On the Complexity of Propositional and Relational Credal Networks

Fabio Gagliardi Cozman
Escola Politécnica
Universidade de São Paulo, Brazil
fgcozman@usp.br

Denis Deratani Mauá
Instituto de Matemática e Estatística
Universidade de São Paulo, Brazil
denis.maua@gmail.com

Abstract

A credal network associates a directed acyclic graph with a collection of sets of probability measures. Usually these probability measures are specified through several tables containing probability values. Here we examine the complexity of inference in Boolean credal networks when probability measures are specified through formal languages, by extending a framework we have recently proposed for Bayesian networks. We show that sub-Boolean and relational logics lead to interesting complexity results. In short, we explore the relationship between language and complexity in credal networks.

Keywords. Credal networks, Propositional logic, Relational logic, Complexity, Data complexity.

1 Introduction

A credal network represents a set of probability distributions through a directed acyclic graph and an associated set of "local" credal sets [1, 6]. Usually these local credal sets are specified using tables containing probability values, possibly with some additional constraints between them. In practice, any elicitation strategy must adopt some specification language in which to encode probability assessments. For instance, one may allow inequalities such as $\mathbb{P}(A) \geq 1/2$, or perhaps interval-valued assessments such as $\mathbb{P}(A) \in [3/5, 7/10]$; of course, one may have a specification language with propositions and Boolean operators, or even relations and quantifiers.

In this paper we study properties of credal networks as parameterized by specification languages. We look at the balance of expressivity for specification languages and the complexity of inferences. We concentrate on Boolean variables, and focus on a particular semantics for credal networks (the semantics of "strong extensions"). To investigate the interplay between expressivity and complexity, we extend a framework we have recently developed to study the complexity of Bayesian networks [9].

We start with some necessary background in Section 2. We discuss our framework in Section 3, in particular looking at propositional languages. Sections 4 and 5 examine relational languages.

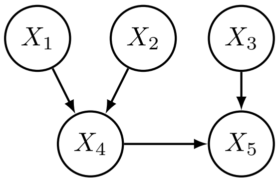
2 Credal networks and their strong extensions

In this paper every possibility space Ω is finite; a random variable is simply a function from Ω into the reals, and we consider only random variables taking on two values, 1 (meaning "true") and 0 (meaning "false"). A set of probability measures is called a *credal set* [18]. We abuse language by referring to sets of probability distributions, and also to sets of probability mass functions, as credal sets. A set of distributions for a variable X is denoted by $\mathbb{K}(X)$. Given a credal set, for any event A we have its *lower* and *upper* probabilities, denoted by $\underline{\mathbb{P}}(A)$ and $\overline{\mathbb{P}}(A)$ respectively: $\underline{\mathbb{P}}(A) = \inf \mathbb{P}(A)$ and $\overline{\mathbb{P}}(A) = \sup \mathbb{P}(A)$. In this paper W, X, Y and Z denote random variables, while A and B denote events or propositions.

A conditional credal set is obtained by applying Bayes rule to each possible distribution in a credal set; we also refer to sets of conditional distributions and conditional mass functions as conditional credal sets. We adopt regular conditioning; that is, $\mathbb{K}(X|A)$ is the set of all conditional distributions that are obtained from distributions such that $\mathbb{P}(A) > 0$ [30]. We denote by $\mathbb{K}(X|Y)$ the set containing a credal set $\mathbb{K}(X|Y = y)$ for each possible value of Y. The sets $\mathbb{K}(X|Y)$ are *separately specified* when there is no constraint on the conditional set $\mathbb{K}(X|Y = y_1)$ that is based on the properties of $\mathbb{K}(X|Y = y_2)$, for any $y_2 \neq y_1$. For events A and B, we define lower and upper conditional probabilities: $\underline{\mathbb{P}}(A|B) = \inf_{\mathbb{P}:\mathbb{P}(B)>0} \mathbb{P}(A|B)$ and $\overline{\mathbb{P}}(A|B) = \sup_{\mathbb{P}:\mathbb{P}(B)>0} \mathbb{P}(A|B)$.

Given some marginal and conditional credal sets, an *extension* of these sets is a joint credal set with the given marginal and conditional credal sets.

A credal network consists of a directed acyclic graph where each node is a random variable X_i, together with a set of constraints on probability values. The graph is assumed to encode independence relations amongst variables, and the constraints convey the probabilistic assessments. The independence relations are given by a Markov condition, soon to be explained. Such a structure is useful as a representation for beliefs, opinions, and statistical summaries that may be available when modeling a particular problem. For instance, suppose we have five variables, representing say economic indicators:



Here we have that X_1 and X_2 are *parents* of X_4; likewise, X_3 and X_4 are parents of X_5. The parents of X_i are denoted by $\text{pa}(X_i)$. The meaning of the graph is conveyed by the *Markov condition*: every X_i is independent of its nondescendants nonparents given its parents. So, X_5 is independent of X_1 and X_2 given X_3 and X_4. Hence by drawing the graph we are expressing our belief that, conditional on X_3 and X_4, no information about X_1 and X_2 can change our assessments on X_5.

To continue the example, we may have some constraints on probabilities. Even though one is free to impose say $\mathbb{P}(X_1 = 0|X_4 = 1) \geq 2/3$ and $\mathbb{P}(X_3 = 1 \wedge X_2 = 0) \leq 1/2$, usually applications constrain assessments to a few simple forms [1, 6]. Typically we have each variable X_i associated with separately specified sets $\mathbb{K}(X_i|\text{pa}(X_i))$. When every credal set $\mathbb{K}(X_i|\text{pa}(X_i) = \pi)$ is a singleton, the resulting model is equivalent to a *Bayesian network*.

