Complexity of Inferences in Polytree-Shaped Semi-Qualitative Probabilistic Networks

Cassio P. de Campos
Dalle Molle Institute for Artificial Intelligence
IDSIA - Galleria 2
Manno 6928, Switzerland

Fabio G. Cozman
University of São Paulo
Av. Prof. Mello Moraes, 2231
São Paulo, SP, Brazil

Abstract

Semi-qualitative probabilistic networks (SQPNs) merge two important graphical model formalisms: Bayesian networks and qualitative probabilistic networks. They provide a very general modeling framework by allowing the combination of numeric and qualitative assessments over a discrete domain, and can be compactly encoded by exploiting the same factorization of joint probability distributions that are behind the Bayesian networks. This paper explores the computational complexity of semi-qualitative probabilistic networks, and takes the polytree-shaped networks as its main target. We show that the inference problem is coNP-Complete for binary polytrees with multiple observed nodes. We also show that inferences can be performed in time linear in the number of nodes if there is a single observed node. Because our proof is constructive, we obtain an efficient linear time algorithm for SQPNs under such assumptions. To the best of our knowledge, this is the first exact polynomial-time algorithm for SQPNs. Together these results provide a clear picture of the inferential complexity in polytree-shaped SQPNs.

Introduction

Qualitative probabilistic networks abstract the precise probability values that are mandatory in Bayesian networks. Instead of displaying precise values, a qualitative probabilistic network (QPN) only states algebraic relations among probability values (Druzdzel and Henrion 1993b; Wellman 1990). There are several efficient algorithms for QPNs (Druzdzel and Henrion 1993b), including algorithms for multiple observations (Renooij, van der Gaag, and Parsons 2000), ambiguous signs (Bolt, Renooij, and van der Gaag 2003), non-monotonic influences (Renooij and van der Gaag 2000) and other relations (Bolt, van der Gaag, and Renooij 2004; 2003; Renooij and Witteman 1999; Renooij, van der Gaag, and Parsons 2002b; van der Gaag, Bodlaender, and Feelders 2004). Parsons and Dohnal (1993) and Renooij and van der Gaag (2002) have proposed semi-qualitative probabilistic networks (SQPN) that mix quantitative and qualitative assessments. For SQPNs the computation of *exact* inferences is generally a more complex undertaking (de Campos and Cozman 2005), and the only existing algorithms either are fast and focus on approximate solutions (Renooij and van der Gaag 2002) or are very inefficient in all but a few particular cases (de Campos and Cozman 2005).

This paper focuses on polytree-shaped SQPNs, that is, networks whose underlying graph after dropping arc directions has no cycles. We present two main results regarding the computational complexity of inferences in such networks: (1) a polynomial-time algorithm for inferences where a single node of the network is observed – to the best of our knowledge, this is the first exact polynomial-time algorithm for SQPNs; (2) a proof that SQPN inferences in binary polytrees with multiple observations is coNP-Complete.

Qualitative Probabilistic Networks

A QPN consists of an acyclic digraph, a set of random variables $\mathcal{X}$ where each variable is associated to a node in the graph, and a collection of constraints on probability values. The digraph conveys a Markov condition: every variable is independent of its nondescendants nonparents given its parents. We start by assuming that variables are binary, with a "higher" value (indicated by x^1) and a "lower" value (indicated by x^0). This is not an intrinsic limitation of QPNs, but it is arguably the most valuable scenario. Nodes and their variables are used interchanged in our discussion. To simplify notation, we denote an event $\{X = x\}$ by x whenever the meaning is clear. The state space of a random variable $X \in \mathcal{X}$ is denoted by Ω_X, while $\mathbf{x} \in \Omega_{\mathbf{X}} = \times_{X \in \mathbf{X}} \Omega_X$, is said to be an instantiation of variables $\mathbf{X} \subseteq \mathcal{X}$.

Constraints in a QPN derive from qualitative *influences* and *synergies* among probability values (Wellman 1990). An *influence* between two variables expresses how the values of one variable influence the probabilities of the values of the other variable. For instance, a *positive influence* of variable A on its effect B expresses that observing higher values for A makes higher values for B more likely, regardless of any other direct influence on B. *Negative influences* and *zero influences* are defined analogously.

Definition 1 *Let $A, B \in \mathcal{X}$ and $\mathbf{X} \subseteq \mathcal{X} \setminus \{A, B\}$. An influence of A on B with respect to variables $\mathbf{X}$, denoted by $S^I(A, B, \mathbf{X})$ (with I equals to $+, -, 0$ for positive, negative and zero influences, respectively) means that*

$$\forall \mathbf{x} \in \Omega_{\mathbf{X}} : \ P\big(b^1 | a^1, \mathbf{x}\big) \, R^I P\big(b^1 | a^0, \mathbf{x}\big) \,, \tag{1}$$

where R^I is $\geq$,$\leq$,$=$, respectively. In case none of them holds, we say that the influence is ambiguous, and denote it by $S^?(A, B, \mathbf{X})$.

Synergies represent interactions among influences. An *additive synergy* between three variables expresses how the values of two variables jointly influence the probabilities of the values of the third variable. For instance, a *positive additive synergy* of variables A and B on their common effect C expresses that the joint influence of A and B on C is greater than the sum of their separate influences, regardless of other influences on C.

Definition 2 *Let $A, B, C \in \mathcal{X}$ and $\mathbf{X} \subseteq \mathcal{X} \setminus \{A, B, C\}$. An additive synergy of A and B on C with respect to variables $\mathbf{X}$, denoted by $Y^I(\{A, B\}, C, \mathbf{X})$ (with I equals to $+, -, 0$ for positive, negative and zero synergies, respectively) means that*

$$\forall \mathbf{x} \in \Omega_{\mathbf{X}} : \qquad P\big(c^1|a^1, b^1, \mathbf{x}\big) + P\big(c^1|a^0, b^0, \mathbf{x}\big) \, R^I \; P\big(c^1|a^0, b^1, \mathbf{x}\big) + P\big(c^1|a^1, b^0, \mathbf{x}\big) \, , \quad (2)$$

where where R^I is $\geq$,$\leq$,$=$, respectively. In case none of them holds, we say that the additive synergy is ambiguous, and denote it by $Y^?(\{A, B\}, C, \mathbf{X})$.

