \Sigma_A \cup \Sigma_B \cup \Sigma_C \\ &= \{(a, \tfrac{1}{4}), (\neg b, \tfrac{1}{2}), (\neg a \vee \neg b \vee c, \tfrac{3}{4}), (\neg a \vee b \vee c, \tfrac{1}{4})\}\end{aligned}$$

The formulas $(a \vee \neg b \vee \neg c, \frac{1}{2})$ and $(a \vee b \vee c, \frac{1}{4})$ are subsumed formulas.

We can check that $\forall\omega$, $\pi_{HG}(\omega) = \pi_{\Sigma}(\omega)$. For instance,
$\pi_{\Sigma}(\neg ab\neg c) = 1 - \frac{1}{2} = \frac{1}{2}$
which is the same that $\pi_{HG}(\neg ab\neg c)$ computed in Example 5.

Note that the transformation from HG into Σ results in the loss of independence relations present in HG.

3.5 From Σ to HG

The encoding of possibilistic knowledge base Σ is immediate. Its associated hybrid possibilistic network HG can be constructed in the following way :
- Select arbitrary a variable A. Assign to A the knowledge base Σ.
- For each variable $B \neq A$, add a link from B to A.
- Assign an empty possibilistic knowledge base to B.
Then,

$$\pi_{HG}(\omega) = \pi_{\Sigma}(\omega) \tag{10}$$

since $\pi_{HG} = \pi_{\Sigma_A}$ and $\Sigma_A = \Sigma$.

Example 8 *Let us consider the possibilistic knowledge base Σ in Example 7. We propose to build a hybrid possibilistic network from Σ.*

- We select the variable A. Then, we assign to A the knowledge base $\Sigma_A = \Sigma$,

- We add a link from B to A and from C to A,
- We assign to B and C empty knowledge bases: $\Sigma_B \leftarrow \emptyset$, $\Sigma_C \leftarrow \emptyset$.

It can be checked that $\forall \omega$, $\pi_{HG}(\omega) = \pi_\Sigma(\omega)$. For instance,
$\pi_{HG}(\neg ab\neg c) = \min(\frac{3}{4}, \frac{1}{2}, 1) = \frac{1}{2}$
which is the same that $\pi_\Sigma(\neg ab\neg c)$ computed in Example 7 where π_Σ is obtained using Equation 2 and π_{HG} is obtained using Equation 5.

4 Prior propagation in hybrid singly-connected possibilistic networks

Polytrees are simply connected networks where there are no two nodes that can be connected by more than one path. For sake of simplicity, we only recall prior propagation. The propagation process is based on message passing from roots to leafs. Roots send their a priori distributions to their children. Each node (except roots), receiving a message, computes its own marginal distribution and send it to its children. The marginal distribution $\Pi(A)$ at the level of each node A is obtained by the following equation :

$$\begin{aligned}\pi(a) &= \max_{u_1,..,u_n} \min \pi(a, u_1, ..., u_n),\\ &= \max_{u_1,..,u_n} \min(\pi_i(a|u_1, .., u_n), \pi_i(u_1, .., u_n),\\ &= \max_{u_1,..,u_n} \min(\pi_i(a|u_1, .., u_n), \pi_i(u_1), .., \pi_i(u_n))\end{aligned} \tag{11}$$

where u_i $(i = 1, .., n)$ is an instance of U_i $(i = 1, .., n)$ which is a parent of A. $\pi(u_i)$ denotes local possibility measures at the level of the node U_i (parent of A) when $U_i = u_i$ after receiving message from it's parents. In the last step, $\pi_i(u_1, .., u_n) = \min(\pi_i(u_1), .., \pi_i(u_n))$ is used since parents of a common node are independent (i.e. parents are d-separated by all their common children).

