Possibilistic Conditional Independence: a similarity-based measure and its application to causal network learning

Ramón Sangüesa Solé sanguesa@lsi.upc.es
Joan Cabós Fabregat jcabos@goliat.upc.es
Ulises Cortés García ia@goliat.upc.es
Dept. Llenguatges i Sistemes Informátics
Universitat Politécnica de Catalunya
Pau Gargallo, 5, 08028 Barcelona
SPAIN

Abstract

A new definition for similarity between possibility distributions is introduced and discussed as a basis for detecting dependence between variables by measuring the similarity degree of their respective distributions. This new definition is used to detect conditional independence relations in possibility distributions derived from data. This is the basis for a new hybrid algorithm for recovering possibilistic causal networks. The algorithm POSSCAUSE is presented and its applications discussed and compared with analogous developments in possibilistic and probabilistic causal networks learning.

1. Learning causal networks: the possibilistic case

As more and more databases are used as a source for Knowledge Discovery (Piatetsky-Shapiro & Frawley, 1991), the interest of automating the construction of a well-defined and useful knowledge representation as belief networks (Pearl, 1994, 1993, 1988), becomes apparent. Several methods have been devised to recover both the structure and the probability distributions corresponding to it. Such methods can be roughly divided into *goodness of fit* methods (Cooper & Herskovitz, 1992; Herskovitz & Cooper, 1990; Heckerman, 1995), *conditional independence test* methods (Rebane & Pearl, 1989; Pearl & Verma, 1991; Verma & Pearl, 1992)and *hybrid* methods (Singh & Valtorta, 1995, 1993). The first ones approximate a belief network that is as close as possible to the joint distribution implied by a database of cases; their advantage is that they can rank several possible resulting networks, but their tendency to approach as close as possible the data tends to result in networks that are too complicated to be easily understood an used by humans. The second family of methods, uses tests for conditional independence between variables to recover a tentative structure and, then, calculate the corresponding conditional probability distributions; their disadvantage is their need of a pre-determined order between variables and the demands on data in order to carry on correct statistical tests. Finally, hybrid methods, combine the first and second kind of methods in order to recover a network.

All these methods have been applied using a single uncertainty formalism, i.e., probability. However, uncertainty about a domain can be due to other factors beyond those for which probability is adequate. When imprecision or ambiguity are inherent to the domain, possibility theory (Dubois & H, 1986; Gebhardt & Kruse, 1993) is a good alternative.

These circumstances (imprecision and ambiguity) do arise in many real-world situations. For example, data may come from multiple sensors with unknown fault probability (Josslyn, 1994). Some tasks, too, may have some degree of ambiguity as it is the case in diagnosis when there is added uncertainty about symptoms being related to more than one fault in a non-exclusive way (Dubois & Prade, 1990).

The idea that belief networks can use uncertainty formalisms other than probability is, thus, a natural development. Several alternative formalizations exist: valuation-based systems (Shenoy, 1991; Cano, Delgado, & Moral, 1993); possibilistic networks (Fonck, 1993, 1992, 1991), probability intervals (De Campos & Huete, 1993). Due to the peculiar characteristics of such formalisms, new learning methods have been devised. In the context of possibilistic networks some interesting work has been done by Gebhardt and Kruse (Gebhardt & Kruse, 1995) in creating a learning method for possibilistic networks along lines similar to previous work in bayesian learning (Cooper & Herskovitz, 1992).

Our aim in this paper has been to develop a method for building possibilistic networks that reflects in a consistent way all the dependence relations present in a database. We felt that the learning method had to be as independent as possible from expert knowledge and that it should recover as precise (i.e. close to the data) but, at the same time, as easy to understand networks as possible. Moreover, we were interested in using it with real-world data where imprecision pervaded all knowledge. So, possibility theory was a natural choice. Problems, however arose in several shortcomings of current possibilistic counterparts of concepts such as independence, conditioning and measurement of possibilistic information. So, we have put forth new definitions and mesures that have proven quiet useful in our work.

In the first section we will review the basic concepts of *epossibilistic withpossibilistic withxtended belief networks*, conditioning, and independence in possibilistic settings; in the second section a new measure of possibilistic dependence is discussed that combines similarity and information relevance concepts; the third section shows how this measure can be applied to a learning method; in the fourth section we comment on our experiences in applying **POSSCAUSE**(Possibilistic Causation) to real-world data and the last section is devoted to conclusions and future lines of research.

2. General belief networks and possibilistic causal networks

We modify slightly here the notion of belief network usually identified with bayesian networks.

Definition 2.1 *General belief network*
For a domain $U = \{x_1 \dots x_n\}$ the corresponding belief network is a directed acyclic graph (DAG) where nodes stand for variables and links for direct association between variables. Each link is quantified by the conditional uncertainty distribution relating the variables connected to it, $\mathcal{P}$. By uncertainty distribution we mean the distribution based on any confidence measure used to represent uncertainty about evidence.

Belief networks have two interesting characteristics. Firstly, any given node x_i in a belief network is conditionally independent of the rest of the variables in U, given its direct predecessors in the graph, i.e., its parents 'shield' the variable from the influence of the

previous variables in the graph. Secondly, the joint uncertainty distribution induced by the DAG representing the dependences in a given domain can be factorized into the conditional distribution of each variable with respect to its inmediate predecessors ('parents'). That is:

$$\mathcal{P}(x_1 \ldots x_n) = \otimes \mathcal{P}(x_i|pa_i)$$

where $\mathcal{P}$ represents an uncertainty distribution (probability, possibility, etc.) and $\otimes$ is a factorizing operator. In the case of probability this operator is the product of conditional distributions (Pearl, 1988); in the case of possibility it can be the product or the minimum operator (Fonck, 1993), pa_i is the set of direct parents for variable x_i.

