Acyclic directed graphs to represent conditional independence models

Marco Baioletti[1] Giuseppe Busanello[2] Barbara Vantaggi[2]

1 Dip. Matematica e Informatica, Università di Perugia, Italy,
e-mail: baioletti@dipmat.unipg.it
2 Dip. Metodi e Modelli Matematici, Università "La Sapienza" Roma, Italy,
e-mail: {busanello, vantaggi}@dmmm.uniroma1.it

Abstract. In this paper we consider conditional independence models closed under graphoid properties. We investigate their representation by means of acyclic directed graphs (DAG). A new algorithm to build a DAG, given an ordering among random variables, is described and peculiarities and advantages of this approach are discussed. Finally, some properties ensuring the existence of perfect maps are provided. These conditions can be used to define a procedure able to find a perfect map for some classes of independence models.

Key words: Conditional independence models, Graphoid properties, Inferential rules, Acyclic directed graphs, Perfect map.

1 Introduction

Graphical models [6, 8–10, 12, 15] play a fundamental role in probability and statistics and they have been deeply developed as a tool for representing conditional independence models. It is well known (see for instance [6]) that, under the classical definition, the independence model $\mathcal{M}$ associated to any probability measure P is a semi–graphoid and, if P is strictly positive, $\mathcal{M}$ is a graphoid. On the other hand, an alternative definition of independence (a reinforcement of cs–independence [4, 5]), which avoids the well known critical situations related to 0 and 1 evalutations, induces independence models closed under graphoid properties [13].

In this paper the attention is focusing on graphoid structures and we consider a set J of conditional independence statements, compatible with a (conditional) probability, and its closure $\bar{J}$ with respect to graphoid properties. Since the computation of $\bar{J}$ is infeasible (its size is exponentially larger than the size of J), then, as shown in [10, 11, 1], we will use a suitable set J_* of independence statements (obviously included in $\bar{J}$), that we call "fast closure", from which it is easy to verify whether a given relation is implied, i.e. whether a given relation belongs to $\bar{J}$. Some of the main properties of fast closure will be described.

The fast closure is also relevant for building the relevant acyclic directed graph (DAG), which is able to concisely represent the independence model. In fact we will define the procedure BN-DRAW which builds, starting from this set and an ordering on the random variables, the corresponding independence map. The main difference between BN-DRAW and the classical procedures (see e.g. [7, 9]) is that the relevant DAG is built without referring to the whole closure.

Finally, we give a condition assuring the existence of a perfect map, i.e. a DAG able to represent all the independence statements of a given independence model. By using this result it is possible to define a correct but incomplete method to find a perfect map. First, a suitable ordering, satisfying this condition, is searched by means of a backtracking procedure. If such an ordering exists, a perfect map for the independence model can be found by using the procedure BN-DRAW.

Since the above condition is not necessary, but only sufficient, as shown in Example 3, such condition can fail even if a perfect map exists. The provided result is a first step to look for a characterization of orderings giving rise to perfect maps.

2 Graphoid

Throughout the paper the symbol $\tilde{S} = \{Y_1, \dots, Y_n\}$ denotes a finite not empty set of variables. Given a probability P, a conditional independence statement $Y_A \perp\!\!\!\perp Y_B | Y_C$ (compatible with P), where A, B, C are disjoint subsets of the set $S = \{1, \dots, n\}$ of indices associated to $\tilde{S}$, is simply denoted by the ordered triple (A, B, C). Furthermore, $S^{(3)}$ is the set of all ordered triples (A, B, C) of disjoint subsets of S, such that A and B are not empty. A conditional independence model $\mathcal{I}$, related to a probability P, is a subset of $S^{(3)}$. As recalled in the introduction we refer to probabilistic independence models even if the results are valid for any graphoid structure.