Definition 3 *A qualitative probabilistic network (QPN) $\mathcal{N} = (\mathcal{G}, \mathcal{D})$ consists of an acyclic digraph $\mathcal{G}$ with nodes associated to binary random variables $\mathcal{X}$ and a Markov condition given by the graph, where each variable is independent of its nondescendants nonparents given its parents; and a set of qualitative assessments $\mathcal{D}$ of the form $S^I(A, B, \mathbf{X})$, with[1] $\mathrm{pa}_{\mathcal{G}}(B) = \mathbf{X} \cup \{A\}$, or $Y^I(\{A, B\}, C, \mathbf{X})$, with $\mathrm{pa}_{\mathcal{G}}(C) = \mathbf{X} \cup \{A, B\}$. If an assessment is not given between variables and their parents, it is assumed to be ambiguous (nodes without parents in $\mathcal{G}$, called* root nodes*, are assumed to be ambiguous w.r.t. all qualitative assessments).*

There are several extensions to the QPNs just defined (Bolt, Renooij, and van der Gaag 2003; Bolt, van der Gaag, and Renooij 2003; Renooij, van der Gaag, and Parsons 2000; Renooij and van der Gaag 2000), including *product synergies* which express how the value of a variable influences the probabilities of the values of another variable given the value of a third variable, *non-monotonic* influences of a variable on a child *provoked* by a sibling, *situational signs* that capture the sign of a non-monotonic influence, besides the *enhanced* formalism for qualitative networks (Renooij and Witteman 1999). Our complexity results extend to networks that allow most of these extra qualitative statements, even though we do not explicitly account for them in this paper.

An SQPN consists of an acyclic digraph with nodes associated to variables and a Markov condition, where each node A is either associated to conditional distributions $P(A|\mathrm{pa}(A))$, or associated to qualitative statements from QPNs (this encompasses other definitions found in literature (Parsons and Dohnal 1993; Renooij and van der Gaag 2002)). Thus SQPNs offer a combination of QPNs and Bayesian networks. One might hope that such a combination would not be harder than the hardest of its components; that is, no harder than Bayesian networks. In fact SQPNs are harder than QPNs and Bayesian networks, and these two types of networks should be viewed as lower complexity special cases of the former (de Campos and Cozman 2005).

[1]The parents of a node Z in $\mathcal{G}$ are denoted by $\mathrm{pa}_{\mathcal{G}}(Z)$.

Definition 4 *A semi-qualitative probabilistic network $\mathcal{N} = (\mathcal{G}, \mathcal{D})$ consists of an acyclic digraph $\mathcal{G}$ with nodes associated to binary random variables $\mathcal{X}$ and a Markov condition given by the graph; a set of assessments $\mathcal{D}$ of two types: qualitative assessments of the form $S^I(A, B, \mathbf{X})$, with $\mathrm{pa}_{\mathcal{G}}(B) = \mathbf{X} \cup \{A\}$, or $Y^I(\{A, B\}, C, \mathbf{X})$, with $\mathrm{pa}_{\mathcal{G}}(C) = \mathbf{X} \cup \{A, B\}$; or quantitative assessments specified by conditional probability mass functions $P\big(A|\mathrm{pa}_{\mathcal{G}}(A)\big)$ (specified as numbers in $[0, 1]$ for every $\pi \in \Omega_{\mathrm{pa}_{\mathcal{G}}(A)}$), with the restriction that each variable has either qualitative or quantitative assessments associated to it (but not both). Furthermore, each root node A has either a marginal mass function $P(A)$ associated to it, or is ambiguous (as in a QPN).*

A (S)QPN $\mathcal{N}$ can be viewed as a compact way to represent all joint probability mass functions P that satisfy its constraints. Hence, we employ the notation $P \in \mathcal{N}$ to indicate that P is in accordance with all qualitative and quantitative assessments of $\mathcal{N}$, and respects

$$P(\mathbf{x}) = \prod_{i=1}^{n} P\big(X_i = x_i | \mathrm{pa}_{\mathcal{G}}(X_i) = \pi_i\big) \, ,$$

with all x_i, π_i, for every i, conforming with $\mathbf{x} \in \Omega_{\mathcal{X}}$.

Generally, an *inference* in a (S)QPN refers to the qualitative question of how the observation of some variables changes the probability of a query variable. Suppose Q is the query variable and $\mathbf{e} \in \Omega_{\mathbf{E}}$ is our observed event for the set of variables $\mathbf{E}$. We need to evaluate $P\big(q^1|\mathbf{e}\big) - P\big(q^1\big)$. The idea is to identify the type of influence that evidence $\mathbf{e}$ has on Q. When $P\big(q^1|\mathbf{e}\big) - P\big(q^1\big)$ is non-positive for every $P \in \mathcal{N}$, we have a *negative influence*[2] of $\mathbf{e}$ over Q. If $P\big(q^1|\mathbf{e}\big) - P\big(q^1\big)$ is always non-negative, then a positive influence exists. If both hold, then $P\big(q^1|\mathbf{e}\big) = P\big(q^1\big)$ and we have a zero influence. Otherwise, we have an ambiguous influence. The formal definitions are as follows.