Once assessments are given, we can construct their joint *extension*; that is, we can construct a credal set consisting of those joint distributions that satisfy the assessments. We have some freedom here, for we can interpret the "independence relations" in the Markov condition in various ways. There are several concepts of independence that apply to credal sets [7]; we might for instance consider extensions that interpret the Markov condition through *epistemic irrelevance* [11]. In this paper we adopt the most common concept of independence for credal sets; namely, we adopt *strong independence*: X and Y are strongly independent given Z if $\mathbb{K}(X, Y|Z = z)$ is the convex hull of a set of distributions that factorize; that is, if any $p(X, Y|Z = z)$ in this latter set satisfies $p(X, Y|Z = z) = p(X|Z = z)\, p(Y|Z = z)$.

We are always interested in the *largest* extension that satisfies given assessments and independence relations. We refer to such extensions, when strong independence is adopted, as *strong extensions*. Our results are *also* valid if one adopts *complete independence*, provided one always keeps the interest in the largest possible extension: X and Y are completely independent given Z if any probability mass $p(X, Y|Z = z)$ in $\mathbb{K}(X, Y|Z = z)$ satisfies $p(X, Y|Z = z) = p(X|Z = z)\, p(Y|Z = z)$. To simplify the presentation, we focus only on strong independence and strong extensions.

Given a credal network (graph and assessments) and its resulting extension, we are interested in computing conditional upper probabilities such as $\overline{\mathbb{P}}(X_1 = 0|X_2 = 1)$.

3 A Framework for Complexity Analysis

We now extend a framework for complexity analysis that we have recently developed for Bayesian networks [9], so as to include probability intervals. The basic idea is to restrict assessments to two simple forms that are inspired by probabilistic rules [22, 26] and structural models [21]. The framework lets one move down to sub-Boolean constructs and up to relations and quantifiers. In the context of credal networks and strong extensions, our framework is valuable as it imposes some regularity into the specification, for instance automatically implying that all local credal sets are separately specified. So, it offers a combination of flexibility and restraint that should be useful in practical elicitation scenarios.

We will refer to existing complexity classes in our results. To recap, the class PP consists of those problems that can be solved by a nondeterministic polynomially-bounded Turing machine where the acceptance condition is that more than half of computation paths accept [20]. And $\mathsf{NP}^{\mathsf{PP}}$ consists of those problems that can be solved by a nondeterministic polynomially-bounded Turing machine with an oracle that solves PP decision problems [19]. In proofs we use reductions from E-MAJSAT, an $\mathsf{NP}^{\mathsf{PP}}$-complete problem [19]. The E-MAJSAT decision problem is: given a pair (ϕ, k) where ϕ is a Boolean sentence with n propositions, and $k \in [1, n]$ is an integer, is there an assignment of the first k propositions such that the majority of assignments to the remaining propositions satisfies ϕ?

Returning to the specification framework: Consider a set of atomic propositions, $A_1, \ldots, A_n$, and take the

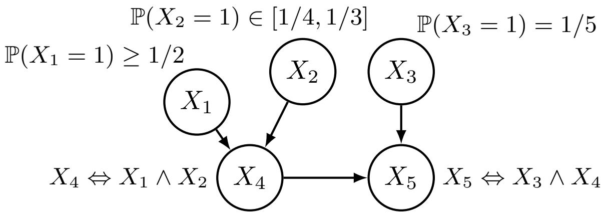


Figure 1: A simple credal network.

set Ω of 2^n truth assignments. Associate a binary variable X_i with atomic proposition A_i, such that $X_i(\omega) = 0$ when A_i is false, and $X_i(\omega) = 1$ when A_i is true, for $\omega \in \Omega$. Our credal networks are to be specified over $X_1, \dots, X_n$; to simplify the presentation, we equate atomic propositions and their associated variables. That is, we write propositional sentences containing variables and their assignments, and we write probabilities for propositional sentences.

We assume that a directed acyclic graph is given, where each node is a variable X, and that each variable X is associated with either:

- an equivalence $X \Leftrightarrow F(Y_1, \dots, Y_m)$, or
- a probabilistic assessment $\mathbb{P}(X = 1) \in [\alpha, \beta]$,

where F is a formula on propositions $Y_1, \dots, Y_m$ that are parents of X, and where α and β are nonnegative rationals in $[0, 1]$. We call the former a *logical assessment*, and the latter a *probabilistic assessment*.

By adopting this restricted syntax, the graph is actually redundant. One can simply give a set of assessments, and as long as there are no cycles in the specification, the graph can be then constructed out of the assessments.

Note that we avoid direct assessments of conditional probability. First, such an assessment may essentially create negation (by imposing $\mathbb{P}(X = 1|Y = 1) = \mathbb{P}(X = 0|Y = 0) = 0$); we wish to control the use of negation. Second, by avoiding conditional probabilities we do not need to start by discussing conditioning on events that can have probability zero, a discussion that is always difficult for the novice [8].

To illustrate the framework, consider the specification in Figure 1. One might interpret this network as follows: X_4 is a health condition that is identified with the conjunction of two risk factors, and X_5 is an illness that depends probabilistic on X_4, with X_3 acting as "inhibitor".

The strong extension of this credal network is simply the convex hull of all extreme Bayesian networks, where an extreme Bayesian network is obtained by taking extreme (upper or lower) probabilities [12, 13]. Hence we have eight possible configurations of variables, and four extreme joint probability distributions. For instance, one such distribution assigns probability 1/2 to $\{X_1 = 1\}$ and probabilty 1/4 to $\{X_2 = 1\}$, while another distribution assigns probability 1 to $\{X_1 = 1\}$ and probability 1/4 to $\{X_2 = 1\}$.

