Complexity Results for Enhanced Qualitative Probabilistic Networks

Johan Kwisthout and Gerard Tel
Department of Information and Computer Sciences
University of Utrecht
Utrecht, The Netherlands

Abstract

While quantitative probabilistic networks (QPNs) allow the expert to state influences between nodes in the network as influence signs, rather than conditional probabilities, inference in these networks often leads to ambiguous results due to unresolved trade-offs in the network. Various enhancements have been proposed that incorporate a notion of strength of the influence, such as enhanced and rich enhanced operators. Although inference in standard (i.e., not enhanced) QPNs can be done in time polynomial to the length of the input, the computational complexity of inference in such enhanced networks has not been determined yet. In this paper, we introduce relaxation schemes to relate these enhancements to the more general case where continuous influence intervals are used. We show that inference in networks with continuous influence intervals is *NP*-hard, and remains *NP*-hard when the intervals are discretised and the interval $[-1, 1]$ is divided into blocks with length of $\frac{1}{4}$. We discuss membership of *NP*, and show how these general complexity results may be used to determine the complexity of specific enhancements to QPNs. Furthermore, this might give more insight in the particular properties of feasible and infeasible approaches to enhance QPNs.

1 Introduction

While probabilistic networks (Pearl, 1988) are based on a intuitive notion of causality and uncertainty of knowledge, elicitating the required probabilistic information from the experts can be a difficult task. Qualitative probabilistic networks (Wellman, 1990), or QPNs, have been proposed as a qualitative abstraction of probabilistic networks to overcome this problem. These QPNs summarise the conditional probabilities between the variables in the network by a sign, which denotes the direction of the effect. In contrast to quantitative networks, where inference has been shown to be *NP*-hard (Cooper, 1990), these networks have efficient (i.e., polynomial-time) inference algorithms. QPNs are often used as an intermediate step in the construction of a probabilistic network (Renooij and van der Gaag, 2002), as a tool for verifying properties of such networks (van der Gaag et al., 2006), or in applications where the exact probability distribution is unknown or irrelevant (Wellman, 1990).

Nevertheless, this qualitative abstraction leads to ambiguity when influences with contrasting signs are combined. *Enhanced* QPNs have been proposed (Renooij and van der Gaag, 1999) in order to allow for more flexibility in determining the influences (e.g., *weakly* or *strongly* positive) and partially resolve conflicts when combining influences. Also, mixed networks (Renooij and van der Gaag, 2002) have been proposed, to facilitate stepwise quantification and allowing both qualitative and quantitative influences to be modelled in the network.

Although inference in quantitative networks is *NP*-hard, and polynomial-time algorithms are known for inference in standard qualitative networks, the computational complexity of inference in enhanced networks has not been determined yet. In this paper we recall the definition of QPNs in section 2, and we introduce a framework to relate various enhancements, such as enhanced, rich enhanced, and interval-based operators in section 3. In section 4 we show that inference in the general, interval-based case is NP-hard. In section 5 we show that it remains NP-hard if we use discrete - rather than continuous - intervals. Furthermore, we argue that, although hardness proofs might be nontrivial to obtain, it is unlikely that there exist polynomial algorithms for less general variants of enhanced networks, such as the enhanced and rich enhanced operators suggested by Renooij and Van der Gaag (1999). Finally, we conclude our paper in section 6.

2 Qualitative Probabilistic Networks

In qualitative probabilistic networks, a directed acyclic graph $G = (V, A)$ is associated with a set Δ of qualitative influences and synergies (Wellman, 1990), where the influence of one node to another is summarised by a sign[1]. For example, a *positive* influence of a node A on its successor B, denoted with $S^{+}(A, B)$, expresses that higher values for A make higher values for B more likely than lower values, regardless of influences of other nodes on B. In binary cases, with $a > \bar{a}$ and $b > \bar{b}$, this can be summarised as $\Pr(b \mid ax) - \Pr(b \mid \bar{a}x) \geq 0$ for any value of x of other predecessors of B. *Negative* influences, denoted by S^{-}, and *zero* influences, denoted by S^{0}, are defined analogously. If an influence is not positive, negative, or zero, it is *ambiguous*, denoted by $S^{?}$. Influences can be direct (causal influence) or *induced* (inter-causal influence or *product synergy*). In the latter case, the value of one node influences the probabilities of values of another node, given a third node (Druzdzel and Henrion, 1993b).