Definition 5 *Let $\mathcal{N} = (\mathcal{G}, \mathcal{D})$ be an (S)QPN specified by rational numbers, $Q \in V_{\mathcal{G}}$ be a variable of the domain and $\mathbf{e} \in \Omega_{\mathbf{E}}$, with non-empty $\mathbf{E} \subseteq V_{\mathcal{G}} \setminus \{Q\}$, be an observation of some variables. We name as* negative influence query*, or* `(S)QPN-NEGINF`*($\mathcal{N}, Q, \mathbf{e}$) for short, the task of deciding whether $\forall P \in \mathcal{N} : P\big(q^1|\mathbf{e}\big) \leq P\big(q^1\big)$. We name as* positive influence query*, or* `(S)QPN-POSINF`*($\mathcal{N}, Q, \mathbf{e}$) for short, the task of deciding whether $\forall P \in \mathcal{N} : P\big(q^1|\mathbf{e}\big) \geq P\big(q^1\big)$.*

Before any computational complexity analysis, we must define the size to encode the input of our queries, and for

[2]Strictly speaking, this is a "non-positive" influence, because a zero influence would satisfy the assertion. We abuse notation using the name negative.

that we need the size to encode an SQPN $\mathcal{N} = (\mathcal{G}, \mathcal{D})$. Let $|\mathcal{X}| = |V_\mathcal{G}| = n$ be the overall number of random variables. We parametrize our analysis by the maximum number of parents a node may have in $\mathcal{G}$, called d, and we consider it to be "small" unless explicitly said otherwise. We do so because a large number of parents per node may lead to an exponentially large input, as $\Theta(2^{|\mathrm{pa}_\mathcal{G}(A)|})$ numbers[3] are needed to specify the mass functions of a node A, and even an exponential-time algorithm (in n) could be considered to be polynomial in the input size if a node is allowed to have $\Theta(n)$ parents. Hence, the conditional mass functions that are given for a node have at most $O(2^d)$ numbers. Nodes without quantitative assessments may have qualitative ones. We assume that all qualitative assessments can be written using at most $O(n2^d)$ numbers (this is achievable by using a reasonable encoding). The digraph of the SQPN can be encoded using $\Theta(nd)$ numbers, and so we can assume that the whole SQPN is encoded by $\Theta(n2^d)$ numbers (a more precise analysis could consider the exact number of parents per node, but it would lead us to the same conclusions, as it will become apparent from the theorems later). In terms of bits, we define as $\mathrm{SZ}(\mathcal{N}) = \sum_{i=1}^{N} \mathrm{sz}(N_i)$ the size of the SQPN $\mathcal{N}$, where N_i represents its i-th number, with $N \in \Theta(n2^d)$ its total number of numbers. The size $\mathrm{sz}(N_i)$ of a rational number N_i is given by $\lceil \log_2 N_i \rceil$, while the size of a recursive (i.e. computable) real number N_i is defined by the size to encode a deterministic Turing Machine (DTM) that, given b, is capable of producing a rational r such that $|r - N_i| < 2^{-b}$ (i.e. at least b bits of precision of N_i) in time $poly(b)$. When clear from the context, we write SZ without its argument $\mathcal{N}$.

Complexity Results

A polynomial inference algorithm for QPNs has been proposed by Druzdzel and Henrion (1993a) with observation in a single node, later extended by Renooij et al. (2002a) to handle multiple observations: the algorithm generates a sign for each variable, implied by the observations in the QPN, and these signs are precisely the answer to the desired queries. For SQPNs, the situation is much more complicated, and different network topologies lead to distinct hardness of inference. De Campos and Cozman (2005) showed that the general inference in SQPNs is $\mathsf{NP}^{\mathsf{PP}}$-hard (although the definition of inference used there is slightly less precise than the one we use here). This leaves no hope for truly efficient algorithms for general SQPNs. Because of such hardness, here we focus on *polytrees*, when the underlying graph of $\mathcal{G}$ after dropping arc directions has no cycles.

The first part of this section demonstrates that, in a loose sense, inferences in SQPNs with some real numbers are not harder than SQPNs with rationals. This holds for any SQPN (not only polytrees) and directly extends to Bayesian networks too, which can be seen as a subcase of SQPNs where all nodes are defined through quantitative assessments. The result is important for the proof of Theorem 11 presented later on. The second part concerns polytree SQPNs with a single observed variable. We show constructively that inferences can be performed in polynomial time. In fact, the described algorithm is very efficient and runs in linear time in n. This algorithm is probably the best result in terms of tractability (i.e. exact polynomial-time algorithms) for which we can hope, as the third part of this section shows that allowing multiple observations already leads to a hardness-of-inference result.

Recursive Real versus Rational Numbers

The following results regard the numerical precision of SQPNs' specifications. They show that recursive real numbers provide no extra power (in a loose sense) to the problem with respect to using only rational numbers. This question about input specification is an important matter, at least from the computational complexity viewpoint, because the way we define the input may drastically change the complexity class on which a problem lies. Moreover, it is not unlike that reductions between problems in graphical models need the use of recursive real numbers for them to be useful (Mauá, de Campos, and Zaffalon 2012; de Campos 2011). The following lemmas show that we can approximate an SQPN with recursive real numbers as precise as we want (up to a polynomial number of bits) using a SQPN with only rational numbers.

Lemma 6 *Given an SQPN $\mathcal{N}_R$ specified by rational and/or recursive real numbers, and a rational $\varepsilon \geq 2^{-poly(SZ)}$, we can construct, in polynomial time in SZ, an SQPN $\mathcal{N}$ with rational numbers such that $\forall P_R \in \mathcal{N}_R : \exists P \in \mathcal{N}$ where $|P_R(\mathbf{x}) - P(\mathbf{x})| \leq \varepsilon$, for every $\mathbf{x} \in \Omega_\mathbf{X}$ and $\mathbf{X} \subseteq \mathcal{X}$ (and vice-versa, that is, $\forall P \in \mathcal{N} : \exists P_R \in \mathcal{N}_R$).*

Proof Take $\mathcal{N}$ to be equal to $\mathcal{N}_R$ except that each recursive real number t used in the specification of $\mathcal{N}_R$ is replaced by the rational r with $|r - t| < 2^{-2n}\varepsilon$ (some special care has to be taken to keep distributions summing one, but that can be done without affecting the precision). As $\varepsilon \geq 2^{-poly(\mathrm{SZ})}$, we can use the DTM for t to obtain r in polynomial time in $poly(\mathrm{SZ}) + 2n$, and hence polynomial in SZ.