Denote by $\mathcal{C}(\mathcal{L})$ the set of credal networks that can be produced through the framework above, with formulas F from a language $\mathcal{L}$ (a language $\mathcal{L}$ is simply a set of well-formed formulas). Then $\mathsf{INF_d}(\mathcal{L})$ denotes the set of decision problems that yield YES if $\overline{\mathbb{P}}(Q|\mathbf{E}) > \gamma$ for an assignment Q, a conjunction $\mathbf{E}$ of assignments, a rational $\gamma \in [0, 1]$, and a credal network in $\mathcal{C}(\mathcal{L})$, and NO otherwise [10]. The set $\mathbf{E}$ is the *evidence*; we focus only on conjunctions of assignments, and leave for the future the study of more general languages in which to express evidence. To simplify the statement of some results, we denote by $\mathsf{INF_d}^+(\mathcal{L})$ the decision problems defined as in $\mathsf{INF_d}(\mathcal{L})$, with the additional constraint that all assignments are "positive" (that is, variables are only set to true).

Denote by $\mathsf{Prop}(\wedge, \neg)$ the language of well-formed propositional sentences with conjunction and negation. First note that $\mathsf{Prop}(\wedge, \neg)$ can specify any distribution over variables $X_1, \dots, X_n$ that can be specified by a Bayesian network over these variables. To see why, suppose we have a Bayesian network over $X_1, \dots, X_n$. Consider first a variable X with two parents Y_1 and Y_2. Impose:

$$X \Leftrightarrow (\neg Y_1 \wedge \neg Y_2 \wedge Z_{00}) \vee (\neg Y_1 \wedge Y_2 \wedge Z_{01}) \vee (Y_1 \wedge \neg Y_2 \wedge Z_{10}) \vee (Y_1 \wedge Y_2 \wedge Z_{11}),$$

where Z_{ab} are fresh binary variables (that do not appear anywhere else), associated with assessments $\mathbb{P}(Z_{ab} = 1) = \mathbb{P}(X = 1|Y_1 = a, Y_2 = b)$. Obviously we can always produce disjunction using conjunction and negation, so $\vee$ appears as syntactic sugar in this latter expression. Now for a variable X with many parents, we just repeat this structure, by taking into account any possible configuration of parents. The marginal distribution of $X_1, \dots, X_n$ is exactly the distribution specified by the original Bayesian network.

By allowing interval-valued assessments in our framework, we obtain a similar result for credal networks: $\mathsf{Prop}(\wedge, \neg)$ allows us to specify any (separately specified) strong extension over variables $X_1, \dots, X_n$. To see why, suppose we have a separately specified credal network over $X_1, \dots, X_n$. Consider again a variable X with two parents Y_1 and Y_2, and suppose $\mathbb{K}(X|Y_1, Y_2)$ is such that each $\mathbb{K}(X|Y_1 = a, Y_2 = b)$ has two extreme points, $p_0(X|Y_1 = a, Y_2 = b)$ and $p_1(X|Y_1 = a, Y_2 = b)$. Introduce fresh variables W_{ab}

and Z_{abc}, and let

$$X \Leftrightarrow \bigvee_{\substack{a\in\{0,1\}\\ b\in\{0,1\}\\ c\in\{0,1\}}} (Y_1 = a)\wedge(Y_2 = b)\wedge(W_{ab} = c)\wedge(Z_{abc} = 1),$$

and assessments $\mathbb{P}(Z_{abc} = 1) = p_c(X = 1|Y_1 = a, Y_2 = b)$ and $\mathbb{P}(W_{ab} = 1) \in [0, 1]$. This encodes the desired local, separately specified, credal sets. The idea is that a and b select a particular configuration of Y_1 and Y_2, while c selects a particular extreme point of the corresponding local credal set (and then Z_{abc} carries the appropriate probability value). By repeating this structure to take into account any configuration of parents of X, we construct a joint credal set whose marginal is the strong extension of the original credal network (note that we may have to use additional variables with the same role as W_{ab}, in case we have more than two extreme points per credal set).

Given the generality of $\mathsf{Prop}(\wedge, \neg)$, we have that $\mathsf{INF_d}(\mathsf{Prop}(\wedge, \neg))$ is $\mathsf{NP}^{\mathsf{PP}}$-complete [10]. Now consider a more restricted language: denote by $\mathsf{Prop}(\wedge, (\neg))$ the language that uses only conjunction and *atomic* negation (defined as negation that can appear only before a proposition that is associated with a probabilistic assessment). Note that the credal network in Figure 1 belongs to $\mathcal{C}(\mathsf{Prop}(\wedge, (\neg)))$. We know that inference within $\mathsf{Prop}(\wedge, (\neg))$ for Bayesian networks is polynomial as long as evidence is "positive" [9]. Somewhat surprisingly, this result applies to credal networks:

Theorem 1 $\mathsf{INF_d}^+(\mathsf{Prop}(\wedge, (\neg)))$ *can be solved in polynomial time.*

Proof. Consider first a network with just conjunction, and consider a query $Q = \{X_Q = 1\}$. Note first that if a node X appears in Q or in $\mathbf{E}$, then its ascendants must all be set to true. So we first add to $\mathbf{E}$ all ascendants of nodes originally in $\mathbf{E}$; also, if a node has all parents set to true, then it must be true and can be added to $\mathbf{E}$, so we repeat this until no more nodes can be added to $\mathbf{E}$. Now if any descendant of X_Q is in the evidence, then X_Q is necessarily true, so we have $\underline{\mathbb{P}}(Q|\mathbf{E}) = \overline{\mathbb{P}}(Q|\mathbf{E}) = 1$. So, either we have evidence assigned to a descendant of X_Q, and then the solution is immediate, or all descendants are barren nodes that can be discarded. So, to proceed we suppose that X_Q has no descendants. Now continue by d-separation. Collect all nodes that are ascendants of X_Q; these are d-connected to X_Q. Now suppose one of these nodes, say W, points both to an ascendant of X_Q, and to a non-ascendant, say Y, of X_Q. Now if Y is not in $\mathbf{E}$, then it is a barren node that must be discarded. And if Y is in $\mathbf{E}$, then W itself must be in $\mathbf{E}$, hence Y is to be discarded. For instance, consider Figure 2, and suppose $\{Y = 1\}$ is the evidence. Then W, Z and W'