Various properties hold for these qualitative influences, namely *symmetry*, *transitivity*, *composition*, *associativity* and *distribution* (Wellman, 1990; Renooij and van der Gaag, 1999). If we define $\hat{S}^{\delta}(A, B, t_i)$ as the influence S^{δ}, with $\delta \in \{+, -, 0, ?\}$, from a node A on a node B along trail t_i, we can formalise these properties as shown in table 1. The $\otimes$- and $\oplus$-operators that follow from the transitivity and composition properties are defined in table 2.

symmetry	$\hat{S}^{\delta}(A, B, t_i) \in \Delta \Leftrightarrow$
	$\hat{S}^{\delta}(B, A, t_i^{-1}) \in \Delta$
transitivity	$\hat{S}^{\delta}(A, B, t_i) \wedge \hat{S}^{\delta'}(B, C, t_j) \Rightarrow$
	$\hat{S}^{\delta\otimes\delta'}(A, C, t_i \circ t_j)$
composition	$\hat{S}^{\delta}(A, B, t_i) \wedge S^{\delta'}(A, B, t_j) \Rightarrow$
	$S^{\delta\oplus\delta'}(A, B, t_i \circ t_j)$
associativity	$S^{(\delta\oplus\delta')\oplus\delta''} = S^{\delta\oplus(\delta'\oplus\delta'')}$
distribution	$S^{(\delta\oplus\delta')\otimes\delta''} = S^{(\delta\otimes\delta'')\oplus(\delta'\otimes\delta'')}$

Table 1: Properties of qualitative influences

$\otimes$	+	−	0	?
+	+	−	0	?
−	−	+	0	?
0	0	0	0	0
?	?	?	0	?

$\oplus$	+	−	0	?
+	+	?	+	?
−	?	−	−	?
0	+	−	0	?
?	?	?	?	?

Table 2: The $\otimes$- and $\oplus$-operator for combining signs

Using these properties, an efficient (polynomial time) inference algorithm can be constructed (Druzdzel and Henrion, 1993a) that propagates observed node values to other neighbouring nodes. The basic idea of the algorithm, given in pseudo-code in figure 1, is as follows. When entering the procedure, a node I is instantiated with a '+' or a '−' (i.e., *trail* $= \emptyset$, *from* $=$ *to* $= I$ and *msign* $=$ *'+' or '−'*). Then, this node sign is propagated through the network, following active trails and updating nodes when needed. Observe from table 2 that a node can change at most two times: from '0' to '+', '−', or '?', and then only to '?'. This algorithm visits each node at most two times, and therefore halts after a polynomial amount of time.

[1]Note that the network $Q = (G, \Delta)$ represents infinitely many *quantitative* probabilistic networks that respect the restrictions on the conditional probabilities denoted by the signs of all arcs.

```
procedure PropagateSign(trail, from, to, msign):
sign[to] ← sign[to] ⊕ msign;
trail ← trail ∪ { to };
for each active neighbour V_i of to
do lsign ← sign of influence between to and V_i;
    msign ← sign[to] ⊗ lsign;
    if V_i ∉ trail and sign[V_i] ≠ sign[V_i] ⊕ msign
    then PropagateSign(trail, to, V_i, msign).
```

Figure 1: The sign-propagation algorithm

3 Enhanced QPNs

These qualitative influences and synergies can of course be extended to preserve a larger amount of information in the abstraction (Renooij and van der Gaag, 1999). For example, given a certain cut-off value α, an influence can be *strongly* positive ($\Pr(b \mid ax) - \Pr(b \mid \bar{a}x) \geq \alpha$) or *weakly* negative ($-\alpha \leq \Pr(b \mid ax) - \Pr(b \mid \bar{a}x) \leq 0$). The basic '+' and '−' signs are enhanced with signs for strong influences ('++' and '−−') and augmented with multiplication indices to handle complex dependencies on α as a result of transitive and compositional combinations. In addition, signs such as '$+_?$' and '$-_?$' are used to denote positive or negative influences of unknown strength. Using this notion of strength, tradeoffs in the network can be modelled by compositions of weak and strong opposite signs.