For now, assume that $\mathbf{x} \in \Omega_\mathbf{X}$ is a complete instantiation, that is, $\mathbf{X} = \mathcal{X}$. In this case, the desired result follows from the binomial expansion of the factorization of P_R and P (there will be $2^n - 1$ terms with (at least) one factor ε multiplied by probabilities that are less than or equal to 1):

$$\begin{aligned} P(\mathbf{x}) \quad &\leq \quad \prod_{i=1}^{n} (P_R(x_i|\pi_\mathcal{G}(X_i)) + 2^{-2n}\varepsilon) \leq 2^n 2^{-2n}\varepsilon \\ &\qquad + \prod_{i=1}^{n} P_R(x_i|\pi_\mathcal{G}(X_i)) = P_R(\mathbf{x}) + 2^{-n}\varepsilon, \end{aligned}$$

and similarly to obtain $P(\mathbf{x}) \geq P_R(\mathbf{x}) - 2^{-n}\varepsilon$. For the case where $\mathbf{X} \subset \mathcal{X}$, it is enough to use these inequalities and to sum over all the compatible complete instantiations:

$$P(\mathbf{x}) = \sum_{\mathbf{x}' \in \Omega_{\mathcal{X} \setminus \mathbf{X}}} P(\mathbf{x}', \mathbf{x}) \leq \sum_{\mathbf{x}' \in \Omega_{\mathcal{X} \setminus \mathbf{X}}} (P_R(\mathbf{x}', \mathbf{x}) + 2^{-n}\varepsilon),$$

which is less than $\varepsilon + P_R(\mathbf{x})$, and analogously for the lower bound. Hence we have $|P_R(\mathbf{x}) - P(\mathbf{x})| \leq \varepsilon$. □

[3]We use the standard asymptotic notation of O and Θ, and $poly(v)$ to mean $O(v^c)$, for any desired fixed c.

Lemma 7 *Let* $0 < \varepsilon < u \leq p_1 \leq p_2 \leq 1$ *be real numbers. Then*

$$\frac{p_1}{p_2} - \frac{2\varepsilon}{u} \leq \frac{p_1 - \varepsilon}{p_2 + \varepsilon} \quad \textit{and} \quad \frac{p_1 + \varepsilon}{p_2 - \varepsilon} \leq \frac{p_1}{p_2} + \frac{2\varepsilon}{u - \varepsilon}. \quad (3)$$

Proof We first manipulate the left-hand inequality:

$$\frac{p_1}{p_2} - \frac{2\varepsilon}{u} \leq \frac{p_1 - \varepsilon}{p_2 + \varepsilon} \iff (p_2 + \varepsilon)(p_1 - p_2\frac{2\varepsilon}{u}) \leq p_1 p_2 - p_2 \varepsilon$$

$$\iff (p_1 + p_2)\varepsilon \leq \frac{2\varepsilon}{u} p_2 (p_2 + \varepsilon) \iff u \leq \frac{2p_2(p_2+\varepsilon)}{p_1 + p_2},$$

which is true because $u \leq (p_2 + \varepsilon)$ and $\frac{2p_2}{p_1+p_2} \geq 1$. A similar manipulation proves the result for the right-hand of Expression (3), which we omit for the sake of brevity. $\square$

Lemma 8 *Given* $(\mathcal{N}_R, Q, \mathbf{e})$ *where* $\mathcal{N}_R$ *is an SQPN specified by rational and/or recursive real numbers such that each number is either zero or greater than* $2^{-poly(SZ)}$,[4] *and a rational* $2^{-poly(SZ)} \leq \varepsilon' \leq 1$, *we can construct, in polynomial time in SZ, an SQPN* $\mathcal{N}$ *with rational numbers such* $\forall P_R \in \mathcal{N}_R : \exists P \in \mathcal{N}$ *where*

$$|(P_R(q^1|\mathbf{e}) - P_R(q^1)) - (P(q^1|\mathbf{e}) - P(q^1))| \leq \varepsilon', \quad (4)$$

for any given $Q \in \mathcal{X}$ *and* $\mathbf{e} \in \Omega_{\mathbf{E}}$ *with non-zero probability (and vice-versa, that is,* $\forall P \in \mathcal{N} : \exists P_R \in \mathcal{N}_R$*).*

Proof First, note that $P_R(\mathbf{e}) = \sum_{\mathbf{x}' \in \Omega_{\mathcal{X} \setminus \mathbf{E}}} P_R(\mathbf{x}', \mathbf{e}) \geq 2^{-v}$, with $v = n \cdot poly(SZ)$, because of the assumption that $P_R(\mathbf{e}) > 0$ and the fact that $P_R(\mathbf{x})$, for any complete $\mathbf{x} \in \Omega_{\mathcal{X}}$, is either zero or is greater than $(2^{-poly(SZ)})^n = 2^{-v}$ (it is a product of n numbers).

Build $\mathcal{N}$ from $\mathcal{N}_R$ using Lemma 6 with an $\varepsilon = 2^{-v-2}\varepsilon'$. For any situation where $P_R(q^1, \mathbf{e}) = 0$, Equation (4) follows directly from Lemma 6. If $P_R(q^1, \mathbf{e}) > 0$, then $P_R(q^1, \mathbf{e}) \geq 2^{-v}$ (by the same reasoning as before), and we have the following (the first line uses Lemma 6, and the second uses Lemma 7): $P(q^1|\mathbf{e}) - P(q^1) \leq$

$$\begin{aligned}
&\leq \frac{P_R(q^1, \mathbf{e}) + 2^{-v-2}\varepsilon'}{P_R(\mathbf{e}) - 2^{-v-2}\varepsilon'} - (P_R(q^1) - 2^{-v-2}\varepsilon') \\
&\leq \frac{P_R(q^1, \mathbf{e})}{P_R(\mathbf{e})} + \frac{2 \cdot 2^{-v-2}\varepsilon'}{2^{-v} - 2^{-v-2}\varepsilon'} - P_R(q^1) + 2^{-v-2}\varepsilon' \\
&\leq \frac{P_R(q^1, \mathbf{e})}{P_R(\mathbf{e})} + \frac{2\varepsilon'}{4 - \varepsilon'} - P_R(q^1) + 2^{-v-2}\varepsilon' \\
&\leq P_R(q^1|\mathbf{e}) - P_R(q^1) + \varepsilon',
\end{aligned}$$