Definition 2.2 *Possibilistic causal network*
Possibilistic belief networks are belief networks where the underlying uncertainty distribution is the possibility distribution defined on corresponding to the graph.

A belief network, then, represents the conditional independence relations that exist in a given domain. Now, conditional independence is a relationship between variables or groups of variables that has the following properties (Pearl & Paz, 1985):

1. Trivial independence: $I(X|Z|\emptyset)$
2. Symmetry: $I(X|Z|Y) \Rightarrow I(Y|Z|X)$
3. Decomposition: $I(X|Z|Y \cup W) \Rightarrow I(X|Z|Y)$
4. Weak Union: $I(X|Z|Y \cup W) \Rightarrow I(X|Z \cup Y|W)$
5. Contraction: $I(X|Z|Y) \wedge I(X|Z \cup Y|W) \Rightarrow I(X|Z|Y \cup W)$
6. Intersection: $I(X|Z \cup W|Y \cup W) \wedge I(X|Z \cup Y|W \cup W) \Rightarrow I(X|Z|Y \cup W)$

This characterization of conditional independence is as abstract as possible, thus, it makes no assumption about any particular uncertainty formalism used in order to recognize a given relationship as being an instance of a conditional independence relationship. Now, in learning from data, one has to define an operational criterion for identifyng such relations from summarized information as uncertainty distributions are. We will not review here the various techniques used in probability to detect such relations, the χ^2 test and its variations being the most classical ones.

Our interest lies in defining a criterion for working with possibility distributions derived from data. It will allow us to infer, from the relations between two possibility distributions, wether the corresponding variables are independent or not. As it is the case in probability theory, such criterion rests on the previous notion of *conditional distribution*. Two (or more) variables will be considered as conditionally independent if their conditional distributions obey certain properties. But, while in probability there is a unique formulation for such conditional distributions, several different definitions have been proposed for possibilistic conditioning. We will just give them and then discuss several definitions for independence between variables.

- **Dempster conditioning**(Dempster, 1967)
  It is an specialization of Dempster's rule of conditioning for evidence theory. Given two variables X and Y taking values in $\{x_1, x_2, \ldots, x_n\}$ and $\{y_1, y_2, \ldots, y_n\}$, respectively and the corresponding joint possibility $\pi(x, y)$ distribution the conditional distribution $\pi(x|y)$ is defined as:

$$\pi(X|Y) = \frac{\pi(X,Y)}{\pi_Y(Y)}$$

  where $\pi_Y(Y) = max_{y \in Y}\{\pi(X,Y)\}$

- **Hisdal/Dubois conditioning** (Hisdal, 1978; Dubois, 1986)
  In the same conditions as before:

$$\pi(X|Y) = \begin{cases} \pi(X,Y) & \text{if } \pi(X|Y) < \pi(Y) \\ 1 & \text{otherwise} \end{cases}$$

Now, independence between variables, as we indicated, will require some kind of comparison between their distributions (marginal and conditional), so one or the other of the above conditioning operators will be used in establishing independence. However, at a more abstract level, possibilistic independence between variables or groups of variables can be understood in terms of *mutual information relevance*.

Fonck [Fonck 1994] adheres to this view, putting forth the following interpretation:

- **Conditional independence as mutual information irrelevance** Given three sets of variables X, Y, Z saying that X is independent of Y given Z amounts to the assertion: once the values of Z are known further information about Y is *irrelevant* to X *and* further information about X is *irrelevant* to Y. Given the sets X, Y, Z the independence relation $I(X|Y|Z)$ is true iff

$$\pi^c_{\{X|Y \cup Z\}} = \pi^c_{\{X|Z\}} \text{ and } \pi^c_{\{Y|X \cup Z\}} = \pi^c_{\{Y|Z\}}$$

  is true, where π^c is the distribution that results from applying the c combination operator (i.e. a norm or the corresponding conorm).

This definition is stricter than the one that had been taken previously as a test for independence in possibilisitic settings: non-interactivity. *Non-interactivity* (Zadeh, 1978), means equality between marginal distributions and factored marginal distributions. Fonck has proven that this definition does not obbey all the independence axioms mentioned before but hers does (Fonck, 1995).

Another similar line of work is followed by Huete (Huete, 1995)who explores three different views on independence.

1. **Independence as no change in information**
   When the value of variable Z is known knowing variable Y does not change information about values of X. This can be understood as information about Y being irrelevant for X when Z is known. Note that this a less strict definition than Fonck's.

2. **Independence as no information gain**
   When the value of variable Z is known, knowing variable Y brings no additional information about the values of X. In other words, conditioning represents no information gain.

3. **Independence as similar information**
   When the value of variable Z is known, knowing variable Y brings a similar information about the values of X, this information being *similar* to the one that refered to X before knowing the value of Y.

These three notions of independence are studied by Huete using Hisdal's and Dempster's conditioning operators. The interested reader is referred to (Huete, 1995).

3. Measuring dependence through similarity between distributions

The different interpretations of independence that we have commented above do not reflect completely separated concepts. In fact, they can complement each other. We have adopted an independence characterization based on similarity but that has some relation to information relevance.

Independence between X and Y can be related to the similarity between the marginal possibility distribution $\pi(X)$ and the conditional distribution obtained after conditioning on Y, $\pi(X|Y)$. Extending to the three-variable case

$$I(X|Z|Y) \longleftrightarrow \pi_c(x|yz) \approx \pi_c(x|z) \forall x, y, z$$

where π_c is the distribution obtained by applying one of the usual conditioning operators for possibility distributions.