Propagation algorithms can be efficiently achieved for polytree structures. Therefore, the only situation where hybrid graphs can be useful for polytree is when the set of parent's variables is important, from which, it is impossible to represent conditional possibility distributions.
In this section, we present the counterpart of a priori propagation algorithm for polytree networks using hybrid representation of uncertainty.
The prior propagation algorithm consists of computing the marginal $\Pi(A)$ for each node A. The latter collects information from its parents to update its own beliefs. The collected information in a standard possibilistic network consists of the marginals $\{\pi(U_i) : i = 1, ..., n\}$ of the node's parents (see Equation 11). In case of hybrid representation, these marginals can be computed from local knowledge bases $\{\Sigma_{U_i} : i = 1, ..., n\}$. The main steps of the algorithm are as follows :

4.1 Receiving messages:

When a node A receives a message $\pi(U_i)$ from its parent U_i. A formula $(\neg u_i, 1 - \pi(u_i))$ is added to the local knowledge base Σ_A for each instance u_i of U_i. These added formulas represent the syntactical counterpart of marginalization of the local knowledge Σ_{U_i} (i.e. $\pi(U_i)$) by considering the unique variable U_i.

4.2 Sending messages:

After receiving messages from all its parents, each node A computes the message to send to its children. This message represents the marginal distribution on the variable A. It can be directly and syntactically obtained from the local knowledge base Σ_A by considering the following proposition :

Proposition 2 *Let Σ be a possibilistic knowledge base. Let a be an instance of A. Then,*

$$\pi(a) = 1 - Inc(\Sigma \cup \{(a, 1)\})$$

where $Inc(\Sigma \cup \{(a, 1)\})$ is the inconsistency degree of $\Sigma \cup \{(a, 1)\}$. For computing the inconsistency degree Inc see [8].

Lang [16] proposed an algorithm to compute the inconsistency degree of Σ which requires $log_2\ n$ satisfiability checks using any prover for the propositional satisfiability problem SAT where n is the number of different valuations involved in Σ.

Example 9 *Let us consider the singly-connected possibilistic network given in the Figure 1 and its local knowledge bases defined in Example 5. Suppose that A is sending a message to C. The node A computes the message to send to its child C:*

- $\pi(a) = 1 - Inc(\Sigma_A \cup \{(a, 1)\}) = 1 - 1 = 0$,
- $\pi(\neg a) = 1 - Inc(\Sigma_A \cup \{(\neg a, 1)\}) = 1 - \frac{1}{4} = \frac{3}{4}$.

The node C receiving the message, add the formula $(a, \frac{1}{4})$ to its local knowledge base Σ_C.

5 Propagation in multiply connected graphs

One of well-known algorithm to deal with multiply connected graphs (graphs containing loops) proceeds to a transformation of the initial graph into a junction tree. The main idea is to delete loops from the initial graph gathering some variables in the same node. The resulting graph is a tree where each node, called cluster, is a set of variables. Common variables of two adjacent clusters are grouped into another type of node, called separator.
The propagation is performed by a message passing mechanism. The algorithm stops when the junction tree is globally consistent, namely when adjacent clusters have the same marginal distributions over common variables. For more details on junction tree propagation algorithm in possibility theory framework see [11, 1]. One of the limits of junction tree algorithm is that the transformation step of initial multiply connected graph can produce clusters with a great number of variables. In that case, it may be impossible to get local joint possibility distributions on clusters.
We call hybrid junction tree, denoted HJT, a junction tree where uncertainty is represented over clusters by possibilistic knowledge bases, instead of possibility distributions.

Before introducing the propagation algorithm in hybrid junction tree, we need to present the notion of prioritized forgetting (see [3]) which allow to give the syntactic counterpart of a marginalization process. This approach which dealing with possibilistic knowledge bases is an extension of the one proposed by Lin and Reiter [18] for classical propositional logic (also see Darwiche, Lang and Marquis's works [15, 6] for details).
Let Σ_1 and Σ_2 be two possibilistic knowledge bases. The disjunction of these two bases in possibilistic framework, denoted $\bigcirc\!\!\!\!\vee$, is defined as follows :
$\Sigma_1 \bigcirc\!\!\!\!\vee \Sigma_2 = \{(\varphi_i \vee \psi_j, \min(\alpha_i, \beta_j)) : (\varphi_i, \alpha_i) \in \Sigma_1$ and $(\psi_j, \beta_j) \in \Sigma_2\}$
Prioritized forgetting, denoted $pforget$, can then be defined as follows:

Definition 4 *Let Σ be a possibilistic knowledge base and X be a variable set. The prioritized forgetting of X in Σ, denoted $pforget(\Sigma, X)$, is equivalent to a possibilistic formula defined as follows :*

- $pforget(\Sigma, \emptyset) = \Sigma$,
- $pforget(\Sigma, \{x\}) = \Sigma_1 \bigcirc\!\!\!\!\vee \Sigma_2$,
- $pforget(\Sigma, X \cup \{x\}) = pforget(pforget(\Sigma, X), \{x\})$.

Prioritized forgetting allows to syntacticly capture the base associated with marginal distributions.
Next subsections present the different steps of possibilistic propagation adaptation in case of hybrid representation.

To illustrate main concepts of the propagation algorithm, we will use the following causal networks :

Example 10 *Figure 3 gives an example of multiply-connected possibilistic networks. Table 6 provides conditional possibility distributions of each node given its parents.*

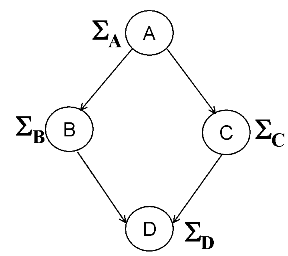

Figure 3: Multiply-connected hybrid possibilistic causal network ΠG

a	$\frac{1}{4}$
$\neg a$	1

$B\|A$	a	$\neg a$
b	$\frac{1}{4}$	$\frac{1}{4}$
$\neg b$	1	1

$C\|A$	a	$\neg a$
c	1	$\frac{1}{2}$
$\neg c$	$\frac{3}{4}$	1

$D\|BC$	bc	$\neg bc$	else
d	1	$\frac{1}{4}$	1
$\neg d$	$\frac{1}{2}$	1	1

Table 6: Local conditional possibility distributions associated with DAG of Figure 3

The joint possibility distribution associated with this network is given in the table 7. Let us build a hybrid possibilistic causal network HG from the standard possibilistic causal network ΠG by associating knowledge bases to each node using Equation 6.
Uncertainty at the level of nodes A, B, C and D is represented by possibilistic knowledge bases Σ_A, Σ_B, Σ_C and Σ_D as follows:

$\Sigma_A = \{(\neg a, \frac{3}{4})\}$
$\Sigma_B = \{(\neg a \vee \neg b, \frac{3}{4}), (a \vee \neg b, \frac{3}{4})\}$
$\Sigma_C = \{(a \vee \neg c, \frac{1}{2}), (\neg a \vee c, \frac{1}{4})\}$
$\Sigma_D = \{(b \vee \neg c \vee \neg d, \frac{3}{4}), (\neg b \vee \neg c \vee d, \frac{1}{2})\}$

A	B	C	D	$\pi(ABCD)$
a	b	c	d	1/4
a	b	c	$\neg d$	1/4
a	b	$\neg c$	d	1/4
a	b	$\neg c$	$\neg d$	1/4
a	$\neg b$	c	d	1/4
a	$\neg b$	c	$\neg d$	1/4
a	$\neg b$	$\neg c$	d	1/4
a	$\neg b$	$\neg c$	$\neg d$	1/4
$\neg a$	b	c	d	1/4
$\neg a$	b	c	$\neg d$	1/4
$\neg a$	b	$\neg c$	d	1/4
$\neg a$	b	$\neg c$	$\neg d$	1/4
$\neg a$	$\neg b$	c	d	1/4
$\neg a$	$\neg b$	c	$\neg d$	1
$\neg a$	$\neg b$	$\neg c$	d	1
$\neg a$	$\neg b$	$\neg c$	$\neg d$	1

Table 7: Joint possibility distribution $\Pi(ABCD)$

5.1 Initialization

This step consists of initializing the junction tree by assigning knowledge bases to clusters and separators.