as $\frac{2\varepsilon'}{4-\varepsilon'} + 2^{-v-2}\varepsilon' \leq \varepsilon'$, for any $\varepsilon' \leq 1$. The lower bound is obtained in a similar way by applying Lemmas 6 and 7: $P(q^1|\mathbf{e}) - P(q^1) \geq$

$$\begin{aligned}
&\geq \frac{P_R(q^1, \mathbf{e}) - 2^{-v-2}\varepsilon'}{P_R(\mathbf{e}) + 2^{-v-2}\varepsilon'} - (P_R(q^1) + 2^{-v-2}\varepsilon') \\
&\geq \frac{P_R(q^1, \mathbf{e})}{P_R(\mathbf{e})} - \frac{2 \cdot 2^{-v-2}\varepsilon'}{2^{-v}} - P_R(q^1) - 2^{-v-2}\varepsilon',
\end{aligned}$$

which is greater than or equal to $P_R(q^1|\mathbf{e}) - P_R(q^1) - \varepsilon'$. $\square$

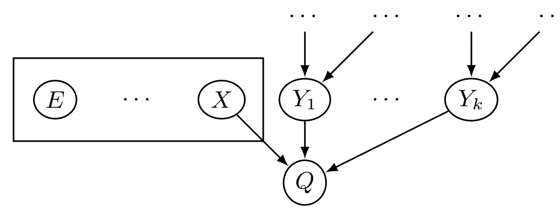

Figure 1: Piece of the network graph used in the proof of Theorem 9.

The results of the previous lemmas will be used later. For now, they tell us that we can compute with networks, such as Bayesian networks, specified by rationals and obtain results as close as we want to the results that would be obtained with networks specified by recursive real numbers, as long as none of them is extremely close to zero.

Single Observation

We devise an efficient algorithm for polytree-shaped networks with observation in a single node. We present the algorithm in the form of the following theorem.

Theorem 9 *Given* $(\mathcal{N}, Q, \mathbf{e})$ *such that* $\mathcal{N}$ *is a polytree SQPN specified with rational numbers, and* $\mathbf{e} \in \Omega_{\mathbf{E}}$ *is such that* $|\mathbf{E}| = 1$, `SQPN-NEGINF` *and* `SQPN-POSINF` *are solvable in linear time in* n *and quadratic in* 2^d *(recall that SZ* $= \Theta(n2^d)$*).*

Proof Let $\mathcal{Q}_{Rel}(q, e)$ denote the statement "$\exists P : P(q|e) - P(q) \; Rel \; 0$", where Rel is replaced by either $>$ or $<$ ($E = e$ is the observation). We will check whether $\mathcal{Q}_{>}(q^1, e)$ is true, which is the complement of `SQPN-NEGINF`, and whether $\mathcal{Q}_{<}(q^1, e)$ is true, which is the complement of `SQPN-POSINF`. As we devise a polynomial-time algorithm, solving `SQPN-NEGINF`/`SQPN-POSINF` or their complement is the same. The first assertion to note is that $\mathcal{Q}_{>}(q, e) \iff \mathcal{Q}_{<}(\neg q, e)$, achievable by simple manipulations (use $P(q|e) = 1 - P(\neg q|e)$ and $P(q) = 1 - P(\neg q)$). Because of that, an algorithm for $\mathcal{Q}_{>}(q, e)$ (for any given q and e) is also an algorithm for $\mathcal{Q}_{<}(q, e)$. The second important fact is that $\mathcal{Q}_{>}(q, e) \iff \mathcal{Q}_{>}(e, q)$, which comes from $P(q|e) - P(q) > 0 \iff P(q, e) - P(e)\,P(q) > 0 \iff P(e|q) - P(e) > 0$.

The algorithm to compute $\mathcal{Q}_{>}(q, e)$ is as follows:

1. If Q is an ancestor[5] of E, then return $\mathcal{Q}_{>}(e, q)$.
2. If E is a parent of Q, then let $\mathbf{y} = (y_1, \ldots, y_k) \in \Omega_{\mathbf{Y}}$, where $\mathbf{Y}$ are the other parents of Q, and $P(\mathbf{y}) = \prod_{i=1}^{k} P(y_i)$. In this case, return true if and only if

$$\exists P : \sum_{\mathbf{y}} (P(q|e, \mathbf{y}) - P(q|\neg e, \mathbf{y}))\, P(\mathbf{y}) > 0. \quad (5)$$

Note that to check whether this last assertion is true, we can obtain upper and lower bounds for each $P(y_i)$ separately, for instance using the 2U algorithm (Fagiuoli and

[4] Note that this is trivially true for the explicitly given numbers, but it is not necessarily the case for the recursive real ones.

[5] A is ancestor of B if there is a directed path from A to B.

Zaffalon 1998). After that, we solve Equation (5) (an exhaustive approach suffices). Note that if Q is associated to qualitative assessments, then this is an even simpler task.

3. Otherwise, let X be the parent of Q in the only path between E and Q. Return

$$\big(\mathcal{Q}_{>}(q, x^1) \wedge \mathcal{Q}_{>}(x^1, e)\big) \vee \big(\mathcal{Q}_{>}(\neg q, x^1) \wedge \mathcal{Q}_{>}(x^0, e)\big) . \tag{6}$$

We prove the correctness of this algorithm by induction. Let $\mathbf{W}$ be the sequence of nodes in the unique path from Q to E (including them). Note that every node that is not an ancestor of a node in $\mathbf{W}$ is a barren node for this query and can be simply discarded. First, suppose that Q is an ancestor of E. In this case, Step 1 calls the algorithm with q and e interchanged, so we can assume that Q is never an ancestor of E. This step is correct as already explained.