Furthermore, an *interval network* can be constructed (Renooij and van der Gaag, 2002), where each arc has an associated influence interval rather than a sign. Such an influence is denoted as $F^{[p,q]}(A, B)$, meaning that $\Pr(b \mid ax) - \Pr(b \mid \bar{a}x) \in [p, q]$. Note that, given this definition, $S^+(A, B) \iff F^{[0,1]}(A, B)$, and similar observations hold for S^-, S^0 and $S^?$. We will denote the intervals $[-1, 0]$, $[0, 1]$, $[0, 0]$ and $[-1, 1]$ as *unit intervals*, being special cases that correspond to the traditional qualitative networks. The ⊗- and ⊕-operator, denoting transitivity and composition in interval networks are defined in table 3. Note that it is possible that a result of a combination of two trails leads to an empty set, for example when combining $[\frac{1}{2}, 1]$ with $[\frac{3}{4}, 1]$, which would denote that the total influence of a node on another node, along multiple trails, would be greater than one, which is impossible. Since the individual intervals might be estimated by experts, this situation is not unthinkable, especially in large networks. This property can be used to detect design errors in the network.

Note, that the symmetry, associativity, and distribution property of qualitative networks do no longer apply in these enhancements. For example, although a positive influence from a node A to B along the direction of the arc also has a positive influence in the opposite direction, the strength of this influence is unknown. Also, the outcome of the combination of a strongly positive, weakly positive and weakly negative sign depends on the evaluation order.

$\otimes_i$	$[r, s]$
$[p, q]$	$[\min X, \max X]$, where $X = \{p \cdot r, p \cdot s, q \cdot r, q \cdot s\}$

$\oplus_i$	$[r, s]$
$[p, q]$	$[p + r, q + s] \cap [-1, 1]$

Table 3: The $\otimes_i$- and $\oplus_i$-operators for interval multiplication and addition

3.1 Relaxation schemes

If we take a closer look at the $\oplus_e$, $\oplus_r$, and $\otimes_e$ operators defined in (Renooij and van der Gaag, 1999) and compare them with the interval operators $\oplus_i$ and $\otimes_i$, we can see that the interval results are sometimes somehow 'relaxed'. We see that symbols representing influences correspond to intervals, but after the application of any operation on these intervals, the result is extended to an interval that can be represented by one of the available symbols. For example, in the interval model we have $[\alpha, 1] \oplus_i [-1, 1] = [\alpha - 1, 1]$, but, while $[\alpha, 1]$ corresponds to ++ in the enhanced model and $[-1, 1]$ corresponds to ?, $+ + \oplus_e ? = ? \equiv [-1, 1]$. The lower limit $\alpha - 1$ is relaxed to -1, because the actually resulting interval $[\alpha - 1, 1]$ does not correspond to any symbol. Therefore, to connect the (enhanced) qualitative and interval models, we will introduce *relaxation schemes* that map the result of each operation to the minimal interval that can be represented by one of the available symbols:

Definition 1. (Relaxation scheme)
R_x will be defined as a relaxation scheme, denoted as $R_x([a,b]) = [c,d]$, if R_x maps the outcome $[a,b]$ of an $\oplus$ or $\otimes$ operation to an interval $[c,d]$, where $[a,b] \subseteq [c,d]$.

In standard QPNs, the relaxation scheme (which I will denote R_I or the *unit scheme*) is defined as in figure 2.

$$R_I(a,b) = \begin{cases} [0,1] & \text{if } a \geq 0 \wedge b > 0 \\ [-1,0] & \text{if } a < 0 \wedge b \leq 0 \\ [0,0] & \text{if } a = b = 0 \\ [-1,1] & \text{otherwise} \end{cases}$$

Figure 2: Relaxation scheme $R_I(a,b)$

Similarly, the $\oplus_e$, $\oplus_r$, and $\otimes_e$ operators can be denoted with the relaxation schemes in figure 3, in which m equals $\min(i,j)$ and α is an arbitrary cut-off value. To improve readability, in the remainder of this paper the $\oplus$- and $\otimes$-operators, when used without index, denote operators on intervals as defined in table 3.