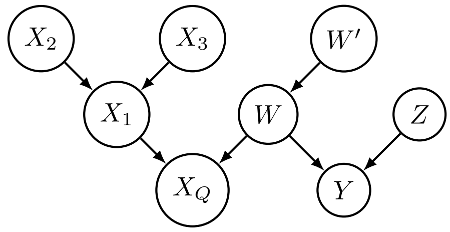


Figure 2: Network in Theorem 1.

are set to true, and we can discard them, as the path emanating from X_Q through W is blocked. Once we have discarded all nodes that are not required for our computation, we are left with an "inverted" tree whose root is X_Q, and where each leaf is either a node set to true, or a node associated with a probability interval. Denote by $X_1, \dots, X_m$ the nodes that are not set to true in this tree; we can then write $X_Q \Leftrightarrow X_1\wedge\cdots\wedge X_m$. So we have $\overline{\mathbb{P}}(X_Q|\mathbf{E}) = \prod_{i=1}^m \overline{\mathbb{P}}(X_i)$; in fact, we also have that $\underline{\mathbb{P}}(X_Q|\mathbf{E}) = \prod_{i=1}^m \underline{\mathbb{P}}(X_i)$. To complete the proof, suppose that atomic negation is allowed, so some variables appear negated. We can run the same procedure already described, with the novelty that we cannot have $X \wedge \neg X$ in the final expression (if that happens, the evidence is inconsistent). □

It seems unlikely that polynomial-time inference can be obtained with other languages for Boolean credal networks, as several simple changes to $\mathsf{Prop}(\wedge, (\neg))$ move us into higher complexity.[1] Consider: even though $\mathsf{INF_d}^+(\mathsf{Prop}(\wedge, (\neg)))$ belongs to P, $\mathsf{INF_d}(\mathsf{Prop}(\wedge, (\neg)))$ does not (as it is PP-hard already when all probability intervals are singletons [9]). Also, if we move from $\mathsf{INF_d}^+(\mathsf{Prop}(\wedge, (\neg)))$ to $\mathsf{INF_d}^+(\mathsf{Prop}(\wedge, \neg))$, then clearly we obtain $\mathsf{NP}^{\mathsf{PP}}$-completeness. Finally, we might move to $\mathsf{INF_d}^+(\mathsf{Prop}(\wedge, \vee, (\neg)))$ by adding disjunction. In doing so, again we move away from polynomial-time behavior, as the following result shows.

Theorem 2 $\mathsf{INF_d}^+(\mathsf{Prop}(\wedge, \vee, (\neg)))$ *is* $\mathsf{NP}^{\mathsf{PP}}$*-complete.*

Proof. Consider an E-MAJSAT problem specified by (ϕ, k). We can code ϕ in CNF within $\mathsf{Prop}(\wedge, \vee, (\neg))$. For a given k, we can associate the first k variables X_i with assessments $\mathbb{P}(X_i) \in [0, 1]$, and the remaining variables X_j with assessments $\mathbb{P}(X_j) = 1/2$. We can then produce a network where each proposition is a root node, and all other nodes are either conjunctions or disjunctions of their parents. This network has size polynomial on the input. Denote by Q an assignment

[1] But note that if network topology is constrained to polytrees, then polynomial behavior is obtained by the 2U algorithm [13]. Hence, by suitably restricting the topology, we still get tractability.

for the leaf node that yields the final conjunction in the CNF. By deciding whether $\overline{\mathbb{P}}(Q) > 1/2$, we solve the E-MAJSAT problem. □

To close this section, we comment on an additional type of assessment that one might allow, namely, assessments where material implication is used instead of equivalence. For instance, suppose that in our previous example we change the logical assessment for X_5 to

$$X_5 \leftarrow X_3 \wedge X_4.$$

A sensible semantics here might be to consider every possible probability measure compatible with this logical constraint; that is, the assessment should mean $\mathbb{P}(A_5|A_3 \wedge A_4) = 1$ and $\mathbb{P}(A_5|\neg(A_3 \wedge A_4)) \in [0,1]$. This suggests that if we are willing to contemplate assessments based on material implication, we should be willing to entertain interval probabilities from the outset. We leave such a discussion for the future, noting here that existing languages such as Poole's Independent Choice Logic (ICL) [23] do have material implication in the syntax, but often adopt special semantics to guarantee sharp probabilities.

4 Relational Credal Networks

Many phenomena in real life depict repetitive patterns. For instance, social networks involve many individuals, several of which may share common characteristics. Epidemiological events may also bring together similar individuals; temporal sequences modeled by hidden Markov models often display similarities across time steps. There are indeed several formalisms that capture repetition in Bayesian network fragments [15, 16, 24, 25]. The simplest strategy is to allow random variables to be parameterized; for instance, we might extend the specification in the previous section as follows:

$$\mathbb{P}(X_1(\mathsf{x}) = 1) \geq 1/2, \tag{1}$$

$$\mathbb{P}(X_2(\mathsf{x}) = 1) \in [1/4, 1/3], \tag{2}$$

$$\mathbb{P}\Big(X_3(\mathsf{x}, \mathsf{y}) = 1\Big) = 1/5, \tag{3}$$

$$X_4(\mathsf{x}) \Leftrightarrow X_1(\mathsf{x}) \wedge X_2(\mathsf{x}), \tag{4}$$

$$X_5(\mathsf{x}, \mathsf{y}) \Leftrightarrow X_3(\mathsf{x}, \mathsf{y}) \wedge X_4(\mathsf{x}). \tag{5}$$

At this point we can simply refer to $\mathsf{x}, \mathsf{y}, \dots$ as *logical variables*, and to $X_1(\mathsf{x}), X_2(\mathsf{x}), X_3(\mathsf{x}, \mathsf{y})$ as *relations*. Again we write sentences that mix variables and Boolean operators. We say that $X(\mathsf{x}_1, \dots, \mathsf{x}_k)$, where each x_i is either a logical variable or an individual, is an *atom*. An atom with no logical variable is a *ground atom*.