We show the algorithm's computation by induction on $\mathbf{W}$. Take Figure 1. First, suppose that $\mathbf{W} = \{Q, E\}$, that is, E is a parent of Q (consider it to be $E = X$ in the figure). This is processed by Step 2 of the algorithm, where Equation (5) is solved by taking the upper and lower bounds for each $P(y_i)$ separately and exhaustively trying their combinations. It is known that the maximum of Equation (5) is achieved by one of such combinations (Fagiuoli and Zaffalon 1998). Because of the independences of the network, Equation (5) is equivalent to $\exists P : P(q|e) - P(q|\neg e) > 0 \iff \exists P : P(q|e)\,(1 - P(e)) - P(q, \neg e) > 0 \iff \exists P : P(q|e) - P(q) > 0$, which is exactly $\mathcal{Q}_{>}(q, e)$. This concludes the basis of the induction.

Now assume $|\mathbf{W}| \geq 3$. Take Figure 1, where X is the only parent of $Q \in \mathbf{W}$. By induction hypothesis, suppose that $\mathcal{Q}_{>}(x^1, e)$ and $\mathcal{Q}_{>}(x^0, e)$ are already computed and their results are available to us. Note that $P(q|e) - P(q) =$

$$\begin{aligned}
&= \sum_x P(q|x)\,(P(x|e) - P(x)) \\
&= P\big(q|x^1\big)\,\big(P\big(x^1|e\big) - P\big(x^1\big)\big) + \\
&\quad\; P\big(q|x^0\big)\,\big(P\big(x^0|e\big) - P\big(x^0\big)\big) \\
&= \big(P\big(q|x^1\big) - P\big(q|x^0\big)\big)\,\big(P\big(x^1|e\big) - P\big(x^1\big)\big) .
\end{aligned}$$

Because we know the possible signs of the second factor of this equation (available from $\mathcal{Q}_{>}(x^1, e)$ and $\mathcal{Q}_{>}(x^0, e)$), we only need to find the possible signs of $P\big(q|x^1\big) - P\big(q|x^0\big)$. This latter question is the same as solving the queries $\mathcal{Q}_{>}(q, x^1)$ and $\mathcal{Q}_{>}(\neg q, x^1)$, because $P\big(q|x^1\big) - P\big(q|x^0\big) > 0 \iff P\big(q|x^1\big) - P(q) > 0 \iff \mathcal{Q}_{>}(q, x^1)$, while we have $P\big(q|x^1\big) - P\big(q|x^0\big) < 0 \iff \mathcal{Q}_{>}(\neg q, x^1)$. Moreover, these queries and the queries $\mathcal{Q}_{>}(x^1, e)$ and $\mathcal{Q}_{>}(x^0, e)$ do depend only on disjoint local conditional mass functions, that is, $\mathcal{Q}_{>}(x^1, e)$ and $\mathcal{Q}_{>}(x^0, e)$ are computed using nodes X and "above" (those from X towards E), while $\mathcal{Q}_{>}(q, x^1)$ and $\mathcal{Q}_{>}(\neg q, x^1)$ are computed using node Q, plus $Y_1, \ldots, Y_k$ and their ancestors, so their computations can be done separately without affecting each other. Hence, we obtain Equation (6). The correctness of the whole algorithm follows from the induction.

The complexity of the procedure is as follows. Equation (6) has two terms related to the recursive calls (second and fourth) and two terms (first and third) that are going to be solved by Equation (5) (because we know that the arguments of $\mathcal{Q}$ are directly connected in this case). By caching the results of the queries along the path $\mathbf{W}$, there will be $O(\mathbf{W})$ calls to solve Equation (5), each spending time $O(n_j 2^{2d})$, where n_j is the amount of nodes involved in each computation j along $\mathbf{W}$ (the algorithm 2U that is used as auxiliary to solve Equation (5) takes time at most $O(n_j 2^{2d})$ and the local computation takes time $O(2^{2d})$). The total time is $\sum_j O(n_j 2^{2d}) = O(n 2^{2d})$, because the sets of nodes involved in each of those computation of Equation (5) are disjoint. The algorithm's asymptotic time is linear in n (and also in SZ for any fixed d) and at most quadratic in 2^d (which is relevant only if d is not considered constant). $\square$

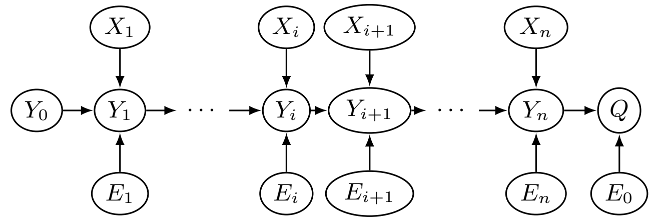

Figure 2: Network graph used in the proof of Theorem 11.

Multiple Observations

Before we show the hardness result for inferences in polytree-shaped SQPNs, we need the definition of the problem on which we base our polynomial-time reduction.

Definition 10 *Given a set of positive integer numbers $C = \{c_1, \ldots, c_n\}$, the* `PARTITION` *problem is the task to decide whether exists $I \subset \{1, \ldots, n\}$ such that $2 \cdot \sum_{i \in I} c_i = \sum_{i=1}^{n} c_i$. The input size SZ is assumed to be $\sum_{i=1}^{n} sz(c_i)$ (number of bits to encode all numbers).*

We can assume without loss that the problem is to find I such that $\sum_{i \in I} a_i = 1$, with $a_i = c_i/s$, and $2s = \sum_{i=1}^{n} c_i$. Note that $0 < a_i \leq 1$ (if there is any $a_i > 1$, then the instance would be trivially answered as NO) and $s \leq 2^{\mathrm{SZ}}$ (the sum of all c_i uses obviously at most SZ bits).

