Acta Universitatis Apulensis No. 36/2013
ISSN: 1582-5329 pp. 155-164

# SOME RESULTS ABOUT ESSENTIAL GRAPHS

A. Garrido

Abstract. When working with directed acyclic graphs (by acronym, DAGs or GDAs), it often happens that the problem is not treatable, because temporary resources consumed, where it can get to solve it, would not be bearable. Therefore, it would be very convenient to approach after a previous classification of DAGs, to reduce as much as possible the time of execution, work space for it in the equivalence class of DAGs. Or in the more general case on the space of equivalence classes of Chain Graphs. In such equivalence classes of DAGs are what are called in various ways, according to the authors, appearing as Essential Graphs, but also as complete patterns or completed patterns, CPDAGs (completed PDAGs), etc. By reducing the number of graphs to consider only one equivalence class, this will allow to improve the efficiency of our calculations, reducing computational complexity. We analyze here some very important current results of this new and useful mathematical tool.

2000 *Mathematics Subject Classification*: 30C45.

## 1. Introduction

Bayesian Networks (BNs, by acronym) represent joint probabilities. They are mathematical tool belonging to Probabilistic Graphical Models [7, 8, 12]. It is said that two BNs are equivalent (denoted by *sim* ), if they represent the same joint probability distribution. We require the compliance with the following properties:

- Reflective: $B \sim B, \forall B$.
- Symmetric: if $B \sim B'$, then $B' \sim B$.
- Transitive: if $B \sim B'$ and $B' \sim B''$, then $B \sim B''$.

So, it is a relationship of equality (or equivalence) defined on the set of BNs on which laid a partition into equivalence classes, as we shall see. Then, we will see some useful characterizations of equivalence between Bayesian Networks [3, 5].

Let $S$ and $S'$ be two such structures Bayesian Networks on $V$. We say that $S$ is equivalent to $S'$, denoted by $S \sim S'$, if for each parameter, $q \in S$, there is another parameter, $q' \in S'$, such that:

$$P(V/S, q) = P(V/S', q')$$

Then $S$ represents any probability distribution that $S$' is capable of representing, and vice versa.

**Theorem.** *$S \sim S'$ iff both structures involves the same IC set (Independence Conditional by IC), following the Global Markov's Property [4].*

In a Directed Acyclic Graph (DAG, by acronym), if you delete the addresses to each of its directed edges, we would obtain the skeleton of the graph.

We call immorality all type setting, $X \rightarrow Y \leftarrow Z$, where we see the links directed, $X \rightarrow Y$ and $Y \leftarrow Z$, but where no nodes $X$ and $Y$ must be adjacent, if we want to remain an immorality [10].

Remember [2, 5] that also may be called v-structures or patterns head-head, according to the authors. When joined together ("married") the corresponding node-parent by a line (undirected link), and we make the cease being directed edges, then we say that "the graph is moralized", or obtained Graph-nest the starting Moral.

**Theorem** (Verma and Pearl, 1991)**.** *Two BN's models are equivalent (∼) iff both has the same skeleton and also the same immoralities.*

To have the same skeleton, must match before the order O (which is the number of vertices, or nodes) and size (number of edges, or links) of the graphs considered. In DAGs, the edge $X \rightarrow Y$ is said covered (a covered edge), if:

$$[S] = \text{class of} S = \{\text{all the models equivalent } to S\}$$

**Theorem** (Chickering, 1995)**.** *Two BN's models are equivalent if exist an inversion sequence transforming edges covered one model to the other.*

Given the $S$ aforementioned, we can denote the equivalence class of $S$ by [S]. In each class, would be contained all Markov Bayesian networks that are equivalent to a given (or $\sim$ together). To get to represent and work with it in a more efficient, it will be enough to take any of its representatives as you own, which will be sapped so the essential graph of this class [3, 9]. There are various ways of representing these classes,

- Through the skeleton, and immoralities.
- Composing the graph by directed links, and undirected links.

A directed edge is said to be compelled, or essential, given a BN structure, $S$, if you will always be present in all structures equivalent to $S$. We knows [1, 5] that every edge involved in one of the immoralities always be one of this kind, "compelled", or "essential". But these are not all essential edges: other links can be compelled to do so without involving in immorality. An essential graph of a Bayesian network structure, $S$, is a PDAG (partial DAG) such that:

- The skeleton is the same as the $S$.
- The essential edges, and only they, would target.