- An empty knowledge base Σ_{C_i} is first assigned to each cluster C_i.

$$\Sigma_{C_i} \leftarrow \emptyset$$

- An empty knowledge base $\Sigma_{S_{ij}}$ is also assigned to each separator S_{ij}.

$$\Sigma_{S_{ij}} \leftarrow \emptyset$$

- For each binary variable A, select a cluster C_i containing $\{A\} \cup U_A$ and we add to the knowledge base Σ_{C_i} the possibilistic base Σ_A associated with A.

$$\Sigma_{C_i} \leftarrow \Sigma_{C_i} \cup \Sigma_A$$

If there are some observations (evidence), then for any observed variable $A_i = a_i$ select a cluster containing the variable A_i, and add the possibilistic formula $(a_i, 1)$ to the knowledge base associated with this cluster.

Proposition 3 *Let HG be a hybrid possibilistic causal network. Let HJT be the junction tree associated to HG. Let $\{\Sigma_{C_i} : i = 1, ..., n\}$ be the knowledges bases associated to clusters $\{C_i : i = 1, ..., n\}$ at the end of the initialization step. Then we have:*

$$\pi_{HG} = \min_{C_i} \pi_{\Sigma_{C_i}}$$

Example 11 *Given the junction tree (Figure 4) built from the hybrid possibilistic graph HG given in Figure 10, local knowledge bases on clusters after the initialization step are as the following :*
- $\Sigma_{C_1} = \Sigma_A \cup \Sigma_B \cup \Sigma_C = \{(\neg a, \frac{3}{4}), (\neg a \vee \neg b, \frac{3}{4}), (a \vee \neg b, \frac{3}{4}), (a \vee \neg c, \frac{1}{2}), (\neg a \vee c, \frac{1}{4})\}$
- $\Sigma_{C_2} = \Sigma_D = \{(b \vee \neg c \vee \neg d, \frac{3}{4}), (\neg b \vee \neg c \vee d, \frac{1}{2})\}$

Let us consider the interpretation $\omega = \neg ab\neg cd$. We have :
$\pi_{HG}(\neg ab\neg cd) = \min(\pi_{\Sigma_{C_1}}(\neg ab\neg c), \pi_{\Sigma_{C_2}}(b\neg cd)) = \frac{1}{4}$
which is the same as the one obtained from Example 10.

After the initialization step, messages are sent between clusters in order to guaratee the consistency conditions.
If, for instance, for given two clusters C_i and C_j we have

$$\max_{C_i \setminus S_{ij}} \pi_{C_i} \neq \max_{C_j \setminus S_{ij}} \pi_{C_j},$$

then C_i and C_j should update their knowledge bases iteratively.

Example 12 *The junction tree given in Example 11 is not globally consistent: the joint distribution among B and C computed from Σ_{C_1} is different from the one computed from Σ_{C_2}. For instance,*
$\pi_{\Sigma_{C_1}}(bc) = \frac{1}{4}$ *is different from* $\pi_{\Sigma_{C_2}}(bc) = 1$.

The following two elementary steps are repeated until reaching consistency:

- A separator S_{ij} computes its knowledge base from C_i (resp. C_j).
- A cluster C_j (resp. C_i) updates its knowledge base taking into account knowledge base of the separator previously computed.

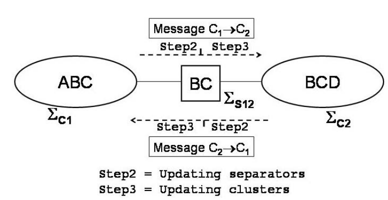

Figure 4: Message passing in the junction tree HJT

5.2 Updating separators

The knowledge base $\Sigma_{S_{ij}}$, associated with a separator S_{ij}, represents the restriction (marginalization) of the base Σ_{C_i} (resp. Σ_{C_j}) on common variables in the separator S_{ij} . This knowledge base is immediately

obtained using the prioritized forgetting notion.
Let V' be the set of variables in $C_i \backslash S_{ij}$. Then,

$$\Sigma_{S_{ij}} = pforget(\Sigma_{C_i}, V')$$

Example 13 *Let us compute the knowledge base $\Sigma_{S_{12}}$, associated with the separator S_{12} from Σ_{C_1}. This leads to forgetting the variable A. Let us apply the definition of pforget :*
- $\Sigma_{a \leftarrow \perp} = \{(\neg b, \frac{3}{4}), (\neg c, \frac{1}{2})\}$
- $\Sigma_{a \leftarrow \top} = \{(\perp, \frac{3}{4}), (\neg b, \frac{3}{4}), (c, \frac{1}{4})\}$