Again, similarity between distributions admits several definitions. Let us suppose in the following that two distributions, π and π' are being compared. These are the current similarity definitions used (Huete, 1995):

- *Iso-ordering*:
  $$\pi \approx \pi^{'} \Leftrightarrow \forall x, x^{'}[\pi(x) < \pi(x^{'}) \Leftrightarrow \pi^{'}(x) < \pi^{'}(x^{'})]$$
  This amounts to establishing that two distributions are similar if their appropriate possibility distributions exhibit the same ordering for $\pi(x)$, for all values.

- α_0-*equality*: Two distributions will be taken as similar if $\pi(x) = \pi'(x)$ for all x and for all values of $\pi(x)$, $\pi'(x)$ that are greater than a fixed possibility value α_0.
  $$\pi \approx \pi^{'} \Leftrightarrow C(\pi, \alpha) = C(\pi^{'}, \alpha), \forall \alpha \geq \alpha_0$$
  where $C(\pi, \alpha)$ is the α-cut set corresponding to the value α.[1]

- *strict similitude*: In this case the idea is that two distributions are similar if the values for each x for $\pi(x)$ and $\pi'(x)$ differ in less than a given value α.

1. The α -cut set is the set $\{x|\pi(x) \geq \alpha\}$ for $\alpha \in [0, 1]$

Now we have to remark some important aspects of the above definitions of similarity. Firstly, all them are extremely fragile. It is enough for a single value not to obey the definition to rule out two distributions as being not similar. This is not very practical nor realistic when working with distributions derived from data. Secondly, and associated with the first disadvantge, it has to be remarked that there is no *degree of similarity*. Distributions are either similar or not similar. However, it is not the same thing if the differing values in the two distributions are separated by a great difference in possibility or by a small one. Thirdly, and this disadvange has implications for learning, as there is no degree of similarity, comparison of dependence strength between two variables respect to a third one is impossible.

We just would like to combine the information gain approaches of the information relevance definitions of independence with the similarity approach. We will define a *graded similarity* measure that will allow for small variations in the form of distributions and that will also take into account how much each different value of a given variable contributes in making the overall distribution different from the one that is compared against.

The rationale of our definition is the following. Given a *difference value* α in [0,1] two distributions π and π' defined on the same domain will be considered similar if, for the *most part* of the x_i values of their domain the appropriate possibility values $\pi(x_i)$ and $\pi'(x_i)$ differ by less than α

Definition 3.1 *α-set*
Given two possibility distributions π and π' over a domain X and a real number $\alpha \in$ [0,1] the α-set for π and π' in the domain X is defined as:

$$\alpha\text{-}set = \{x_i \in X \text{ such that } |\pi(x_i) - \pi'(x_i)| \leq \alpha$$

Definition 3.2 *Similarity degree.*
Given two possibility distributions π and π' over a domain X and a real number $\alpha \in$ [0,1] their degree of similiarity is defined as:

$$Sim(\pi, \pi', \alpha) = \frac{\sum_{x_i \in \alpha - set} |\pi(x_i) - \pi'(x_i)|}{\sum_{x \in X} |\pi(x_i) - \pi'(x_i)|}$$

If two possibility distributions have a similarity degree $Sim(\pi, \pi', \alpha) = \gamma$ then if $\gamma = 0$ they are said to be *dissimilar* at α level; if $\gamma = 1$ they are said to be *identical* at α level [2]. In any other case, they are said to have similarity γ at α level.

Definition 3.3 *α_{min}*
Given two possibility distributions π and π' over a domain X and the set $\{\alpha_i$ such that $Sim(\pi, \pi', \alpha_i) \neq 0\}$ then α_{min} is the infimum of this set.

Definition 3.4 *Maximally similar distributions*
Two possibiltiy distributions π, π' are said to be maximally similar if $\alpha_{min} = 0$ for them.

2. Then our definition reduces to the second one given above

Now that we have defined similarity in terms of proportion of x_i values that are close to a difference of α in their possibility values, we can establish dependence conditions on variables represented by possibility distributions.

This has the advantage of building an ordering on the strength of association between several variables, a possibility that is very useful in learning DAGs. Given three variables x, y y', we want to define a function Dep_α that will allow us to test whether $Dep_\alpha(x|y) \geq Dep_\alpha(x|y')$ or not at the same α level.

Let us suppose that we have three variables x, y and y'. If there are the same number of values differing in $\pi(x|y)$ and $\pi(x|y')$ then we must test wich of the two conditional distributions changes more the distribution $\pi(x)$. The variable which changes it more will be the one that is more dependent with the one we are testing it against. Of course, in measuring this change one has to take into account the difference in possibility values for each x_i but such difference, due to the influence of the y_i value has to be weighted by the corresponding possibility $\pi(y_i)$. Remember that we are working with possibility distributions derived from data. Now, this will give us an aggregated idea of how much does a given variable y influence variable x in front of the influence of variable y'.

Definition 3.5 *Conditional Dependence Degree*
Given two variables x and y with joint possibility distribution $\pi(x,y)$, marginal possibility distributions π_x and π_y, conditional possibility distribution $\pi_{x|y}$ and a real value α in [0,1] we define their conditional dependence degree as

$$Dep(x,y,\alpha) = 1 - \sum_{y_i \in Y} \pi(y_i) \sum_{x_i \in \alpha - set} |\pi(x_i) - \pi'(x_i|y_i)|$$

Notice that $Dep_\alpha(x,y)$ is greater when $Sim(x,y,\alpha) < Sim(x,y',\alpha)$.

4. A Learning method based on possibilistic conditional dependence degrees

Now that we are in a position to test the degree of dependence between two variables x and y by means of the similarity between their distributions we can use this information to guide the building of a DAG that represents the dependences between the variables in a database.

In doing, so we will resort to dependence degrees to establish an order between variables. This is only a first phase of our method. Building DAGs using coditional dependence information must be complemented with information about the whole quality of the resulting DAG.