We recall that a graphoid is a couple $(S, \mathcal{I})$, with $\mathcal{I}$ a ternary relation on the set $S^{(3)}$, satisfying the following properties:

- G1 if $(A, B, C) \in \mathcal{I}$, then $(B, A, C) \in \mathcal{I}$ (Symmetry);
- G2 if $(A, B, C) \in \mathcal{I}$, then $(A, B', C) \in \mathcal{I}$ for any nonempty subset B' of B (Decomposition);
- G3 if $(A, B_1 \cup B_2, C) \in \mathcal{I}$ with B_1 and B_2 disjoint, then $(A, B_1, C \cup B_2) \in \mathcal{I}$ (Weak Union);
- G4 if $(A, B, C \cup D) \in \mathcal{I}$ and $(A, C, D) \in \mathcal{I}$, then $(A, B \cup C, D) \in \mathcal{I}$ (Contraction);
- G5 if $(A, B, C \cup D) \in \mathcal{I}$ and $(A, C, B \cup D) \in \mathcal{I}$, then $(A, B \cup C, D) \in \mathcal{I}$ (Intersection).

A semi–graphoid is a couple $(S, \mathcal{I})$ satisfying only the properties G1–G4.
The symmetric version of rules G2 and G3 will be denoted by

- G2s if $(A, B, C) \in \mathcal{I}$, then $(A', B, C) \in \mathcal{I}$ for any nonempty subset A' of A;
- G3s if $(A_1 \cup A_2, B, C) \in \mathcal{I}$, then $(A_1, B, C \cup A_2) \in \mathcal{I}$.

3 Generalized inference rules

Given a set J of conditional independence statements compatible with a probability, a relevant problem about graphoids is to find, in an efficient way, the closure of J with respect to G1–G5

$$\bar{J} = \{\theta \in S^{(3)} : \theta \text{ is obtained from } J \text{ by } G1 - G5\} .$$

A related problem, called implication, concerns to establish whether a triple $\theta \in S^{(3)}$ can be derived from J, see [16].

It is clear that the implication problem can be easily solved once the closure has been computed. But, the computation of the closure is infeasible because its size is exponentially larger than the size of J. In [1–3] we describe how it is possible to compute a smaller set of triples having the same information as the closure. The same problem has been already faced successfully in [11], with particular attention to semi–graphoid structures.

In the following for a generic triple $\theta_i = (A_i, B_i, C_i)$, the set X_i stands for $(A_i \cup B_i \cup C_i)$.

We recall some definitions and properties introduced and studied in [1–3] useful to efficiently compute the closure of a set of conditional independence statements. Given a pair of triples $\theta_1, \theta_2 \in S^{(3)}$ we say that θ_1 is *generalized–included* in θ_2 (briefly g–included), in symbol $\theta_1 \sqsubseteq \theta_2$, if θ_1 can be obtained from θ_2 by a finite number of applications of G1, G2 and G3.

Proposition 1. *Given $\theta_1 = (A_1, B_1, C_1)$ and $\theta_2 = (A_2, B_2, C_2)$, then $\theta_1 \sqsubseteq \theta_2$ if and only if the following conditions hold*

(i) $C_2 \subseteq C_1 \subseteq X_2$;
(ii) either $A_1 \subseteq A_2$ *and* $B_1 \subseteq B_2$ *or* $A_1 \subseteq B_2$ *and* $B_1 \subseteq A_2$.

Generalized inclusion is strictly related to the concept of dominance $\sqsubseteq_a$ on $S^{(3)}$, already defined in [10, 11]. We say $\theta_1 \sqsubseteq_a \theta_2$ if θ_1 can be obtained from θ_2 with a finite number of applications of G2, G3, G2s and G3s.

Therefore it is easy to see that $\theta' \sqsubseteq \theta$ if and only if either $\theta' \sqsubseteq_a \theta$ or $\theta' \sqsubseteq_a \theta^T$ where θ^T is the transpose of θ ($\theta^T = (B, A, C)$ if $\theta = (A, B, C)$).

The definition of g–inclusion between triples can be extended to sets of triples and its properties are showed in [2, 3].

Definition 1. *Let H and J be subsets of $S^{(3)}$. J is a covering of H (in symbol $H \sqsubseteq J$) if and only if for any triple $\theta \in H$ there exists a triple $\theta' \in J$ such that $\theta \sqsubseteq \theta'$.*

3.1 Closure through one generalized rule

The target of [10, 2, 3] is to find a fast method to compute a reduced (with respect to g–inclusion $\sqsubseteq$) set J^* bearing the same information of $\bar{J}$, that is for any triple $\theta \in \bar{J}$ there exists a triple $\theta' \in J^*$ such that $\theta \sqsubseteq \theta'$.