In a mathematical or formal way: Let C be a class of Markov equivalent DAGs together [2, 4]. Then, the graph is the lower graph essential that is greater than any of those which make the class DAGs. If we denote $G*$, that is to say:

$$G* = \cup G : G \in C$$

Such a union of graphs could be obtained by taking for $G*$ the union of the nodes and arcs of graphs G,

$$V(G*) = \cup V(G), E(G*) = \cup E(G)$$

Directed links connecting the same pair of nodes, but have opposite directions in two graphs belonging to the same class are replaced by a line. So, $G*$ is the least of the dimensions superior to all graphs of the class it represents [5, 7, 8].

Algorithm to compute the essential graph of $S$:

1. We calculate the skeleton of $S$. Orienting edges only those involved in immorality.
2. Then, the edges are guided essential, without introducing new immoralities thereby.

Can orient other edges, but following these rules:

i. for each $X \to Y \leftarrow Z$, such that X and Y are not adjacent, orient $Z - Y$ as $Z \to Y$.

ii. for each such that there is a directed path from X to Y orientate $X - Y$ as $X \to Y$.

iii. for $X - Z - Y$ such that X and Y are not adjacent, with: $X \to W$, $Y \to W$, $Z - W$, orient Z as $Z \to W$.

**Theorem.** *The second step of the above algorithm is consistent and complete.*

*Consistency (or soundness)*: edges oriented by this algorithm are of essential and are correctly oriented. *Completeness*: the algorithm guides all essential edges. This is a very important search algorithm with essential graphs that were not classified even.

## 2. Scoring Function-Invariant Equivalence

Invariant scoring function for equivalence is those which give the same value are applied to score when acts on equivalent models. Sometimes may be called by different and equivalent ways, such as equivalent score or likekihood equivalent [2, 3, 5]. Some could be used to find the value of score equivalence classes. In other cases however it will be not possible. For example, the BIC score really is $\sim -invariant$, given by

$$BIC(S/D) = logP(D/S, q*) - d/2.logN$$

Two equivalent models of BNs will always have the same computational complexity [8, 12]. For inversion (reversal of arcs) of essential links does not change complexity class.

Heckerman et al. (1994) proved that: - The "marginal likelihood", is $\sim -invari-ant$, if properly established a priori parameters (the "priors").

Bayesian network can look each candidate, G, as a potential generating the data set, $D$.

Scoring of BNs: The score P (D/E), when it is a Bayesian Network chordal, have a shape similar to the usual score BN. Recall that tailpiece (or chordal) means if Card $(V) \geq 4$, then there exists at least one edge (the "rope") that goes between non-adjacent nodes [2, 6].

If the graph is not chordal, it would be necessary to go through successive iterations approaching. But if there are hidden variables, everything becomes more complex. Models include that: Given two BN's models, S and S', are said to include $S' \subset S$, if every IC-assumptions (or IC-claims) that are certain to S' will also be valid for S. Example: Consider the models $X \to Y$, $Z$, and $X \leftarrow Y$. They are equivalent.

## 3. Equivalence Relations and Inclusion

$$S \equiv S' \Leftrightarrow (S \subset S') \wedge (S' \subset S)$$

Considered as equivalence relation between models. And we say that: Strictly includes S to S', if S includes S', but S' no includes S. Example. $X \rightarrow Y \rightarrow Z$ nor include $X \leftarrow Y \leftarrow Z$. $X \rightarrow Y \rightarrow Z$ neither includes $X \leftarrow Y \leftarrow Z$.

**Theorem 1** (Chickering, 2002)**.** *A structure of RB, S, includes other, S', if there is a sequence composed by an inversions of covered arches and arches that transforms S in S'.*

**Example** The model $X \rightarrow Y \leftarrow Z$ model includes $X \leftarrow Y$, Z enough for this to be made transform each other:

- An inversion shielded arc, $X \leftarrow Y, Z \Rightarrow X \rightarrow Y, Z$
- And an addition of arc, $X \rightarrow Y, Z \Rightarrow X \rightarrow Y \leftarrow Z$

**Corollary 2.** *A structure of BN, S, includes to another, S' iff S is able to represent whatever joint distribution S' may represent.*

## 4. Inclusion Boundaries, IB.

Let's distinguish between borders [7, 8, 12], so we consider two subtypes, Inferior and Superior in general. Firstly, the Inferior Border Inclusion Model of S: it is designated as $IB_{-}$ (S), being the set of all those models, S', such that:

- Strictly includes S to S'.
- There is no model, S", such that S strictly includes S", and S" strictly includes to S'.