We can then extend our specification framework as follows.

We assume that a directed acyclic graph is given, where each node is a relation, and that every k-ary relation X is associated with either:

- a logical assessment

  $$X(\mathsf{x}_1, \dots, \mathsf{x}_k) \Leftrightarrow F(\mathsf{x}_1, \dots, \mathsf{x}_k, Y_1, \dots, Y_m),$$

  where F is a formula with free logical variables $\mathsf{x}_1, \dots, \mathsf{x}_k$, and possibly with other logical variables that are bound, and where each Y_i is either a relation or an individual; or

- a probabilistic assessment

  $$\mathbb{P}(X(\mathsf{x}_1, \dots, \mathsf{x}_k) = 1) \in [\alpha, \beta],$$

  where α and β are nonnegative rationals in $[0,1]$.

We assume that our languages consist of subsets of function-free first-order logic (referred to as FFFO). Hence we allow existential and universal quantifiers in our syntax.

Concerning the semantics, as often happens when one moves from sharp to interval probabilities, there is more than one way to interpret assessments. In our setting, there are two sensible semantics for well-formed specifications, as we now discuss.

We assume that we have a set $\mathcal{D}$, the *domain*. In this paper every domain is finite, with N elements. Every individual refers to an element of the domain. We will always adopt the *rigidity* assumption that is common in probabilistic logic [3]; that is, we will always assume that the interpretation of individuals is constant across interpretations for a fixed domain. That is, the individual Ann is always mapped to the same element of $\mathcal{D}$, whatever the interpretation of relations. Hence our individuals can be identified with elements of the domain, and given labels such as $1, 2, \dots, N$.

For instance, suppose we take assessments (1)–(5), and a domain with two individuals, say Ann and Bob, respectively denoted by a and b. We have several ground atoms: $X_1(a)$, $X_1(b)$, $X_2(a)$, $X_2(b)$, $X_3(a,a)$, $X_3(a,b)$, and so on. Consider a graph where each ground atom is a node, and where an edge is inserted between two nodes if an edge was present between the relations. In our example, we obtain the graph in Figure 3. Note that grounding produced two disjoint graphs in this case. However, suppose we keep assessments (1)–(4), but we turn X_5 into a unary relation such that:

$$X_5(\mathsf{x}) \Leftrightarrow \forall \mathsf{y} : X_3(\mathsf{x}, \mathsf{y}) \wedge X_4(\mathsf{y}). \tag{6}$$

Then grounding takes us to Figure 4.

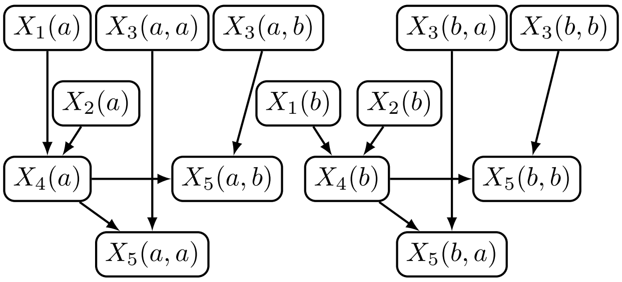


Figure 3: Grounding assessments (1)–(5) with respect to domain $\mathcal{D} = \{a, b\}$.

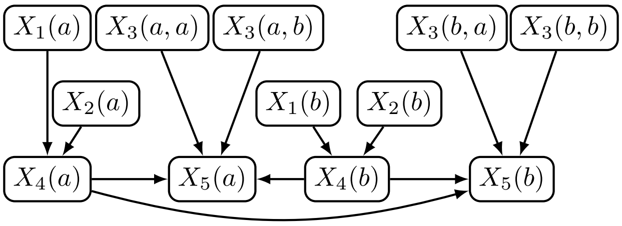


Figure 4: Grounding assessments (1)–(4) and (6) with respect to domain $\mathcal{D} = \{a, b\}$.

So far, our procedure to produce a single grounded graph out of the logical assessments and a fixed domain seems uncontroversial. Now consider the probability assessments; for instance, take

$$\mathbb{P}(X_1(\mathbf{x})) \in [1/2, 1].$$

What does it mean? Does it mean that

- for each $\gamma \in [1/2, 1]$,

  $$\forall \mathbf{x} \in \mathcal{D} : \mathbb{P}(X_1(\mathbf{x})) = \gamma$$

  is a possible assessment, or that

- for each $\mathbf{x} \in \mathcal{D}$,

  $$\mathbb{P}(X_1(\mathbf{x})) = \gamma$$

  is a possible assessment for each $\gamma \in [1/2, 1]$?

The difference between these two interpretations is substantial, even though both share the same grounded graph (for given N). In the first interpretation the assessments are viewed as a set of relational Bayesian networks. That is, each selection of probability values defines a relational Bayesian network, that itself can be grounded into a Bayesian network given a domain. For assessments (1)–(5), we have 4 extreme Bayesian networks that are generated given $\mathcal{D} = \{a, b\}$; we have for instance an extreme Bayesian network where $\mathbb{P}(X_1(a)) = \mathbb{P}(X_1(b)) = 1/2$, and also we have a Bayesian network where $\mathbb{P}(X_1(a)) = \mathbb{P}(X_1(b)) = 1$ (but we do not have $\mathbb{P}(X_1(a)) = 1/2$ and $\mathbb{P}(X_1(b)) = 1$). In the second interpretation the assessments directly yield a credal network with separately specified local credal sets. In our example, the latter semantics yields grounded assessments

$$\mathbb{P}(X_1(a) = 1) \geq 1/2, \mathbb{P}(X_2(b) = 1) \geq 1/2,$$

$$\mathbb{P}(X_2(a) = 1) \in [1/4, 1/3], \ldots, \mathbb{P}(X_3(b, b) = 1) = 1/5,$$

and there are 16 extreme Bayesian networks given $\mathcal{D} = \{a, b\}$.