$\Sigma_{S_{12}} = pforget(\Sigma_{C_1}, \{A\}) = \{(\neg b, \frac{3}{4}), (\neg c, \frac{1}{2}), (\neg b \vee \neg c, \frac{1}{2}), (\neg b \vee c, \frac{1}{4})\} = \{(\neg b, \frac{3}{4}), (\neg c, \frac{1}{2})\}$.
$(\neg b \vee \neg c, \frac{1}{2})$ *and* $(\neg b \vee c, \frac{1}{4})$ *are subsumed formulas.*

5.3 Updating clusters

When receiving message from separator S_{ij}, the cluster C_i updates its knowledge base as follows :

$$\Sigma_{C_j} \leftarrow \Sigma_{S_{ij}} \cup \Sigma_{C_j} \quad (12)$$

This is justified by the following proposition :

Proposition 4 *Let HG be a hybrid possibilistic causal network. Let HJT a junction tree associated with HG. Let $\{\Sigma_{C_i} : i = 1, ..., n\}$ be the knowledge bases associated to clusters $\{C_i : i = 1, ..., n\}$ after each updating step (see Equation 12). Then, we have : $\forall \omega$,*

$$\pi_{HG}(\omega) = min_{C_i} \pi_{\Sigma_{C_i}}(\omega)$$

The algorithm continues updating separators and clusters until reaching stability (global consistency) in the junction tree. Formally, after propagation, $\Sigma_{S_{ij}}$ must satisfy the following condition:

$$\pi_{\Sigma_{S_{ij}}} = \max_{C_i \backslash S_{ij}} \pi_{\Sigma_{C_i}} = \max_{C_i \backslash S_{ij}} \pi_{\Sigma_{C_j}} \quad (13)$$

Example 14 *The knowledge base Σ_{C_2} associated with the cluster C_2 receiving $\Sigma_{S_{12}}$ is :*

$$\begin{aligned} \Sigma_{C_2} &= \Sigma_{C_2} \cup \Sigma_{S_{12}} \\ &= \{(b \vee \neg c \vee \neg d, \tfrac{3}{4}), (\neg b, \tfrac{3}{4}), (\neg c, \tfrac{1}{2})\} \end{aligned}$$

The formula $(\neg b \vee \neg c \vee d, \frac{1}{2})$ is subsumed by $(\neg b, \frac{3}{4})$.

At the end of propagation process, we obtain the following local knowledge bases:
- $\Sigma_{C_1} = \{(\neg a, \frac{3}{4}), (\neg b, \frac{3}{4}), (\neg c, \frac{1}{2})\}$.
- $\Sigma_{C_2} = \{(\neg b, \frac{3}{4}), (\neg c, \frac{1}{2}), (b \vee \neg c \vee \neg d, \frac{3}{4})\}$.

It can be checked that HJT is consistent.

6 Conclusion

This paper has presented hybrid possibilistic causal networks where knowledge bases are used instead of local conditional possibility distributions. We have shown how propagation algorithms in polytrees, and in junction trees, can be adapted. For clarity reasons, variables were restricted to binary ones. However, the results in this paper can be extended to non-binary variables.
For polytrees, our representation can only be useful for a specific situation where nodes can have a high number of parents. For multiply connected graphs, our representation is very useful.

First experimental results [3] are promising since the algorithm allows to deal with examples of multiply connected graphs that cannot be represented by junction tree with local conditional possibility distributions.

This paper basically focuses on improving one of well-known possibilistic propagation algorithms, which is based on junction tree construction. The improvement concerns space complexity namely when clusters' sizes are large.
Of course, there are alternative approaches to junction tree like local computation frameworks [21, 20, 12] on possibilistic hypergraphs [9].
Other approaches use local computation ideas for computing propositional or possibilistic logical deduction [21, 20]. These approaches have perspectives different from the one paper.