Therefore, the computation of J^* provides a simple solution to the implication problem for J. The strategy to compute J^* is to use a generalized version of the remaining graphoid rules G4, G5 and their symmetric ones (see also [11]).

These two inference rules are called generalized contraction (G4*) and generalized intersection (G5*). The rule G4* allows to deduce from θ_1, θ_2 the greatest (with respect to $\sqsubseteq$) triple τ which derives from the application of G4 to all the possible pairs of triples θ_1', θ_2' such that $\theta_1' \sqsubseteq \theta_1$ and $\theta_2' \sqsubseteq \theta_2$. The rule G5* is analogously defined and it is based on G5 instead of G4.

It is possible to compute the closure of a set J of triples in $S^{(3)}$, with respect to G4* and G5*, that is

$$J^* = \{\tau : J \vdash_G^* \tau\} \tag{1}$$

where $J \vdash_G^* \tau$ means that τ is obtained from J by applying a finite number of times the rules G4* and G5*.

In [2, 3] it is proved that J^*, even if it is smaller, is equivalent to $\bar{J}$ with respect to graphoids, in that $J^* \sqsubseteq \bar{J}$ and $\bar{J} \sqsubseteq J^*$.

A further reduction is to keep only the "maximal" (with respect to g–inclusion) triples of J^*

$$J^*_{/\sqsubseteq} = \{\tau \in J^* : \nexists \bar{\tau} \in J^* \text{ with } \bar{\tau} \neq \tau, \tau^T \text{ such that } \tau \sqsubseteq \bar{\tau}\}.$$

Obviously, $J^*_{/\sqsubseteq} \subseteq J^*$.

In [2, 3] it is proved that $J^* \sqsubseteq J^*_{/\sqsubseteq}$, therefore there is no loss of information by using $J^*_{/\sqsubseteq}$ instead of J^*. Then, given a set J of triples in $S^{(3)}$, we compute the set $J^*_{/\sqsubseteq}$, which we call "fast closure" and denote with J_*.

In [2, 3] it is proved that the fast closure set $\{\theta_1, \theta_2\}_*$ of two triples $\theta_1, \theta_2 \in S^{(3)}$ is formed with at most eleven triples. Furthermore, these triples have a particular structure strictly related to θ_1 and θ_2 and they can be easily computed. By using $\{\theta_1, \theta_2\}_*$, it is possible to define a new inference rule

C : from θ_1, θ_2 deduce any triple $\tau \in \{\theta_1, \theta_2\}_*$.

We denote with J^+ the set of triples obtained from J by applying a finite number of times the rule C. As proved in [2, 3], J^+ is equivalent to $\bar{J}$ with respect to graphoids, that means $J^+ \sqsubseteq \bar{J}$ and $\bar{J} \sqsubseteq J^+$. Obviously, by transitivity J^+ is equivalent to J_*.

Therefore, it is possible to design an algorithm, called FC1, which starts from J and recursively applies the rule C and a procedure, called FindMaximal (which computes $H_{/\sqsubseteq}$ for a given set $H \subseteq S^{(3)}$), until it arrives to a set of triples closed with respect to C and maximal with respect to g–inclusion.

In [2, 3] completeness and correctness of FC1 are proved.

Note that, as confirmed in [2, 3] by some experimental results, FC1 is more efficient than the algorithms based on G4* and G5*.

4 Graphs

In the following, we refer to the usual graph definitions (see e.g. [9]). We denote by $G = (\mathcal{U}, E)$ a graph with set of nodes $\mathcal{U}$ and oriented arcs E formed by ordered pairs of nodes. In particular, we consider directed graphs having no cycles, i.e. acyclic directed graphs (DAG). We denote for any $u \in \mathcal{U}$, as usual, with $pa(u)$ the parents of u, $ch(u)$ the child of u, $ds(u)$ the sets of descendants and $an(u)$ the set of ancestors. We use the convention that each node u belongs to $an(u)$ and to $ds(u)$, but not to $pa(u)$ and $ch(u)$.