We will refer to a set of well-formed assessments as a *relational credal network*. When the first semantics is adopted, we say that the relational credal network has *coupled parameters*; when the latter semantics is used, we say the relational network has *decoupled parameters*. To simplify the language, we often refer to *coupled relational credal networks* and *decoupled relational credal networks*.

5 The Complexity of Relational Languages

We can now consider inference problems for selected relational languages $\mathcal{L}$. The input to our inference problems is a relational credal network, evidence, and the size of the domain, denoted by N. We assume that the arity of all relations is bounded.

Domain size N can be given either in binary or unary encoding. In computational terms, binary encoding for N implies that almost every calculation requires exponential effort (as there may be exponentially long numbers in the output) [9]. For this reason, it makes sense to assume that N is specified in unary notation. So, we denote by $\mathsf{INF_d}(\mathcal{L})$ and by $\mathsf{INF_d}^+(\mathcal{L})$ respectively the decision problems for language $\mathcal{L}$, as before, for unary N, where the query Q is an assignment to a grounded atom, and evidence $\mathbf{E}$ is a set of assignments for grounded atoms (evidence is understood as the conjunction of those assignments). Recall that all relations have bounded arity (and the bound is known). Note that for relatively simple languages we already have $\mathsf{NP^{PP}}$-complete inference, from the results for propositional languages (Section 3).

Consider then function-free first-order logic (we refer to it by FFFO). The following result is not surprising:

Theorem 3 $\mathsf{INF_d}^+(\mathsf{FFFO})$ *is* $\mathsf{NP^{PP}}$*-complete both for decoupled and for coupled relational credal networks.*

Proof. For pertinence, ground the relational credal network into a credal network specified using $\mathsf{Prop}(\wedge, \neg)$. Inference in the grounded credal network is a $\mathsf{NP^{PP}}$-complete problem. For hardness, note that a domain with a single individual can already define an arbitrarily complex credal network. □

To obtain more insightful results concerning complexity, we have previously proposed an analysis with respect to *data complexity* and to *domain complexity* [9]. We have started such an analysis for relational Bayesian networks, and we now present results for relational credal networks.

We refer to the complexity of computing a conditional probability, given a relational credal network, evidence (a set of assignments), and an integer N (the size of the domain in unary notation), as the *combined complexity*. Theorem 3 deals with combined complexity. We refer to the complexity of computing a conditional probability, for a fixed relational network, when evidence and N are inputs, as the *data complexity*. And we refer to the complexity of computing a conditional probability, for a fixed relational network and fixed evidence, when N is the input, as the *domain complexity*.[2]

When we focus on relational Bayesian networks, the data complexity of FFFO is PP-complete [9]. So the combined and data complexities are identical for relational Bayesian networks as far as first-order logic is concerned. For relational credal networks the data complexity depends on the semantics, as we now show: as often happens when one moves from sharp to indeterminate probabilities, concepts that collapse in the former case do not collapse in the latter case, and we must deal with more nuanced scenarios.

We use $\mathsf{DINF_d}(\mathcal{L})$ to indicate the data complexity of relational credal networks specified through language $\mathcal{L}$. We can state our main results:

Theorem 4 $\mathsf{DINF_d}(\mathsf{FFFO})$ *is* $\mathsf{NP^{PP}}$*-complete for decoupled relational credal networks.*

Proof. For decoupled relational credal networks, pertinence to $\mathsf{NP^{PP}}$ is easy (even the combined complexity is in $\mathsf{NP^{PP}}$ by Theorem 3). To prove hardness, we adapt the proof for a similar result for Bayesian networks [9]. Take an E-MAJSAT problem with pair (ϕ, k), where ϕ is in CNF with m clauses, each one of them with three literals (for each clause, we refer to the "left" literal, the "middle" literal, and the "right" literal). Suppose propositions are $A_1, \dots, A_n$. If the number of clauses m is smaller than n, then add trivial clauses such as $A_1 \vee A_1 \vee \neg A_1$ until $m = n$. These clauses do not change the output of MAJSAT. If instead $n < m$, then add fresh propositions $A_{n+1}, \dots, A_m$. These propositions do not change the output of MAJSAT. Introduce unary relations $\mathsf{sat}(\mathbf{x})$ and $\mathsf{choice}(\mathbf{x})$; impose $\mathbb{P}(\mathsf{sat}(\mathbf{x})) = 1/2$, $\mathbb{P}(\mathsf{choice}(\mathbf{x})) \in [0,1]$. The idea is that $\mathsf{sat}(\mathbf{x})$ refers to proposition $A_{\mathbf{x}}$ for $\mathbf{x} \in \{k+1, \dots, n\}$, while $\mathsf{choice}(\mathbf{x})$ refers to proposition $A_{\mathbf{x}}$ for $\mathbf{x} \in \{1, \dots, k\}$. Introduce binary relations $\mathsf{aux}^{\mathsf{sat}}_{ij}(\mathbf{x}, \mathbf{y})$ and $\mathsf{aux}^{\mathsf{choice}}_{ij}(\mathbf{x}, \mathbf{y})$, where i can be either left, middle, and right, while j can be either $+$ or $-$. Adopt $\mathbb{P}\left(\mathsf{aux}^{\mathsf{sat}}_{ij}(\mathbf{x}, \mathbf{y})\right) = \mathbb{P}\left(\mathsf{aux}^{\mathsf{choice}}_{ij}(\mathbf{x}, \mathbf{y})\right) = \alpha$ for some $\alpha \in (0,1)$; the specific value of α will not matter. To be concrete, adopt $\alpha = 1/2$. Also, introduce auxiliary relations $\mathsf{literal}_i(\mathbf{x})$ where i can be left, middle, right. Impose