A future work will be to experimentally compare these alternative approaches with our algorithm. We will also consider the use of compilation approaches (like the one based on d-DNNF format [7]) for computing prioritized forgetting variables.

Acknowledgment

This work has been supported by the French national project ACI (Actions Concertées Incitatives), called DADDi (Dependable Anomaly Detection with Diagnosis).

References

[1] N. Benamor. *Qualitative possibilistic graphical models : From independence to propagation algorithms. PHD Thesis, Institut supérieur de gestion*: Tunis, 2002.

[2] S. Benferhat, D. Dubois, L. Garcia, H. Prade. On the transformation between possibilistic logic bases and possibilistic causal networks. *Interna-*

tional Journal of Approximate Reasoning, Vol. 29, N. 21, 35-173, 2002.

[3] S. Benferhat, S. Smaoui. Hybrid possibilistic networks. *To appear in proceeding of the Twentieth National Conference on Artificial Intelligence (AAAI-05),AAAI Press*. Pittsburgh, 2005.

[4] G. De Cooman, E. E. Kerre. A new approach to possibilistic independence. *Proceeding of the Third IEEE International Conference on Fuzzy Systems, 1994*, pp. 1446-1451.

[5] G. F. Cooper. Computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, vol 42, 393-405, 1990.

[6] A. Darwiche and P. Marquis. Compiling propositional weighted bases. *Artif. Intell.,Elsevier Science Publishers Ltd*. vol 157, number 1-2, pp.81-113,Essex, UK.

[7] A. Darwiche. New Advances in Compiling CNF into Decomposable Negation Normal Form. *ECAI 2004*: pp.328-332

[8] D. Dubois ,J. Lang, H. Prade. Possibilistic logic. *Handbook on Logic in Artificial Intelligence and Logic Programming*. vol 3, pp.439-513. Oxford University press, 1994.

[9] D. Dubois, H. Prade. Possibility theory: An approach to computerized processing of uncertainty.*Plenum Press, New York*, 1988.

[10] P. Fonck. A Comparative Study of Possibilistic Conditional Independence and Lack of Interaction. *International Journal of Approximate Reasoning*, vol 16, 149-171, 1997.

[11] P. Fonck. *Réseaux d'inférence pour le raisonnement possibiliste*. Université de Liège, Faculté des Science, 1994.

[12] L. Hernandez, S. Moral. Inference with Idempotent Valuations. *Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence (UAI-97)*: 229-237, San Francisco, CA,1997,Morgan Kaufmann Publishers.

[13] E. Hisdal. Conditional possibilities independence and non interaction. *Fuzzy Sets ans Systems*, 1:283-297, 1978.

[14] F. V. Jensen. *Introduction to Bayesien networks*, UCL Press, University college, London, 1996.

[15] J. Lang, P. Marquis. Complexity results for independence and definability. *Proceeding of the 6th International Conference on Knowledge Representation and Reasoning (KR'98)*: 356-367, Trento, 1998.

[16] J. Lang. Possibilistic logic: complexity and algorithms. *Handbook of Defeasible Reasoning and Uncertainty Management Systems (D. Gabbay and Ph. Smets, eds.)*: Vol. 5, 179-220, 2000, Kluwer Academic Publishers.

[17] S. L. Lauritzen ,D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society* , vol 50, pp.157-224, 1988.

[18] F. Lin, R. Reiter. Forget it!. *Proceeding of AAAI Fall Symposium on Relevance*: 154-159, New Orleans (LA), 1994.

[19] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. San Francisco (California). Morgan Kaufmman, 1988.

[20] N. Wilson, J. Mengin. Logical deduction using the local computation framework. *Lecture Notes in Computer Science*: vol 1638, pp. 386–??,1999.

[21] N. Wilson, J. Mengin. Embedding logics in the local computation framework. *Journal of Applied Non-Classical Logics*, vol 11(3-4) : 239-267, 2001.