Theorem 11 *Given ($\mathcal{N}, Q, \mathbf{e}$) such that $\mathcal{N}$ is a polytree SQPN with only rational numbers,* `SQPN-NEGINF` *and* `SQPN-POSINF` *are coNP-Complete.*

Proof We prove the desired result by showing that their complements are NP-Complete. The complement of `SQPN-NEGINF`, namely `not-SQPN-NEGINF`, decides whether $\exists P \in \mathcal{N} : P\big(q^1|\mathbf{e}\big) - P\big(q^1\big) > 0$. Pertinence in NP is immediate, because there is a polynomial certificate, that is, given $P \in \mathcal{N}$, we can compute $P\big(q^1|\mathbf{e}\big) - P\big(q^1\big)$ in polynomial time using two queries of belief propagation (the SQPN becomes a Bayesian network) and check the sign of the difference. Hardness comes with a reduction from `PARTITION`.

Firstly, we build a SQPN $\mathcal{N}_R$ formed of rational and recursive real numbers and show that a given not-SQPN-NEGINF query solves PARTITION. Later we apply Lemma 8 to show that an SQPN formed only by rationals is enough, thus completing the proof. For each number a_i in the PARTITION problem, with $i = 1, \ldots, n$, we build three nodes: X_i, Y_i, E_i, such that X_i and E_i are root nodes, $P(e_i^1) = \varepsilon/(1+\varepsilon)$ ($\varepsilon > 0$ will be specified later), $P(X_i)$ is ambiguous, and Y_i has E_i, X_i and Y_{i-1} as parents (Y_0 is defined with $P(y_0^1) = 1$) and

$$\begin{array}{ll} P(y_i^1|y_{i-1}^1, e_i^1, x_i^1) = t_i, & P(y_i^1|y_{i-1}^1, e_i^1, x_i^0) = 1, \\ P(y_i^1|y_{i-1}^1, e_i^0, x_i^1) = (t_i)^2, & P(y_i^1|y_{i-1}^1, e_i^0, x_i^0) = 1, \\ P(y_i^1|y_{i-1}^0, e_i, x_i) = 0 & \text{for all } e_i, x_i, \end{array}$$

where $1/2 \leq t_i = 2^{-a_i} < 1$. Figure 2 depicts the network graph we have built for this reduction. Let $\mathbf{e} = (e_1^1, \ldots, e_n^1)$ and $I = \{i : X_i = x_i^1\}$. Now, we have that (using $\mathcal{I}$ as the indicator function)

$$\begin{aligned} P(y_n^1|\mathbf{e}) &= \prod_{i=1}^{n} \left(\mathcal{I}_{X_i = x_i^1} t_i + \mathcal{I}_{X_i = x_i^0}\right) = \prod_{i \in I} t_i, \\ P(y_n^1) &= \prod_{i=1}^{n} \left(\mathcal{I}_{X_i = x_i^1} \left(\frac{1}{1+\varepsilon}(t_i)^2 + \frac{\varepsilon}{1+\varepsilon} t_i\right) + \right. \\ &\qquad \left. \mathcal{I}_{X_i = x_i^0} \left(\frac{1}{1+\varepsilon} + \frac{\varepsilon}{1+\varepsilon}\right)\right) \\ &= \prod_{i \in I} \left(\frac{1}{1+\varepsilon}(t_i)^2 + \frac{\varepsilon}{1+\varepsilon} t_i\right), \end{aligned}$$

and using $0 < \frac{\varepsilon}{1+\varepsilon} < \varepsilon$ and $1 - \varepsilon < \frac{1}{1+\varepsilon} < 1$, we have

$$-\varepsilon \leq -\varepsilon \prod_{i \in I} (t_i)^2 < P(y_n^1) - \prod_{i \in I} (t_i)^2 < \varepsilon \prod_{i \in I} t_i \leq \varepsilon.$$

Let $\mathbf{t} = \prod_{i \in I} t_i = 2^{-\sum_{i \in I} a_i}$. Then we have the following guarantee: $|\left(P(y_n^1|\mathbf{e}) - P(y_n^1)\right) - \mathbf{t}(1-\mathbf{t})| < \varepsilon$. Note from Figure 2 that E_0 is a root node and Q has E_0 and Y_n as parents. Define $\alpha = \frac{1}{4} - 2\varepsilon > 0$, $P(e_0^0) = \alpha$ and $P(q^1|Y_n, E_0) = \alpha$ for $\{y_n^1, e_0^1\}$ and $\{y_n^0, e_0^0\}$, 2α for $\{y_n^1, e_0^0\}$ and zero for $\{y_n^0, e_0^1\}$. Now some simple manipulations give us $P(q^1|\mathbf{e}, e_0^1) - P(q^1) = \alpha\left(P(y_n^1|\mathbf{e}) - P(y_n^1) - \alpha\right)$, and thus

$$\left|\left(P(q^1|\mathbf{e}, e_0^1) - P(q^1)\right) - \alpha\left(\mathbf{t}(1-\mathbf{t}) - \alpha\right)\right| < \alpha\varepsilon. \quad (7)$$

The function $h(\mathbf{t}) = \alpha(\mathbf{t}(1-\mathbf{t}) - \alpha)$ is concave on $\mathbf{t}$ and has maximum h^* at $\alpha(\frac{1}{4} - \alpha) = 2\alpha\varepsilon$. This maximum happens when $\sum_{i \in I} a_i = 1$ (hence $\mathbf{t} = 2^{-1}$), and in this case the instance of PARTITION is an YES instance. Because our SQPN computes h with an error of at most $\alpha\varepsilon$ (by Equation (7)), an YES instance of PARTITION certainly achieves $P(q^1|\mathbf{e}, e_0^1) - P(q^1) > 0$, so it is an YES instance in the query not-SQPN-NEGINF with input $(\mathcal{N}_R, Q, \{\mathbf{e}, e_0^1\})$.

Now, we must prove that a NO instance of PARTITION corresponds to a NO instance here. Because there is a gap of at least $\frac{1}{s}$ among distinct values of $\sum_{i \in I} a_i$, every NO instance of PARTITION achieves at most $h' = \max\{h(2^{-(1+1/s)}), h(2^{-(1-1/s)})\} < h^* - 2^{-g}$, with $g = poly(\mathrm{SZ})$, because $s \leq 2^{\mathrm{SZ}}$. Although our query does not compute h' precisely, if we set $\varepsilon = 2^{-g}$, then we are sure that

$$h' < h^* - 2^{-g} = 2\alpha\varepsilon - 2^{-g} = \frac{\varepsilon}{2} - 4\varepsilon^2 - \varepsilon < -2\alpha\varepsilon,$$

and again by Equation (7), a NO instance of PARTITION is a NO of not-SQPN-NEGINF with input $(\mathcal{N}_R, Q, \{\mathbf{e}, e_0^1\})$.