$$R_{\otimes_e}(a,b) = \begin{cases} [-1,1] & \text{if } a < 0 \wedge b > 0 \\ [a,b] & \text{otherwise} \end{cases}$$

$$R_{\oplus_e}(a,b) = \begin{cases} [\alpha^m,1] & \text{if } a = \alpha^i + \alpha^j \leq b \\ [-1,-\alpha^m] & \text{if } b = -(\alpha^i + \alpha^j) \geq a \\ [0,1] & \text{if } a \leq b = \alpha^i + \alpha^j \\ [-1,0] & \text{if } a = -(\alpha^i + \alpha^j) \leq b \\ [0,1] & \text{if } a = (\alpha^i - \alpha^j) \\ & \text{and } b \geq 0 \text{ and } i < j \\ [-1,0] & \text{if } a = -(\alpha^i - \alpha^j) \\ & \text{and } b \leq 0 \text{ and } i < j \\ [-1,1] & \text{if } a \leq 0 \text{ and } b \geq 0 \\ [a,b] & \text{otherwise} \end{cases}$$

$$R_{\oplus_r}(a,b) = \begin{cases} [-1,1] & \text{if } a < 0 \wedge b > 0 \\ [a,b] & \text{otherwise} \end{cases}$$

Figure 3: Relaxation schemes $R_{\otimes_e}$, $R_{\oplus_e}$, and $R_{\oplus_r}$

This notion of a relaxation scheme allows us to relate various operators in a uniform way. A common property of most of these schemes is that the $\oplus$-operator is no longer associative. The result of inference now depends on the order in which various influences are propagated through the network.

3.2 Problem definition

To decide on the complexity of inference of this general, interval-based enhancements of QPNs, a decision problem needs to be determined. We state this problem, denoted as iPieqnetd[2], as follows.

iPieqnetd

Instance: Qualitative Probabilistic Network $Q = (G, \Delta)$ with an instantiation for $A \in V(G)$ and a node $B \in V \setminus \{A\}$.
Question: Is there an ordering on the combination of influences such that the influence of A on $B \subset [-1,1]$?

3.3 Probability representation

In this paper, we assume that the probabilities in the network are represented by fractions, denoted by integer pairs, rather than by reals. This has the advantage, that the length of the result of addition and multiplication of fractions is polynomial in the length of the original numbers. We can efficiently code the fractions in the network by rewriting them, using their least common denominator. Adding or multiplying these fractions will not affect their denominators, whose length will not change during the inference process.

4 Complexity of the problems

We will prove the hardness of the inference problem iPieqnetd by a transformation from 3sat. We construct a network Q using clauses C and boolean variables U, and prove that, upon instantiation of a node I to $[1,1]$, there is an ordering on the combination of influences such that the influence of I on a given node $Y \in Q \setminus \{I\}$ is a true subset of $[-1,1]$, if and only if the corresponding 3sat instance is satisfiable. In the network, the influence of a node

[2]An acronym for *Interval-based Probabilistic Inference in Enhanced Qualitative Networks*

A on a node B along the arc (A, B) is given as an interval; when the interval equals $[1, 1]$ (i.e., $\Pr(b \mid a) = 1$ and $\Pr(b \mid \bar{a}) = 0$) then the interval is omitted for readability. Note, that the influence of B on A, against the direction of the arc (A, B), equals the unit interval of the influence associated with (A, B).

As a running example, we will construct a network for the following 3SAT instance, introduced in (Cooper, 1990):

Example 1. (3SAT_{ex})
$U = \{u_1, u_2, u_3, u_4\}$, and $C = \{(u_1 \vee u_2 \vee u_3), (\neg u_1 \vee \neg u_2 \vee u_3), (u_2 \vee \neg u_3 \vee u_4)\}$

This instance is satisfiable, for example with the truth assignment $u_1 = T$, $u_2 = F$, $u_3 = F$, and $u_4 = T$.