In the case of probabilistic belief networks, several measures have been defined in order to assess the quality of the network. For example, a typical one is measuring the cross-entropy of the distribution induced by the network and the distribution underlying the database. Chow and Liu (Chow & Liu, 1968) defined a measure that minimized cross-entropy when the DAG was a tree. This idea has been used by several authors to develop CI-test mehods. See (Rebane & Pearl, 1989; Huete & De Campos, 1993) and in a different setting (Lam & Bacchus, 1994, 1993). Other oftenly used measure is overall entropy of the DAG: one

searches among a space of low entropy networks. A method representing such orientation is (Herskovitz & Cooper, 1990).

In possibility theory, Non-specificity is the concept corresponding to entropy in probability. Gebhard and Kruse [Gebhard and Kruse 1995] defined an overall measure of non-specificity in order to select at any step in their method a variable that, once added and linked to the DAG, resulted in the most specific joint distribution, given the data. Given those DAGs and the data it is important to recover the one that has an minimum overall nonspecificity. That is, we are interested in recovering the DAG that is more precise given the data.

One measure of the non-specificity associated with a possibility distribution is U-uncertainty (Klir & Folger, 1988).

Definition 4.1 *U-uncertainty*
Given a variable X with domain $\{x_1 \ldots x_n\}$ and an associated possibility distribution $\pi_x(x_i)$ the U-uncertainty for $\pi(x)$ is:

$$U(\pi(x)) = \int_0^1 \lg_2 card(X_\rho) d\rho$$

where X_ρ is the ρ cut for X. That is, $X_\rho = \{\ x_i$ such that $\pi(x_i) \geq \rho\ \}$.

U-uncertainty can be extended for joint and conditional distributions in the following way:

Definition 4.2 *Joint U-uncertainty*

Given a variable $X_1 \ldots X_n$ variables with associated possibility distributions $\pi_{X_1} \ldots \pi_{X_n}$ their joint nonspecificity measured as U-uncertainty is:

$$U(\pi_{X_1} \ldots \pi_{X_n})) = \int_0^1 \lg_2 card(X_{1_\rho} \times \ldots X_{n_\rho}) d\rho$$

Definition 4.3 *Conditional U-uncertainty*
Given two variables X, Y with associated possibility distributions π_X, π_Y their conditional nonspecifcity measured as conditional U-uncertainty is:

$$U(\pi_X(x)|\pi_Y(y)) = \int_0^1 \lg_2 \frac{card(X_\rho \times Y_\rho)}{card(Y_\rho)} d\rho$$

Note that $U(X|Y) = U(X,Y) - U(Y)$

Now, we are interested in finding the overall U-uncertainthy of a given DAG. That is, the U-uncertainty of the joint possibility distribution induced by the DAG. Making use of the factorizing property of belief networks, we can define the *Global non-specificity* for a given DAG. First we need a previous definition that of the nonspecificity due to the conditional distribution of a variable and its parents.

Definition 4.4 *Parent-children non-specificity*
Let G be a DAG representing the conditional independence relationships existing between the variables in a domain $U = \{x_1 \dots x_n\}$. For any given variable x_i with parent set pa_i, the parent-children non-specificity is:

$$U(x_i|pa_i) = U(x_i, pa_i) - U(pa_i)$$

when $pa_i = \emptyset$ then $U(x_i|pa_i) = U(x_i)$

Definition 4.5 *DAG non-specificity*
For a given DAG G defined on the same domain as in the previous case the DAG non-specificity is:

$$U(G) = \sum_{x_i \in U} U(x_i|pa_i)$$

Now, the space of possible DAGs is enormous, so information about known dependencies can help in pruning it. The idea is to use dependence information to build a non-oriented DAG and then select the best orientations by means of the non-specificity of the graph. POSSCAUSE starts with the null DAG and at each step selects a variable for inclusion. The selected variable will be the one with highest calculated dependence. The orientation in which it will become part of the DAG will be the one that minimizes non-specificity. It is in this sense that POSSCAUSE is a hybrid method interleaving CI-test methods with a quality measure. We used as a basic method for building graphs (polytrees) from conditional independency methods the algorithm HC devised by Huete Campos and Moral (Huete, 1995), which recovers DAGs by using only zero and first order conditional tests. Then we modified this learning algorithm in order to take into account specific aspects of possibility theory. The resulting algorithm, POSSCAUSE, is based on an earlier version, HCS (Sangüesa, Cabós, & Cortés, 1996) that was developed in order to combine CI-test methods and entropy-based methods.

4.1 The HCS algorithm

Central to our methods is the idea of *variable sheaths* due to Huete, (Huete & De Campos, 1993). A sheath Ψ_{x_i} for variable x_i is the subgraph corresponding to those other variables in U that are direct causes and effects of x_i. Sheaths are obtained by repeatedly expanding the set of variables that are marginally dependent with respect to x_i, i.e., those y_i in U for which $I(x_i|\emptyset|y_i)$ holds. This set is called Λ_{x_i}. In Huete's method, after expansion of Λ_{x_i} for all variables x_i in the domain, a polytree-like DAG is recovered by fusing the resulting partial sheaths Ψ_{x_i}. Finally, and according to polytree properties, orientations for links are introduced. Orientation is not made until the whole graph is built.

There are some aspects that are worth commenting. First, as many other CI-test methods, HC algorithm takes as input a list of existing conditional dependences on U. Secondly, orientation is made after expanding each sheath. And thirdly, after expansion, of Λ_{x_i} *direct* causes and effects involve not only those direct ancestors and succesors of a variable in the DAG but also their their neighbours, i.e., those variables for which successors and predecessors of x_i act as a separating set.

The HCS algorithm is a combined algorithm for the recovery of DAGs, using Huete's method and a measure of quality for network orientation (conditional entropy in the probabilistic case, conditional overall non-specificty for possibility theory).