Definition 2. *If A, B and C are three disjoint subsets of nodes in a DAG G, then C is said to d–separate A from B, denoted $(A, B, C)_G$, if there is no path between a node in A and a node in B along which the following two conditions hold:*

1. *every node with converging arrows is in C or has a descendent in C;*
2. *every other node is outside C.*

In order to study the representation of a conditional independence model, we need to distinguish between dependence map and independence map, since there are conditional independence models that cannot be completely represented by a DAG (see e.g. [9, 11]).

In the following we denote with J (analogously for $\bar{J}$, J_*) both a set of triples and a set of conditional independence relations, obviously, the triples are defined on the set S and the independence relations on $\tilde{S}$. Then a graph representing the conditional independence relations of J has S as node set.

Definition 3. *Let J be a set of conditional independence relations on a set $\tilde{S}$ of random variables. A DAG $G = (S, E)$ is a* dependence map *(briefly a D–map) if for all triple $(A, B, C) \in S^{(3)}$*

$$(A, B, C) \in \bar{J} \Rightarrow (A, B, C)_G.$$

Moreover, $G = (S, E)$ is an independence map *(briefly an I–map) if for all triple $(A, B, C) \in S^{(3)}$*

$$(A, B, C)_G \Rightarrow (A, B, C) \in \bar{J}.$$

G is a minimal *I–map of J if deleting any arc, G is no more an I-map.*

G is said to be a perfect map *(briefly a p–map) if it is both a I–map and a D–map.*

The next definition and theorem provide a tool to build a DAG given a independence model $\bar{J}$.

Definition 4. *Let $\bar{J}$ be an independence model defined on S and let $\pi =< \pi_1, \ldots, \pi_n >$ an ordering of the elements of S. The* boundary strata *of $\bar{J}$ relative to π is an ordered set of subsets $< B_1, B_2, \ldots, B_i, \ldots >$ of S, such that* *each B_i is a minimal set satisfying $B_i \subseteq S_{(i)} = \{\pi_1, \ldots, \pi_{i-1}\}$ and $\gamma_i = (\{\pi_i\}, S_{(i)} \setminus B_i, B_i) \in \bar{J}$.*

The DAG created by setting each B_i as parent set of the node π_i is called boundary DAG *of J relative to π.*

The triple γ_i is known as *basic triple*.

The next theorem is an extension of Verma's Theorem [14] stated for conditional independence relations (see [9]).

Theorem 1. *Let J be a independence model closed with respect to the semi–graphoid properties. If G is a boundary DAG of J relative to any ordering π, then G is a minimal I–map of J.*

The previous theorem helps to build a DAG for an independence model $\bar{J}_P$ induced by a probability assessment P on a set of random variables $\tilde{S}$ and a fixed ordering π on indices of S.

Now, we recall an interesting result [9].

Corollary 1. *An acyclic directed graph $G = (S, E)$ is a minimal I–map of an independence model J if and only if any index $i \in S$ is conditionally independent of all its non-descendants, given its parents $pa(i)$, and no proper subset of $pa(i)$ satisfies this condition.*

It is well known (see [9]) that the boundary DAG of J relative to π is a minimal I-map.

5 BN-draw function

The aim of this section is to define the procedure BN–DRAW, which builds a minimal I–map G (see Definition 3) given the fast closure J_* (introduced in Section 3) of a set J of independence relations. The procedure is described in the algorithm 1.

Given the fast closure set J_*, we cannot apply the standard procedure (see [7, 9]) described in Definition 4 to draw an I–map because, in general, the basic triples related to an arbitrary ordering π could not be elements of J_*, but they could be just g–included to some triples of J_*, as shown in Example 1.

Example 1. Given $J = \{(\{1\}, \{2\}, \{3, 4\}), (\{1\}, \{3\}, \{4\})\}$, we want to find the corresponding basic triples and to draw the relevant DAG G related to the ordering $\pi =< 4, 2, 1, 3 >$.

By the closure with respect to graphoid properties we obtain
$\bar{J} = \{$ $(\{1\}, \{2\}, \{3, 4\})$, $(\{1\}, \{3\}, \{4\})$, $(\{1\}, \{2, 3\}, \{4\})$, $(\{1\}, \{2\}, \{4\})$,
$(\{1\}, \{3\}, \{2, 4\})$, $(\{2\}, \{1\}, \{3, 4\})$, $(\{3\}, \{1\}, \{4\})$, $(\{2, 3\}, \{1\}, \{4\})$, $(\{2\}, \{1\}, \{4\})$,
$(\{3\}, \{1\}, \{2, 4\})$ $\}$
and the set of basic triples is $\Gamma = \{(\{1\}, \{2\}, \{4\}), (\{3\}, \{1\}, \{2, 4\})\}$.