$$\begin{aligned}\mathsf{literal}_i(\mathbf{x}) \Leftrightarrow\ & (\exists \mathbf{y} : \mathsf{aux}^{\mathsf{sat}}_{i+}(\mathbf{x}, \mathbf{y}) \wedge \mathsf{sat}(\mathbf{y})) \\ \vee\ & (\exists \mathbf{y} : \mathsf{aux}^{\mathsf{sat}}_{i-}(\mathbf{x}, \mathbf{y}) \wedge \neg\mathsf{sat}(\mathbf{y})) \\ \vee\ & (\exists \mathbf{y} : \mathsf{aux}^{\mathsf{choice}}_{i+}(\mathbf{x}, \mathbf{y}) \wedge \mathsf{choice}(\mathbf{y})) \\ \vee\ & (\exists \mathbf{y} : \mathsf{aux}^{\mathsf{choice}}_{i-}(\mathbf{x}, \mathbf{y}) \wedge \neg\mathsf{choice}(\mathbf{y})).\end{aligned}$$

Introduce unary relation $\mathsf{clause}(\mathbf{x})$ and impose

$$\mathsf{clause}(\mathbf{x}) \Leftrightarrow \mathsf{literal}_{\mathsf{left}}(\mathbf{x}) \vee \mathsf{literal}_{\mathsf{middle}}(\mathbf{x}) \vee \mathsf{literal}_{\mathsf{right}}(\mathbf{x}).$$

Finally, introduce query and

$$\mathsf{query} \Leftrightarrow \forall \mathbf{x} : \mathsf{clause}(\mathbf{x}).$$

Take $N = n$; given our previous discussion we have $n = m$. Individuals are referred as $\{1, \dots, N\}$ and have a dual purpose, indexing both propositions and clauses.

Take evidence $\mathbf{E}$ as follows. For the ith clause, suppose the left literal is A_j. If $j > k$, set $\mathsf{aux}^{\mathsf{sat}}_{\mathsf{left}+}(i, j)$ to true, and all other $\mathsf{aux}^{\mathsf{sat}}_{\mathsf{left}+}(i, \mathbf{y})$ to false; also set all $\mathsf{aux}^{\mathsf{sat}}_{\mathsf{left}-}(i, \mathbf{y})$ to false, all $\mathsf{aux}^{\mathsf{choice}}_{\mathsf{left}+}(i, \mathbf{y})$ to false, and all $\mathsf{aux}^{\mathsf{choice}}_{\mathsf{left}-}(i, \mathbf{y})$ to false. If instead $i \leq k$, set $\mathsf{aux}^{\mathsf{choice}}_{\mathsf{left}+}(i, j)$ to true, and all other $\mathsf{aux}^{\mathsf{choice}}_{\mathsf{left}+}(i, \mathbf{y})$ to false; also set all $\mathsf{aux}^{\mathsf{choice}}_{\mathsf{left}-}(i, \mathbf{y})$ to false, all $\mathsf{aux}^{\mathsf{sat}}_{\mathsf{left}+}(i, \mathbf{y})$ to false, and all $\mathsf{aux}^{\mathsf{sat}}_{\mathsf{left}-}(i, \mathbf{y})$ to false.

Suppose instead that for the ith clause the left literal is $\neg A_j$; follow the previous paragraph, but exchange $+$ and $-$. Repeat similarly for middle and right literals, but using middle and right as appropriate. Finally, decide whether $\overline{\mathbb{P}}(\mathsf{query}(1) = 0) > 1/2$. If YES, the E-MAJSAT problem is accepted, if NO, it is not accepted. Hence we have the desired reduction. □

Theorem 5 $\mathsf{DINF_d}(\mathsf{FFFO})$ *is* PP*-complete for coupled relational credal networks.*

Proof. For coupled relational credal networks, PP-hardness is obtained by encoding any E-MAJSAT problem with $k = 0$ in the previous proof, and noting that MAJSAT is a PP-complete problem [27]. To prove pertinence to PP, we will use the fact that PP is

[2] Data and domain complexity are respectively related to the existing notions of lqe-liftability and liftability [17, 28]; lqe-liftability means that data complexity is polynomial, and liftability means that domain complexity is polynomial.

closed under union, a celebrated result in complexity theory [4]. Take a fixed relational credal network and note that there is a fixed number (possibly large) of relational Bayesian networks that can be generated by selecting each one of the possible endpoints of probability intervals. Each one of these M relational Bayesian networks specifies a set of strings, consisting of those strings containing N and associated evidence, such that the inference problem yields YES if a string is accepted. That is, we have M sets of accepted strings, and our problem is: given a string with N and evidence, accept it if any one of those M sets of strings contains it. But note that each set of strings defines a PP decision problem (the problem of accepting the strings), as each relational Bayesian network can be grounded into a polynomially larger Bayesian network, and inference can be conducted in the latter network. So our main problem is to consider a set of strings that is the union of the M set of strings; because PP is closed under union, the main problem is in PP as well. □

As noted previously, PP is the class of problems that can be solved by "majority" Turing machines with a polynomial bound; they are usually related to counting problems [20]. And $\mathsf{NP}^{\mathsf{PP}}$ is the class of problems that can be solved by a nondeterministic Turing machine with a polynomial bound, with the "help" of an oracle that returns the solution PP given problems. Intuitively, we should expect the latter problems to be significantly more taxing than the former (but current literature does not seem to have a result on whether they are different or not).

6 Conclusion

We have explored the balance of expressivity and complexity in Boolean credal networks. We have recently proposed a framework for such an analysis, geared to Bayesian networks [9]; this paper is a first step in extending the framework to credal networks.

We have discussed both propositional and relational languages, and for relational languages we have studied combined and data complexities. Theorem 1 reveals a class of credal networks that admits polynomial inference, a property shared by few other classes [10]; the result is surprising in that it reproduces the polynomial character of Bayesian networks under the same language. And in the opposite direction, Theorems 4 and 5 show distinctions between Bayesian and credal networks, as in the latter there is more than one reasonable semantics to choose from, and the choice does have an impact on complexity. Surprisingly, for coupled relational credal networks the data complexity is identical to the data complexity of relational Bayesian networks.