Inclusion Border Higher S. Designee by $IB^{+}(S)$, when

- S' includes strictly to S, and there will be no model, S", such that S' strictly includes S", and S" also includes strictly S.

General Inclusion Boundary may be defined as

$$IB = IB^{+} \cup IB_{-}$$

$IB^{+}(S)$ consists of all those models which can be obtained from $S$ by:

- A series of arches covered inversions.
- The addition of a single arc.
- Other inversion covered arches.

While $IB_{-}(S)$ would consist of those from $S$ by:

- A series of arches covered inversions.
- By removing a single arch.
- Other inversions on covered arches.

## 5. Enumeration of DAGs.

It may show as an example the case of the classification of all built GDAS from three nodes. That is, for directed acyclic graph, G, of order three, $O(G) = 3$. Its size (T) depends on the class to which they belong, being $T(G) \in \{0, 1, 2, 3\}$. We can observe that it change the degree or valence of nodes. Are identifiable in this instance equivalence classes, here you can see how DAGs group C. As is known, the number of all possible structures for BNs with a number n of nodes, denoted by r(n), is given by the following recurrence equation,

In our case, for graphs of order three, $O(G) = n = 3$, we have

In the case of graphs of order O = 4, everything becomes more complicated, as expected. Applying the recurrence formula obtains 543 different possible configurations from connecting or not said four nodes by directed edges,

If X and Y are not adjacent nodes in a DAG, you can add a directed edge between them (either the $X \to Y$ or $Y \to X$), without thereby display a directed cycle. Let S and P a DAG and a CPD (conditional probability distribution), respectively. Suppose we add to S an edge $X \to Y$ produces no cycles. Then, it said to add that edge to S is useless, if $X \perp Y/Pas(Y)$. If not, it would seem that adding that edge to S is useful. Both cases respect to the probability distribution $P$ (w.r.t. $P$, by acronym). A scoring function is said to be locally consistent, if add an edge increases the score useful, and add an edge decreases it useless [12].

## 6. Graph algorithms and essential graphs

**Theorem 3** (Castelo and Kocka, 2002; Chickering, 2002)**.** *Consider a climbing search algorithm (hill-climbing), Alg, using a score function $f$, based on a set of data, D. Assume that:*

1. *D has been sampled from a probability distribution, P, faithful to S, and that the sample size is sufficiently large.*

2. *The models discussed in each step of Alg contains:*

   *i) All models $IB_-$ current model.*

   *ii) All models are available now through the inclusion of a single arch.*

*So, Alg reaches a model in the class [S], and stop there.*

Then we will see an application to GES (acronym by Greedy Equivalence Search, or Greedy Search). Before, clarify that a good structural learning algorithm must be able to discover the "truth", provided that we have enough data. That is, it must be able to reconstruct the true model, from the sample. Assuming that the set of data generated by the BN is complete, we may analyze when and how to achieve the reconstruction of the model [2, 3, 11].

From the model, which has no arcs, or links (edges), we distinguish two phases, for GES:

1. Repeat until a maximum local:

   - Browse all models in the current model $IB^+$.
   - Select, from among them, giving the best score.

2. Repeat until reaching maximum local:

   - Browse all models $IB_-$ current model.
   - Select the best present a score value.

Thus, in the first stage, the algorithm finds a model that includes the true model. In the second phase reduced to the true model. According to the latest mentioned theorem, would only be necessary to add arcs in the first phase. But generating all models in the IB model that theorem applied directly can be very expensive computationally ally. The solution would be essential work with graphs, that is, should be a graph by each equivalence class of DAGs, as his representative. We can see how you can search inclusion boundaries implicit in the models, which would be used to search operators on essential graphs [4, 7, 8]. However, there are still some problems in order to determine the local maxima: we'll never have infinite data and furthermore, these are not usually generated by conditional probabilities faithful to GDAs. GES also may not be able to recover the "generative model", i.e. the original model. Nevertheless, it is better that the classical search method or gradient escalation, with the DAGs. As the search is necessary, we ended using

fast algorithms. For example, we have [12] Chow-Liu's algorithm for trees (1968). For learning Bayesian Network can be simple, if the order of the variables is well established, and a limit on the number of parents. In general, it uses Heuristic Search on network space. Rather than return a single best candidate for the real network, we will return an approximation to the posterior distribution over all possible networks. For this approach, is used MCMC (Monte Carlo Method for Markov Chains). The BN's learning algorithms typically use different search methods, but the same space, that of DAGs. Although alternatives exist, such as for example [7, 8],

- The space of orderings of the variables, with a secondary search at the DAGs space support, given an ordering.
- The space of essential graphs, as representing PDAGs canonically equivalence classes of DAGs.
- The space of RPDAGs (or PDAGs restricted), who also represent equivalence classes of DAGs.