By FC1 we botain $J_* = \{(\{1\}, \{2, 3\}, \{4\})\}$ and it is simple to observe that $\Gamma \sqsubseteq J_*$.

Algorithm 1 DAG from J_* given an order π of S

```
function BN-DRAW(n, π, J_*)                          ▷ n is the cardinality of S
    for i ← 2 to n do
        pa ← S_(i)
        for each (A, B, C) ∈ J_* do
            if π_i ∈ A then
                p ← C ∪ (A ∩ S_(i))
                r ← B ∩ S_(i)
                if (C ⊆ S_(i)) and (p ∪ r = S_(i)) and (r ≠ ϕ) and (|p| < |pa|) then
                    pa ← p
                end if
            end if
            if π_i ∈ B then
                p ← C ∪ (B ∩ S_(i))
                r ← A ∩ S_(i)
                if (C ⊆ S_(i)) and (p ∪ r = S_(i)) and (r ≠ ϕ) and (|p| < |pa|) then
                    pa ← p
                end if
            end if
        end for
        draw an arc from each index in pa to π_i
    end for
end function
```

The procedure BN-DRAW finds, for each π_i, with $i = 2, \ldots, n$, and possibly for each $\theta \in J_*$, a triple $(\{\pi_i\}, B, C) \sqsubseteq \theta$ such that $B \cup C = S_{(i)}$ and C has the minimum cardinality (analogously for the triples of the form $(A, \{\pi_i\}, C)$). It is easy to see that for each π_i, the triple with the smallest cardinality of C among all the selected triples, coincides with the basic triple γ_i, if γ_i exists. The formal justification of this statement is given by the following result:

Proposition 2. *Let $\bar{J}$ be an independence model on an index set S, J_* its fast closure and π an ordering on S. Then, the set*

$$\mathcal{B}_i = \{(\{\pi_i\}, B, C) \in S^{(3)} : B \cup C = S_{(i)}, \{(\{\pi_i\}, B, C)\} \sqsubseteq J_*\}$$

is not empty if and only if the basic triple $\gamma_i = (\{\pi_i\}, S_{(i)} \setminus B_i, B_i)$ related to π exists, for $i = 1, \ldots, |S|$.

Proof. Suppose that $\mathcal{B}_i$ is not empty. If there are two triples θ^1, θ^2 having the same cardinality of $pa(i)$ then, by definition of J_*, there is also the triple θ^3 obtained by applying the intersection rule between them. Since the third component of θ^3 has a smaller cardinality than those of θ^1 and θ^2, this means that the triple with the minimum cardinality of C is unique and it coincides with the basic triple γ_i. Vice versa, if the basic triple $\gamma_i = (\{\pi_i\}, S_{(i)} \setminus B_i, B_i)$ for π_i exists, then it is straightforward to see that $\gamma_i \in \mathcal{B}_i$. $\square$

Note that BN–DRAW allows to make the corresponding I–map related to π in linear time with respect to the cardinality of J_*, while the standard procedure requires a time proportional to the size of $\bar{J}$, which is usually much larger, as shown also in the empirical tests in [1]. Also the space needed in memory is almost exclusively used to contain the fast closure. Note that a theoretical comparison between the size of the whole closure and the size of the fast closure has not already been found and seems to be a very difficult problem.

The next example compares the standard procedure recalled in Definition 4 with BN-DRAW to build the I–map, given a subset J of $S^{(3)}$ and an ordering π among the elements of S.

Example 2. Consider the same independence set J of Example 1 and the ordering $\pi =< 4, 2, 1, 3 >$, we compute the basic triple by applying BN-DRAW to $J_* = \{\theta = (\{1\}, \{2, 3\}, \{4\})\}$. For $i = 2$ we have $2 \in B$, $p = \phi$, $r = \phi$, $C = \{4\} \subseteq \{4\}$, then there is no basic triple. For $i = 3$ we have $1 \in A$, $p = \phi$, $r = \{2\}$, $C = \{4\}$ then $(1, 2, 4)$ is a basic triple g–included to θ. For $i = 4$ we have $3 \in B$, $p = \{2\}$, $r = \{1\}$, $C = \{4\}$ then $(\{3\}, \{1\}, \{2, 4\})$ is a basic triple g–included to θ.