HCS Algorithm

1. For each x_i in U
   (a) Calculate Λ_{x_i}.
   (b) Calculate Ψ_{x_i}.
   (c) For each y in Ψ_{x_i}
      i. Calculate the set of possible neighbours $N_{x_i}(y)$.
      ii. If $N_{x_i}(y) = \emptyset$ then eliminate y from Ψ_{x_i}
   (d) Create G_{x_i} [3]
      i. For each y in Ψ_{x_i}. If there exists no link between x_i and y then
         A. If x_i is a root node
            .Create graph G_1 by adding to G_1 the link $y \rightarrow x$.
            .Calculate $U(G_1)$
            .Create graph G_2 by adding to G_1 the link $x \rightarrow y$.
            .Calculate $U(G_2)$
            . If $U(G1) > U(G2)$ then $G_{x_i} = G1$ Else G=G2
         B. If x_i is not a root node.
            Then add the link $x \rightarrow y$
2. Merge all G_{x_i} to obtain G.
3. Test wether the resulting graph is simple. If it is not then **FAIL**

Results obtained by applying the HCS algorithm are commented in section 5.

4.2 The POSSCAUSE system

We set ourselves to the task of making such method a bit more independent of initial information about dependences and also, to make it recover more general DAGs. Moreover, we wanted to use information about the quality of the network in order to decide on the orientation of the links. The idea behind that was that subgraphs based on partial sheaths would involve less nodes and links and then the cost of orientation would be inferior than delaying it to the final non-oriented graph. In addition, we wanted to use a measure, or a combination of measures, that produced a resulting DAG that were accurate (specific) with respect to data but not to the point of being too complex. Huete proposed using Kulblack-Leiber cross-entropy measure in order to select the best orientations but calculations are cumbersome. Moreover such measure tends to favor too precise networks that are more complex than informative. Finally, we wanted to make the resulting algorithm as amenable to parallell computation as possible.

3. The partial graph relating all variables in Ψ_{x_i}

The general schema of the algorithm is as follows. For each variable x_i in U find its corresponding Λ_{x_i} build the *reduced sheath* ρ_{x_i} for it (i.e. only those direct causes that are direct ancestors or predecessors of x_i); orient the reduced sheath by means of non-specificity tests and merge the resulting sheaths for all variables in U.

Definition 4.6 *Reduced sheath*
For a node x_i in a DAG representing the conditional independence relationships in a given domain U, with sheath Φ_{x_i}, the reduced sheath of x_i, ρ_{x_i}, is the set of those vars y in Φ_{x_i} such that $y \in Adj(x_i)$ where $Adj(x_i)$ is the set of variables in the DAG that are adjacent to x_i.

For any couple of variables $\{y, z\}$ $y, z \in \rho_{x_i}$ belonging the following conditions hold:

1. $I(y|x_i|z)$
2. $\neg I(x_i|z)$
3. $\neg I(x_i|y)$

Definition 4.7 *Indirect causes*
Given a variable x_i with sheath Φ_{x_i} and reduced sheath ρ_{x_i}, the set of indirect causes of x_i is $\sigma_{x_i} = \Phi_{x_i} - \rho_{x_i}$ Let us suppose $\sigma_{x_i} = \{y_1 \ldots y_m\}$,then for any y_k, $I(x_i|y_k|z_j)$ for some variable z_j not in the reduced sheath of x_i.

Definition 4.8 *Focus of a sheath*
The focus of a sheath ρ_{x_i} is the variable around which the sheath is built, x_i

In this way, we distinguish between the direct parents and children of a given node x_i and other variables related to x_i through these direct parents and children. These other variables, in turn, may be be the direct parents or children of some other variable in the final DAG.

DAG construction proceeds in parallell. The idea is to find the direct parents and children of each variable, then orient this reduced sheath and then merge all oriented sheaths. There exists a process for each variable x_i in the domain. Each one builds the reduced sheath for x_i. During this process some variables $\{y_{i_1} \ldots y_{i_m}\}$ will be detected as dependent with x_i but that mediate between x_i and some other variables $\{z_{i_1} \ldots z_{i_m}\}$. That is, for each y_{i_k} the relation $I(x_i|y_{i_k}|z_i)$, holds. Evidently, no z_i can belong to the reduced sheath of x_i.The processes that are building the reduced sheath of the variables $\{z_{i_1} \ldots z_{i_m}\}$ must know that $\{y_{i_1} \ldots y_{i_m}\}$ belong to their reduced sheaths.

Now a method can be devised in order to recover a possibilistic DAG from data.

- **Input**: DB, a database on a domain $U = \{x_1 \ldots x_n\}$
- **Output**: the minimum nonspecificity possibilistic DAG, D_{min} compatible with DB or an error message

1. **For each** x in U
   (a) Build the set of marginal dependent variables for x, Λ_x

(b) Build the set of direct causes and effects for x. ρ_x

(c) Orient each ρ_x according to the minimum non-specificity alternative

2. Create D_{min}, the graph resulting from joining all minimum nonspecificty ρ_x.

3. **If** there are cycles in D_{min} **then FAIL**
else return D_{min}

Deriving Λ_{x_i} for each x_i amounts to calculating the Dep_α values for the rest of the variables in the domain. The result is a symmetrical table. This task is done in parallell with no special difficulty.

Now we will see how orientation testing (step 1-c) can be done. First, we have a variable sheath that basically represents the skeleton of a subgraph. That is, a subgraph with no orientation. Orienting such structure reduces to finding the most plausible parents and children of the focus of the sheath, x_i. In fact, while doing so, several shortcuts can be applied. As every triplet $\{x, y, z\}$ in ρ_x obeys the conditions in definition (2) it is enough to test only three orientations: $y \rightarrow x \rightarrow z$, $y \leftarrow x \leftarrow z$ and $y \leftarrow x \rightarrow z$.