Perhaps the most compelling aspect of our framework is the number of questions it raises. Consider a simple fact. It is usually assumed that one can arbitrarily choose between computing an upper or a lower probability, as they are directly related by $\overline{\mathbb{P}}(A|B) = 1 - \underline{\mathbb{P}}(A^c|B)$ [29]. But if a language does not have negation, it may not be possible to formulate $\underline{\mathbb{P}}(A^c|B)$ as a query, and it may then be harder to produce a lower probability than an upper probability. This sort of phenomena can only be explored when we pay attention to languages. In fact, the key difference between Bayesian and credal networks is the language that is used to express assessments.

There are many languages to explore concerning the complexity of credal networks. There are several fragments of function-free first-order logic that are widely used, such as monadic logic [5]; there are guarded fragments and description logics such as DL-Lite, $\mathcal{EL}$, $\mathcal{ALC}$ [2]; there are also languages based on second-order logic and various modal logics. For all these logical languages, one can ask combined and data complexity, not only for inference, but also for other problems of common interest such as maximum a posteriori configurations (MAP). All such problems await detailed investigation.

Acknowledgements

The first author is partially supported by CNPq (grant 305395/2010-6) and the second author is supported by FAPESP (grant 2013/23197-4).

References

[1] Thomas Augustin, Frank P. A. Coolen, Gert de Cooman, and Matthias C. M. Troffaes. *Introduction to Imprecise Probabilities.* Wiley, 2014.

[2] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, and P.F. Patel-Schneider. *Description Logic Handbook.* Cambridge University Press, 2002.

[3] F. Bacchus. *Representing and Reasoning with Probabilistic Knowledge: A Logical Approach.* MIT Press, Cambridge, 1990.

[4] Richard Beigel, Nick Reingold, and Daniel Spielman. PP is closed under intersection. *Journal of Computer and System Sciences*, 50(2):191–202, 1995.

[5] Egon Börger, Erich Grädel, and Yuri Gurevich. *The Classical Decision Problem*, Springer, 1997.

[6] Fabio G. Cozman. Graphical models for imprecise probabilities. *International Journal of Approximate Reasoning*, 39(2-3):167–184, 2005.

[7] Fabio G. Cozman. Sets of probability distributions, independence, and convexity. *Synthese*, 186(2):577–600, 2012.

[8] Fabio G. Cozman. Independence for full conditional probabilities: Structure, factorization, non-uniqueness, and Bayesian networks. *International Journal of Approximate Reasoning*, 54:1261–1278, 2013.

[9] Fabio G. Cozman and Denis Deratani Mauá. Bayesian networks specified using propositional and relational constructs: Combined, data, and domain complexity. In *AAAI Conference on Artificial Intelligence*, pages 3519–3525, 2015.

[10] Cassio Polpo de Campos and Fabio G. Cozman. The inferential complexity of Bayesian and credal networks. In *International Joint Conference on Artificial Intelligence*, pages 1313–1318, Edinburgh, United Kingdom, 2005.

[11] Gert de Cooman and Enrique Miranda. Independent natural extension for sets of desirable gambles. *Journal of Artificial Intelligence Research*, 45:601–640, 2012.

[12] Gert de Cooman, Enrique Miranda, and Marco Zaffalon. Independent natural extension. *Artificial Intelligence*, 175:1911–1950, 2011.

[13] E. Fagiuoli and Marco Zaffalon. 2U: An exact interval propagation algorithm for polytrees with binary variables. *Artificial Intelligence*, 106(1):77–107, 1998.

[14] Lance Fortnow. Counting complexity. In L. Hemaspaandra and A. Selman, editors, *Complexity Theory Retrospective II*, pages 81–107, Springer, 1997.

[15] Lise Getoor and Ben Taskar. *Introduction to Statistical Relational Learning*. MIT Press, 2007.

[16] Manfred Jaeger. Relational Bayesian networks. *Conference on Uncertainty in Artificial Intelligence*, pages 266–273, 1997.

[17] Manfred Jaeger and Guy Van Den Broeck. Liftability of probabilistic inference: Upper and lower bounds. In *2nd Statistical Relational AI (StaRAI-12) Workshop*, 2012.

[18] Isaac Levi. *The Enterprise of Knowledge*. MIT Press, Cambridge, Massachusetts, 1980.

[19] Michael L. Littman, Judy Goldsmith, and Martin Mundhenk. The computational complexity of probabilistic planning. *Journal of Artificial Intelligence Research*, 9:1–36, 1998.

[20] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley Publishing, 1994.

[21] Judea Pearl. *Causality: models, reasoning, and inference*. Cambridge University Press, Cambridge, United Kingdom, 2000.

[22] David Poole. Probabilistic Horn abduction and Bayesian networks. *Artificial Intelligence*, 64:81–129, 1993.

[23] David Poole. The independent choice logic for modelling multiple agents under uncertainty. *Artificial Intelligence*, 94(1/2):7–56, 1997.

[24] David Poole. First-order probabilistic inference. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 985–991, 2003.

[25] Luc De Raedt. *Logical and Relational Learning*. Springer, 2008.

[26] Taisuke Sato and Yoshitaka Kameya. Parameter learning of logic programs for symbolic-statistical modeling. *Journal of Artificial Intelligence Research*, 15:391–454, 2001.

[27] Leslie G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing* 8(3):410–421, 1979.

[28] Guy van den Broeck. On the completeness of first-order knowledge compilation for lifted probabilistic inference. In *Neural Processing Information Systems*, 2011.

[29] Peter Walley. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, London, 1991.

[30] Kurt Weichselberger. The theory of interval-probability as a unifying concept for uncertainty. International Journal of Approximate Reasoning 24(2-3):149–170, 2000.