Therefore, we obtain the same set Γ computed in Example 1.

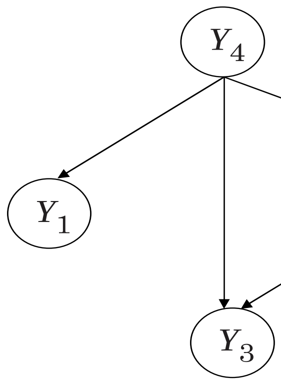

Fig. 1. $I - map$ related to $\pi =< 4, 2, 1, 3 >$

6 Perfect map

In this section we introduce a condition ensuring the existence of a perfect map for an independence model $\bar{J}$, starting from the fast closure J_* of J (avoiding to build the whole $\bar{J}$, as recalled in Section 2). Given an ordering π on S, we denote with G_π the corresponding I–map of J_* with respect to π. Furthermore, we associate to any index $s \in S$ the set $S_{(s)}$ of indices appearing in π before s and the minimal subset B_s of $S_{(s)}$ such that $\gamma_s = (s, S_{(s)} \setminus B_s, B_s)$ is the basic triple, if any, as introduced in Definition 4.

Before stating the sufficient condition that ensures the existence of a perfect map G_π of the fast closure J_*, with respect to π, we want to underline some relations among the indices of a triple represented in G_π and g–included in J_*. In particular, we focus our attention on the relationship among the nodes associated to C and those related to $pa(A \cup B)$.

By Corollary 1, for any index $i \in S$ the triple $\{(\{i\}, S \setminus ds(i), pa(i))\}$ is represented in G_π and, since G_π is an I–map, $\{(\{i\}, S \setminus ds(i), pa(i))\} \sqsubseteq J_*$. Moreover, by d–separation, for any $K \subseteq ds(i) \setminus \{i\}$ such that $pa(i)$ d–separates $S \setminus ds(i)$ and K, also $\{(\{i\} \cup K, S \setminus ds(i), pa(i)), (\{i\}, S \setminus ds(i), pa(i) \cup K)\} \sqsubseteq J_*$.

Moreover, we have the following result considering I–maps and fast closure J_*.

Proposition 3. *Let J be a set of conditional independence relations and J_* its fast closure.*

If G_π is an I–map of J_*, with respect to π, then, for any triple $\theta = (A, B, C) \in J_*$ represented in G_π, it holds $pa(D_A^B) \cup pa(D_B^A) \subseteq C$, where $D_A^B = ds(A) \cap B$ and $D_B^A = ds(B) \cap A$.

Proof. Suppose by absurd that there exist $\alpha \in D_A^B$ and $j \in pa(\alpha)$ such that $j \notin C$. We want to show that $\theta' = (A, B \cup \{j\}, C)$ is represented in G_π. Let ρ a path from $a \in A$ to j. If ρ passes through α, then ρ is blocked by C (because $(A, B, C)_{G_\pi}$). Otherwise, if ρ does not pass through α, then the path ρ' from a to α obtained from ρ adding the edge (j, α) is blocked by C. Since j is not a converging node (and $j \notin C$), ρ is blocked by C.

Since G_π is an I–map, it follows that $\{\theta'\} \sqsubseteq J_*$, but $\theta \sqsubseteq \theta'$ and then θ would not be a maximal triple. □

By the previous observations and Proposition 3 it comes out the idea related to the relationship between the component C and $pa(A \cup B)$ of a triple (A, B, C), behind the condition introduced in the next proposition assuring the existence of a perfect map.

Proposition 4. *Let J be a set of conditional independence relations and J_* its fast closure.*

Given an ordering π on S, if for any triple $\theta = (A, B, C) \in J_*$ the ordering π satisfies the following conditions

1. *all indices of C appear before all indices belonging to one of the sets A or B;*
2. *all indices of $X = (A \cup B \cup C)$ appear in π before all indices belonging to $S \setminus X$;*

then the related I–map G_π is a perfect map.