Orientation step

- **Input**: a non-oriented reduced sheath for a variable x_i in U, ρ_{x_i}
- **Output**: the minimum non-specificity oriented subgraph corresponding to the subgraph ρ_{x_i}, $D_{\rho_{x_i}}$
- Let $\rho x_i = \emptyset$
- For each y, z in ρ_{x_i}
  1. Find the miminum nonspecificity configuration min_{pc} of the set $\{y \rightarrow x_i \rightarrow z, y \leftarrow x_i \rightarrow z, y \leftarrow x_i \leftarrow z\}$
  2. Let $result = result \cup min_{pc}$

Finding the minimum non-specificity parent-children set of x_i is equivalent to testing for each pair of variables y, z in ρ_{x_i} which of the three above mentioned orientations reduces in a greater amount the accumulated non-specificity.

Currently, the POSSCAUSE algorithm has been implemented on a Sun workstation simulating parallel processes. It is currently being ported to a parallel IBM-SP2 computer under PVM-E software. The system allows for several modifications of the above mentioned algorithms. For example, information about known dependences can be entered by an expert. If evidence against them is not conclusive, they are accepted and are used as a guide in building the variables' sheaths. We will comment on that in the next section. The system is divided into a dependence calculation module, an input/output module and a graphical interface. Generated DAGs are stored using the preliminary version of Microsoft Standard Format for Belief Network storage.

5. Experimental results

The algorithm HCS has been applied to two artificial databases: the first one is taken from Musick's work on Belief Network Induction (Musick, 1994) and the second one is the well-known LED database from the UCI Machine Learning Database Repository (Murphy & Aha, 1996). We will comment then in turn.

Musick's database is a small example that is represented as a simple DAG on five variables. The corresponding database contains 100 cases and the original dependence relations between them are depicted in figure 5.

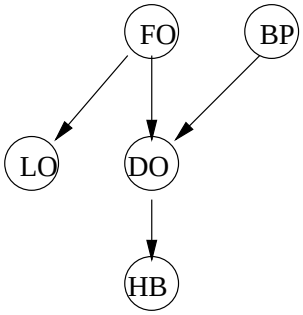

Figure 1: Musick's 'dog out' example.
BP: Bowel Problems, DO: Dog Out, LO: Lights Out,FO: Family Out,HB: Hear Barking

The LED database was used by Fung and Crawford to test their Markov network construction program CONSTRUCTOR (Fung & Crawford, 1990). It represents an array of seven LEDs that are commanded to be on a off by selecting the number of the LED. The arangement is faulty: the state of the first LED depends on the state of the selector conditionally to the state of the LEDS 2 and 3. So the dependency structure of the problem is the one that appears in 5.2.

5.1 Musick's database and HCS

This database was tested with different transformations from probability to possibility and also with different dependence γ_{cut} degrees.The best results were obtained by using the minimum information loss probability-possibility transformation (Dubois, Prade, & Sandri, 1991). Establishing a limit of .08 for defining the level of dependence, a very good approximation to the original structure was found. Only variable LO appeared as a parent of DO in the most part of examples. This occurred over different dependence levels. In fact, it is a consequence of the fusion algorithm that makes no use of the fact that some variables should not be merged because the corresponding subgraphs include variables the are not very highly related with the variables in the other subgraph. In such cases, subgraphs should not be connected through this variable or an alternative orientation should be searched using the non-specificity criterion.

Other results showed changes in the parent-child relation between variables in the same subgraph. This may be due to the different dependency levels used. When not enough evidence is found for orientation a non-specificity test was used.

When using order or dependence relations introduced by the expert, HCS recovered the DAG exactly, as it was to be expected, given the nature of the basic CI-test algorithm used.

It is also interesting to remark that it was sufficient to introduce dependence knowledge for just those variables with too low evidence of association. Notice that those variables gathered not enough evidence because in the data there were insufficient cases to support all possible value combinations, so some degree of incompleteness appeared.

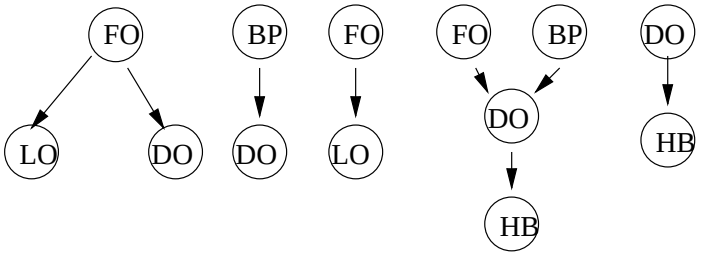

Figure 2: Partial graphs associated to G_{FO}, G_{BP}, G_{LO}, G_{DO} and G_{HB}

In any case it was a remarkable characteristic of the HCS algorithm that all recovered graphs exhibited a closer structure to the original DAGs than the ones recovered with porbabilistic measures, confirming in a sense, that in the presence of insufficient data, possibilistic methods are more robust.

5.2 The LED database and HCS

The LED domain was tested with samples of 100, 1000 and 10000 cases in order to cover as much as possible data values combinations and to give sufficient coverage for statistical χ^2 tests before transformation into possibility distributions. Again, the bests results were obtained by using the minimum information loss transformation to possibility. However, and as it was to be expected, the algorithm was never able to recover the true structure of the domain, because it is not a simm ple DAG: variables 2 and 3 have a common child and also common parents. Many spurious dependencies where detected even lowering the α values and raising the limit for independence.