As different DAGs can determine the same equivalence class of Markov, will be of interest to know how much is gained in efficiency for each class when we take one representative, instead of going considering exhaustively (and more costly in resources computational) each DAGs. To find out, developed a certain program [5], that of Gillispie and Perlman (2001), from which it could enumerate the equivalence classes of DAGs, according to this criterion of equivalence between Bayesian networks (and statement) of Verma and Pearl. With it is estimated that the proportion of C-classes on DAGs is (asymptotically) of 3.75. It is quite reduced (nearly a quarter), although some were less than expected.

## 7. Chain Graphs and Essential Graphs

Chain graphs are clear generalization both as directed graphs and as undirected graphs, i.e. containing both oriented and non-oriented edges [4, 11]. This will increase the computational complexity. Both directed graphs and undirected graphs are special cases of chain: it would be both of those edges containing only one of the two types. These graphs were then mixed simultaneously represent useful both associative units (such as UGs, undirected graphs) as causal dependencies (the case of DAGs). Hence, UGs and DAGs are particular cases of GCs (chain graphs). So try to represent possible dependencies between random variables. Both types of representations have their own scope. So, for undirected graphs, includes models for image analysis and spatial dependence, while DAGs are more suitable

for statistical analysis. Be applicable in fields such as genetics and psychometrics, or expert systems for modeling and Bayesian networks. In conclusion, the essential graph (here denoted by $G*$) are chain graphs.

## Final note

Hence, $G*$ is the only graph equivalent Markov Chain to $G$. One result of AMP (Anderson, Madigan and Perlman, 1995), which was also set by Meek and Chickering (1995), characterized by four conditions when a graph is a graph chain essential. The property of Markov Chain graphs was established by Frydenberg (1990). And the results on $\sim$ (the equivalence of Chain Graphs, CGs), by LWF (Lauritzen, Wermuth and Frydenberg); AMP (Anderson, Madigan and Perlman); Meek, and Studeny, among others.

## References

[1] S.A. Anderson, D. Madigan, M.D. Perlman, *A characterization of Markov equivalence classes for acyclic digraphs*, Ann. Stat. 25, N$^{o}$ 2, pp. 505-541, 1997.

[2] R. Castelo, M.D. Perlman, *Learning Essential Graph Markov Models from Data*, PGM'2002.

[3] D.M. Chickering, *, A transformational characterization of equivalent Bayesian Networks structures*, UAI, 1995, pp. 87-98.

[4] M. Frydenberg, *The chain graph Markov property*, Scand. J. Statist. 17, pp. 333- 353, 1990.

[5] S.B. Gillispie, M.D. Perlman, *Enumerating Markov Equivalence Classes of Acyclic Digraphs*, UAI'2001, pp. 171-177.

[6] T. Kocka, R. Castelo, *Improved Learning of Bayesian Networks*, UAI' 2001, pp. 269-276, 2001.

[7] S.L. Lauritzen, *Graphical Models*, OUP (Oxford University Press), 1996.

[8] D. Madigan, *Probabilistic Graphical Models*, Seminar given at Rutgers University, 2006.

[9] A. Roverato, M. Studený, *A graphical representation of equivalence classes of AMP chain graphs*, JMLR 7, pp. 1045 -1078, 2006.

[10] M. Studený, *Characterization of essential graphs by means of the operation of legal merging of components*, Int. J. of Unc., Fuzziness and Knowledge-Based Systems 12, pp. 43-62, 2004.

[11] M. Volf, M. Studený, *A graphical characterization of the largest chain graphs*, IJAR 20, pp. 209-236, 1999.

[12] N.L. Zhang, *Introduction to Bayesian Networks*, Hong Kong University, 2007.

Angel Garrido
Department of Fundamental Mathematics,
Faculty of Sciences,
UNED, Madrid, Spain
email: *agarrido@mat.uned.es*