Finally, we apply Lemma 8 using $\varepsilon' = \alpha\varepsilon/2$, which satisfies $2^{-poly(\mathrm{SZ})} \leq \varepsilon' < 1$ (and all numbers in $\mathcal{N}_R$ are either zero or greater than $2^{-poly(\mathrm{SZ})}$), and thus obtain an SQPN $\mathcal{N}$ only with rationals where the same decision for PARTITION holds, because from Equation (7) and Lemma 8 the difference between the result of the rational SQPN $\mathcal{N}$ and the function h will be strictly less than $\alpha\varepsilon + \varepsilon' < 2\alpha\varepsilon$, which suffices since $h^* = 2\alpha\varepsilon$ and $h' < -2\alpha\varepsilon$ (i.e. the sign of h is computed correctly by the query on $\mathcal{N}$).

The same can be applied to not-SQPN-POSINF by interchanging the states q^1 and q^0 in the construction of $\mathcal{N}_R$ (and thus of $\mathcal{N}$). In this case, the result holds for not-SQPN-POSINF because $P(q^0|\mathbf{e}) - P(q^0) > 0 \iff P(q^1|\mathbf{e}) - P(q^1) < 0$. $\square$

We have used a network with very simple topology and qualitative assessment in order to obtain the hardness results; the inclusion of other qualitative influences and synergies, situational signs and non-monotonic relations, can only make the problem harder, but the problem still belongs to coNP as long as we only consider polytrees, because the existence of a polynomial-time certificate is unaltered (given $P \in \mathcal{N}$ of a polytree, we can check whether P falsifies the influence in polynomial time by belief propagation). This implies that exact inferences in other specialized polytree semi-qualitative networks are coNP-Complete too.

Conclusion

We can summarize the contributions of this paper as follows. First, we have characterized the complexity of *exact* inference in polytree-shaped SQPNs to be coNP-Complete. Second, we demonstrated that we can approximate as well as we want the inferences in SQPNs with recursive real numbers (and hence inferences in Bayesian networks and other related probabilistic graphical models too) using networks with only rational numbers (this is valid for any network, not only polytrees). Finally, we have devised a very efficient linear-time algorithm for inferences in polytree-shaped SQPNs with observation in a single node, which is, to the best of our knowledge, the first exact polynomial-time algorithm for SQPNs. As future work, we intend to apply SQPNs to real problems and to develop other efficient but not necessarily exact algorithms for SQPNs.

Acknowledgements

The first author has been partially supported by the Hasler Foundation grant n. 10030 and by the Swiss National Supercomputing Centre with a *small development project*. The second author was partially supported by CNPq.

References

Bolt, J. H.; Renooij, S.; and van der Gaag, L. C. 2003. Upgrading ambiguous signs in QPNs. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 73–80.

Bolt, J. H.; van der Gaag, L. C.; and Renooij, S. 2003. Introducing situational influences in QPNs. In *Proc. of the European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, 113–124.

Bolt, J. H.; van der Gaag, L. C.; and Renooij, S. 2004. The practicability of situational signs for QPNs. In *Proc. of the International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU)*, 1691–1698.

de Campos, C. P., and Cozman, F. G. 2005. Belief updating and learning in semi-qualitative probabilistic networks. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 153–160.

de Campos, C. P. 2011. New complexity results for MAP in Bayesian networks. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2100–2106. AAAI Press.

Druzdzel, M. J., and Henrion, M. 1993a. Belief propagation in qualitative probabilistic networks. In Carrete, N. P., and Singh, M., eds., *Qualitative Reasoning and Decision Technologies*. CIMNE, Barcelona. 451–460.

Druzdzel, M. J., and Henrion, M. 1993b. Efficient reasoning in qualitative probabilistic networks. In *Proc. of the AAAI Conference on Artificial Intelligence (AAAI)*, 548–553.

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

Mauá, D.; de Campos, C.; and Zaffalon, M. 2012. The complexity of approximately solving influence diagrams. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 604–613. AUAI Press.

Parsons, S., and Dohnal, M. 1993. A semiqualitative approach to reasoning in probabilistic networks. *Applied Artificial Intelligence* 7:223–235.

Renooij, S., and van der Gaag, L. C. 2000. Exploiting non-monotonic influences in qualitative belief networks. In *Proc. International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU)*, 1285–1290.

Renooij, S., and van der Gaag, L. C. 2002. From qualitative to quantitative probabilistic networks. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 422–429. San Francisco, CA: Morgan Kaufmann Publishers.

Renooij, S., and Witteman, C. 1999. Talking probabilities: communicating probabilistic information with words and numbers. *International Journal of Approximate Reasoning* 22:169–194.

Renooij, S.; van der Gaag, L. C.; and Parsons, S. 2000. Propagation of multiple observations in qualitative probabilistic networks. In *Proc. of the Belgium-Netherlands Conference on Artificial Intelligence*, 235–242.

Renooij, S.; van der Gaag, L. C.; and Parsons, S. 2002a. Propagation of multiple observations in qpns revisited. In *Proc. of the European Conference on Artificial Intelligence (ECAI)*, 665–669. Amsterdam: IOS Press.

Renooij, S.; van der Gaag, L. C.; and Parsons, S. 2002b. Context-specific sign-propagation in qualitative probabilistic networks. *Artificial Intelligence* 140(1-2):207–230.

van der Gaag, L. C.; Bodlaender, H. L.; and Feelders, A. 2004. Monotonicity in Bayesian networks. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 569–576. AUAI Press.

Wellman, M. P. 1990. Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence* 44(3):257–303.