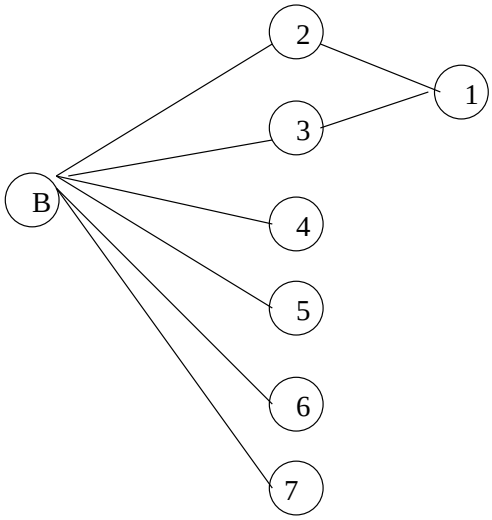

Figure 3: The LED original DAG

5.3 The LED database and POSSCAUSE

The last previously mentioned problem due to the structure of the domain, induced us to extend HCS in order to recover more general DAGs. The algorithm POSSCAUSE implemented such extension. In fact at each step it tries to widen a variable sheath by building a separating set that can include more than one variable. When applying the parallel POSSCAUSE version to LED, we run several tests on databases with 100, 500, 2000 and 20000 cases. There were changes in the results when the number of data increased. But, again, these changes were not as great as the ones induced in the probaiblistic counterpart of the algorithm. It is interesting to note, that, due to data characteristics, even with larger samples, the probability-based version recovered a structure where *all* variables $v1 \ldots v7$ were marginally dependent on $v8$, the one that simulates the LED selector: i.e. the structure corresponding to a *correct* LED display.

6. Conclusions and further research

A measure for similarity between distributions has been the basis for two new hybrid algorithms for causal network construction: HCS and POSSCAUSE. The first one is addressed to recover simple DAGs and the second one to recover general DAGs. When HCS is applied to domains where the underlying structure is a simple dag it recovers faithfully the known dependences of the domain. Contrary to other methods, it needs no information on the order between variables to get such results. Although, there are variations related to order beween the variables. The underlying *dependency model* is correctly recovered whatever the order between input variables is. However, the sheath structure is highly dependent in the order of consideration of variables in the Λ_x set. Huete's CI-test selects variables in the Λ_x set in a fixed order. This bring sthe implication that, in buildign the sheath, vari-

ables wiht less dependency with the sheath focus are included before others that are more strongly associated to it. This results in keeping dependencies that may induce incorrect parent-child link associations. We are testing the effect of selecting first for inclusion the ones that are more dependent. As we mentioned, i n case knowledge is available it also can use it in the form of dependence information, in which case it recovers the true dependencies in the domain. It is important to see that complexity stays at the same order than the original CI-test algorithm HCS is based upon (Huete, 1995) but orientation, based on non-specificity test (or their conditional entropy counterparts for probability theory), allow for a reduction in the cost of the orientation step due in part to the fact that subgraphs are first oriented and then merged. New heuristics, however are under study in order to decide on orientations between subgraphs in order to avoid introducing spurious associations in this step.

The second algorithm, POSSCAUSE, is a parallel extension of HCS and it is able to recover more general DAGs. Here recovered DAGs are able to reconstruct correct domain dependencies even when the underlying dependency model is not a simple DAG. Parallelization introduces an increase in speed of 10 to 30 as it happens with other parallel version of other cuasl netwrok construction algorithms (Singh & Valtorta, 1993). Again, spurious associations appeared in interleaving the orientation and fusing steps, and as it is the case with HCS, heuristics are being investigated.

A very important issue, however, remains to be dealt with. It is referred to the causal nature of the links involved in the recovery process (Sangüesa, 1996). In effect,there is a widespread identification of belief networks with causal networks. This may be too a rapid identification. It may be true for causal networks built directly by experts. Humans tend to think in terms of clusters of causally related variables. It happens that, when asked to build a belief network, experts tend to link causes and effects into the DAG and then elucidating the corresponding uncertainty distributions. So, all cause-effect relationships are close to the conditional independence interpretation on DAGs, but the inverse relationship is not all true as Drudzel and Simon (Drudzel & Simon, 1993)argue. There has been a lot of controversy about how to identify 'true' causal relationships from conditional independence information. As a result there exists a trend in formalizing correct axioms for causal relevance (Galles & Pearl, 1996, 1995) an which allow for identifying true causal links in a DAG built by means of conditional independence relationships.

Our next steps will be to test if Galles and Pearl axioms hold in a possibilistic setting and then create a critiquing module for the POSSCAUSE system in order to refine the obtained networks.

References

Cano, J., Delgado, M., & Moral, S. (1993). An axiomatic framework for the propagation of uncertainty in directed acyclic graphs. *International Journal of Approximate Reasoning*, *8*, 253–280.

Chow, C., & Liu, C. (1968). Approximating discrete probability distributions with dependence trees.. *IEEE Transactions on Information Theory*, *14*, 462–467.

Cooper, G., & Herskovitz, E. (1992). A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, *9*, 320–347.

De Campos, L., & Huete, J. (1993). Learning non-probabilistic belief networks. In *Proceedings of the second European Conference on Quantiative and Symbolyc Approaches to Reasoning under Unceratinty.*

Dempster, A. (1967). Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematics and Statistics*, *38*, 315–329.

Drudzel, M., & Simon, H. (1993). Causality in bayesian belief. In *Proceedings of the Ninth Conference on Uncertainty in Artifical Intelligence*, pp. 3–11 San Mateo, CA. Morgan Kaufmann, San Mateo, CA.

Dubois, D. (1986). Belief strucutures, possibility theory and decomposable confidence measures on finite sets.. *Computers and Artificial Intelligence*, *5*(5), 403–417.

Dubois, D., & H, P. (1986). *Théorie des possibilités. Application á la représentation des connaissances en informatique.* Masson, Paris.