Proof. Consider a triple $\theta = (A, B, C) \in J_*$. Under the hypotheses *1.* and *2.*, consider the restriction π_X of π to $X = (A \cup B \cup C)$. If we assume (without loss of generality) that all indices of C appear before of all those of B in π_X, then any index $b \in B$ has as parents the set of indices $B_b \subseteq C \cup (S_{(b)} \cap B)$. Therefore no index of A is a parent of any index of B. Moreover, no index of A can be a descendent of any index of B. In fact, the basic triple $\gamma_b = (b, S_{(b)} \setminus B_b, B_b)$ associated to b satisfies the condition $A \cap S_{(b)} \subseteq S_{(b)} \setminus B_b$, by construction, and for any index $a \in A$ appearing in π after at least a index $b' \in B$, the basic triple $\gamma_a = (a, S_{(a)} \setminus B_a, B_a)$ satisfies conditions: $B \cap S_{(a)} \subseteq S_{(a)} \setminus B_a$ and $B_a \subseteq C \cup (S_{(a)} \cap A)$.

We prove now that θ is represented in G_π. By the previous observations, in G_π no arc can join any element of A and any element of B. Let us consider a path between a node $a \in A$ and a node $b \in B$. If the path passes through a node y outside X, then y must be a collider (i.e. both edges end in y), because each index of X precedes each index outside X and therefore there are no arc from y to any index of X. Since neither y nor any of its descendent is in C, this path is blocked by C.

On the other hand, if the path passes only inside X, it must pass through a node c in C which cannot be a collider, since c precedes all the elements of B. □

This result is a generalization to triples of J_* to that proved for basic triples in Pearl [9].

The next example shows that even if the conditions *1.* and *2.* of previous proposition are not satisfied, there could exist a perfect map.

Example 3. Let us consider the set $J = \{\theta_1 = (\{1,2,3\}, \{4,5,6,7\}, \{8,9\}), \theta_2 = (\{1,4\}, \{2,5,8\}, \{6,9\})\}$ of independence relations. Then, by applying $FC1$ we obtain
$J_* = \{\theta_1, \theta_2, \theta_3 = (\{1\}, \{2,4,5,6,7,8\}, \{9\}), \theta_4 = (\{2\}, \{1,4,5,6,7\}, \{8,9\}),$
$\theta_5 = (\{4\}, \{1,2,3,5,8\}, \{6,9\}), \theta_6 = (\{5\}, \{1,2,3,4\}, \{6,8,9\}),$
$\theta_7 = (\{2,4\}, \{1,5\}, \{6,8,9\})\}$.

The conditions *1.* and *2.* of Proposition 4 do not hold: in fact, by considering the triples θ_3 and θ_5 it is simple to observe that $7 \in X_3$, but $7 \notin X_5$ and $3 \notin X_3$, but $3 \in X_5$. Then, there is no ordering π satisfying conditions *1.* and *2.* of Proposition 4 for any $\theta \in J_*$.

However, by considering the ordering $\pi =< 1,9,2,8,3,5,6,4,7 >$, we can show that the related I-map G_π is perfect, i.e. it represents any triple of J_*.

By using Proposition 4 it is possible to define the following Algorithm 2, called SearchOrder

The algorithm SearchOrder searches for an ordering satisfying the conditions stated in Proposition 4. It firstly tries to find all the ordering constraints on the variables required by the condition 1. If an inconsistency is found, then no ordering exists. Then, it uses a backtracking procedure which decides, for each triple (A, B, C), if all the indices in C precede all the indices in A or all the indices in B. In this step, an inconsistency causes a backtracking phase. The algorithm can terminate with success by finding an ordering which satisfies proposition4, therefore a p–map for J_* can be found by using BN–draw. On the other hand, the algorithm can report a failure, because no ordering respects the conditions. But as shown in the Example 3, a p–map can still exist, hence the procedure can only give a partial answer to the question if J is representable with a DAG.