Dubois, D., & Prade, H. (1990). Inference in possibilistic hypergraphs. In *Proceedings of the third IPMU Conference*, pp. 250–259.

Dubois, D., Prade, H., & Sandri, S. (1991). On possibility/probability transformations. In Lowen, R., & Roubens, M. (Eds.), *Proceedings of the fourth International Fuzzy System Association Congress, IFSA91*, pp. 50–53.

Fonck, P. (1991). Influence networks in possibility theory. In *Proceedings of the second DRUMS R.P. Group Workshop* Albi, France.

Fonck, P. (1992). Propaganting uncertainty in directed acyclic graphs. In *Proceedings of the fourth IPMU Conference* Mallorca.

Fonck, P. (1993). *Reseaux d'inference pour le raisonnement possibiliste.* Ph.D. thesis, Université de Liege.

Fonck, P. (1995). Conditional independence in possibility theory. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pp. 221–226.

Fung, R., & Crawford, S. (1990). Constructor: A system for the induction of probabilistic models. In *Proceedings of AAAI-90*, pp. 762–765 Boston. MIT Press.

Galles, D., & Pearl, J. (1995). Testing identifiability of causal effects. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pp. 185–195. Morgan Kaufmann, San Mateo, CA.

Galles, D., & Pearl, J. (1996). Axioms for causal relevance. Tech. rep., Cognitive Systems Laboratory, University of California, Los Angeles.

Gebhardt, J., & Kruse, R. (1993). The context model: An integrating view of vagueness and uncertainty. *International Journal of Approximate Reasoning*, *9*, 283–314.

Gebhardt, J., & Kruse, R. (1995). Learning possibilistic networks from data. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics* Fort Lauderdale, FL.

Heckerman, D. (1995). A bayesian approach to learning causal networks. Tech. rep. MSR-TR-95-04, Microsoft Research Advanced Technology Division.

Herskovitz, E., & Cooper, G. (1990). Kutató: an entropy-driven system for the construction of probabilistic expert systems from data. In *Proceedings of the sixth conference on Uncertainty in Artificial Intelligence.*

Hisdal, E. (1978). Conditional possibilities, independence and non-interaction. *Fuzzy Sets and Systems*, *1*, 283–297.

Huete, J. (1995). *Aprendizaje de redes de creencia mediante la detección de independencias: modelos no probabilisticos.* Ph.D. thesis, Universidad de Granada, Granada.

Huete, J., & De Campos, L. (1993). Learning causal polytrees. In Kruse, R., & Clarke, M. (Eds.), *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, Vol. Lecture Notes in Computer Science 747. Springer Verlag.

Josslyn, C. (1994). *Possibilistic Process for Complex System Modelling.* Ph.D. thesis, State University of New York at Binghampton, New York.

Klir, G., & Folger, T. (1988). *Fuzzy Sets, Uncertainty and Information.* Prentice Hall, Englewood Cliffs, NJ.

Lam, W., & Bacchus, F. (1993). Using causal information and local measures to learn bayesian belief networks. In *Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence*, pp. 243–250.

Lam, W., & Bacchus, F. (1994). Using new data to refine a bayesian network. In *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, pp. 383–390.

Murphy, P., & Aha, D. (1996). Uci repository of machine learning databases. machine-readable data repository. Dept. of Information and Computer Science, University of California, Irvine.

Musick, C. (1994). *Belief Network Induction.* Ph.D. thesis, University of California at Berkeley.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, San Mateo, CA.

Pearl, J. (1993). Belief networks revisited. *Artificial Intelligence*, *59*, 49–56.

Pearl, J. (1994). Bayesian networks. Tech. rep. R-216, Computer Science Department, University of California, Los Angeles.

Pearl, J., & Paz, A. (1985). Graphoids: a graph-based logic for reasoning about relevance relations. Tech. rep., Cognitive Science Laboratory, Computer Science Department, University of California, Los Angeles.

Pearl, J., & Verma, T. (1991). A theory of inferred causation. In *Proceedings of the Second International Conference on Knowledge Representation and Reasoning*. Morgan Kaufmann, San Mateo, CA.

Piatetsky-Shapiro, G., & Frawley, W. (Eds.). (1991). *Knowledge Discovery in Databases*. AAAI Press. Menlo Park, Ca.

Rebane, T., & Pearl, J. (1989). The recovery of causal poly-trees from statistical data. In Kanal, L., Levitt, T., & Lemmer, J. (Eds.), *Uncertainty in Artificial Intelligence*, Vol. 3. North-Holland, Amsterdam.

Sangüesa, R. (1996). Learning causal networks from data; a survey. Tech. rep. LSI-96-19-R, Dept. de Llenguatges i Sistemes Informàtics, Universitat Politécnica de Catalunya, Barcelona.

Sangüesa, R., Cabós, J., & Cortés, U. (1996). Experimentación con métodos hibridos de aprendizaje de redes bayesianas. Tech. rep. LSI-96-R, Dept. de Llenguatges i Sistemes Informátics, Universitat Politécnica de Catalunya, Barcelona.

Shenoy, P. (1991). Independence in valuation-based systems. Tech. rep. Working Paper 236, Univesity of Kansas.

Singh, M., & Valtorta, M. (1993). An algorithm for the construction of bayesian network structures from data. In *Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence*, pp. 259–265. Morgan Kaufmann.

Singh, M., & Valtorta, M. (1995). Construction of bayesian network structures from data: A survey and an efficient algorithm. *International Journal of Approximate Reasoning*, *12*, 111–131.

Verma, T., & Pearl, J. (1992). An algorithm for deciding if a set of observed independencies has a causal explanation. In *Proceedings of the Eighth Conference on Uncertainty in Artificial Intelligence*, pp. 323–330. Morgan Kaufmann.

Zadeh, L. (1978). Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, *12*(1), 3–28.