Algorithm 2

```
function SearchOrder(K)                                  ▷ K ⊆ S^(3)
    for θ = (A, B, C) from K do
        X ← A ∪ B ∪ C
        R ← R ∪ (X ⪯ S \ X)
        if R is inconsistent then
            return ⊥
        end if
    end for
    return Backtrack(K, R)
end function
function Backtrack(K, R)                  ▷ K ⊆ S^(3) and R is a partial order
    if K = ∅ then
        return an order taken from R
    else
        R′ ← R
        R ← R ∪ (C ⪯ A)
        if R is not inconsistent then
            r ← Backtrack(K \ θ, R)
            if r ≠ ⊥ then
                return r
            end if
        end if
        R ← R′ ∪ (C ⪯ B)
        if R is not inconsistent then
            return Backtrack(K \ θ, R)
        else
            return ⊥
        end if
    end if
end function
```

The algorithm has been implemented by using a SAT solver to perform the second step (the function Backtrack) because it is possible to formulate the problem of finding an ordering which satisfies the condition 2 as a propositional satisfiability problem. The first empirical results show that this method is quite efficient.

7 Conclusions

We have shown some properties of graphoid structures, which allow to compute efficiently the closure of a set J of conditional independence statements, compatible with a conditional probability ([1–3]). Moreover, from these properties it is possible to design an alternative method to build an I–map G_π, given an ordering π on the variable set S.

We have dealt with the problem of finding a perfect map given the fast closure J_*. In particular, we are looking for an ordering π giving rise to a perfect map, if there exists. Actually, we have made a first step in this direction by obtaining a partial goal. In fact, we have introduced a sufficient condition for the existence of a perfect map.

We are now working to relax this condition with the aim of finding a necessary and sufficient condition for the existence of an ordering generating a perfect map. For what we understand, such a condition will need to explore the relations among the triples in J_* and the components of each triple. We are also interested in translating this condition into an efficient algorithm.

Another strictly related, open problem is to find more efficient techniques to compute J_*, because it is clearly the first step needed by any algorithm which finds, if any, a DAG representing J.

References

1. M. Baioletti, G. Busanello, B. Vantaggi (2008). Algorithms for the closure of graphoid structures. *Proc. of 12th Inter. Conf. IPMU 2008*, Malaga pp. 930–937.
2. M. Baioletti, G. Busanello, B. Vantaggi (2008). Conditional independence structure and its closure: inferential rules and algorithms. In *International Journal of Approximate Reasoning* (submitted).
3. M. Baioletti, G. Busanello, B. Vantaggi (2009). Conditional independence structure and its closure: inferential rules and algorithms. In *Technical Report*, 5/2009 of University of Perugia.
4. G. Coletti, R. Scozzafava (2000). Zero probabilities in stochastical independence. In *Information, Uncertainty, Fusion, Kluwer Academic Publishers, Dordrecht*, B. Bouchon- Meunier, R.R. Yager, L.A. Zadeh (Eds.), pp. 185–196.
5. G. Coletti, R. Scozzafava (2002). *Probabilistic logic in a coherent setting.* Dordrecht/Boston/London: Kluwer (Trends in logic n.15).
6. A.P. Dawid (1979). Conditional independence in statistical theory. *J. Roy. Stat. Soc. B*, 41, pages 15–31.
7. F.V. Jensen (1966). *An Introduction to bayesian Networks*, UCL Press and Springer Verlag.
8. S.L. Lauritzen (1996). *Graphical models.* Clarendon Press, Oxford.
9. J. Pearl (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann, Los Altos, CA.
10. M. Studený (1997). Semigraphoids and structures of probabilistic conditional independence. *Ann. Math. Artif. Intell.*, 21, pages 71–98.
11. M. Studený (1998). Complexity of structural models. *Proc. Prague Stochastics '98*, Prague, pages 521–528.
12. M. Studený, R.R. Bouckaert (1998). On chain graph models for description of conditional independence structures. *Ann. Statist.*, 26 (4), pages 1434–1495.
13. B. Vantaggi (2001). Conditional independence in a coherent setting. *Ann. Math. Artif. Intell.*, 32, pages 287–313.
14. T. S. Verma (1986). Causal networks: semantics and expressiveness. *Technical Report R–65, Cognitive Systems Laboratory*, University of California, Los Angeles.
15. J. Witthaker, J. (1990). *Graphical models in applied multivariate statistic.* Wiley & Sons, New York.
16. S.K.M. Wong, C.J. Butz, D. Wu (2000). On the Implication Problem for Probabilistic Conditional Independency. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 30, 6, pp. 785–805.