4.1 Construct for our proofs

For each variable in the 3SAT instance, our network contains a "variable gadget" as shown in figure 4. After the instantiation of node I with $[1, 1]$, the influence at node D equals $[\frac{1}{2}, 1] \oplus [-\frac{1}{2}, \frac{1}{2}] \oplus [-1, -\frac{1}{2}]$, which is either $[-1, \frac{1}{2}]$, $[-\frac{1}{2}, 1]$ or $[-1, 1]$, depending on the order of evaluation. We will use the non-associativity of the $\oplus$-operator in this network as a non-deterministic choice of assignment of truth values to variables. As we will see later, an evaluation order that leads to $[-1, 1]$ can be safely dismissed (it will act as a 'falsum' in the clauses, making both x and $\neg x$ false), so we will concentrate on $[-\frac{1}{2}, 1]$ (which will be our T assignment) and $[-1, \frac{1}{2}]$ (F assignment) as the two possible choices.

We construct literals u_i from our 3SAT instance, each with a variable gadget Vg as input. Therefore, each variable can have a value of $[-1, \frac{1}{2}]$ or $[-\frac{1}{2}, 1]$ as influence, non-deterministicly. Furthermore, we add clause-networks Cl_j for each clause in the instance and connect each variable u_i to a clause-network Cl_j if the variable occurs in C_j. The influence associated with this arc (u_i, Cl_j) is defined as $F^{[p,q]}(u_i, Cl_j)$, where $[p, q]$ equals $[-1, 0]$ if $\neg u_i$ is in C_j, and $[0, 1]$ if u_i is in C_j (figure 5). Note

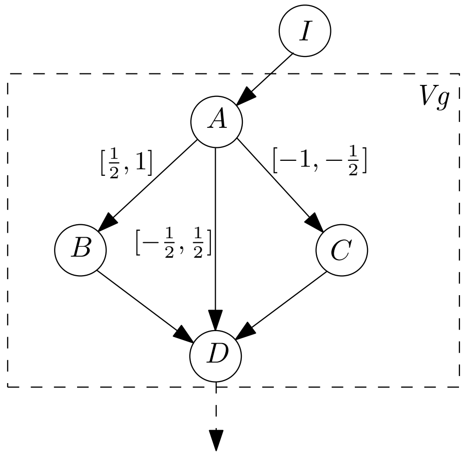

Figure 4: "Variable gadget" Vg

that an $\otimes$-operation with $[-1, 0]$ will transform a value of $[-1, \frac{1}{2}]$ in $[-\frac{1}{2}, 1]$ and vice versa, and $[0, 1]$ will not change them. We can therefore regard an influence $F^{[-1,0]}$ as a negation of the truth assignment for that influence. Note, that $[-1, 1]$ will stay the same in both cases.

If we zoom in on the clause-network in figure 6, we will see that the three 'incoming' variables in a clause, that have a value of either $[-1, \frac{1}{2}]$, $[-\frac{1}{2}, 1]$, or $[-1, 1]$, are multiplied with the arc influence $F_{i,j}$ to form variables, and then combined with the instantiation node (with a value of $[1, 1]$), forming nodes w_i. Note that $[-1, \frac{1}{2}] \oplus [1, 1] = [-1, 1] \oplus [1, 1] = [0, 1]$ and that $[-\frac{1}{2}, 1] \oplus [1, 1] = [\frac{1}{2}, 1]$. Since a $[-1, 1]$ result in the variable gadget does not change by multiplying with the $F_{i,j}$ influence, and leads to the same value as an F variable, such an assignment will never satisfy the 3SAT instance.

The influences associated with these nodes w_i are multiplied by $[\frac{1}{2}, 1]$ and added together in the clause result O_j. At this point, O_j has a value of $[\frac{k}{4}, 1]$, where k equals the number of literals which are true in this clause. The consecutive adding to $[-\frac{1}{4}, 1]$, multiplication with $[0, 1]$ and adding to $[1, 1]$ has the function of a *logical or*-operator, giving a value for C_j of $[\frac{3}{4}, 1]$ if no literal in the clause was true, and $[1, 1]$ if one or more were true.

We then combine the separate clauses C_j into a variable Y, by adding edges from each clause to Y using intermediate variables D_1 to D_{n-1}.

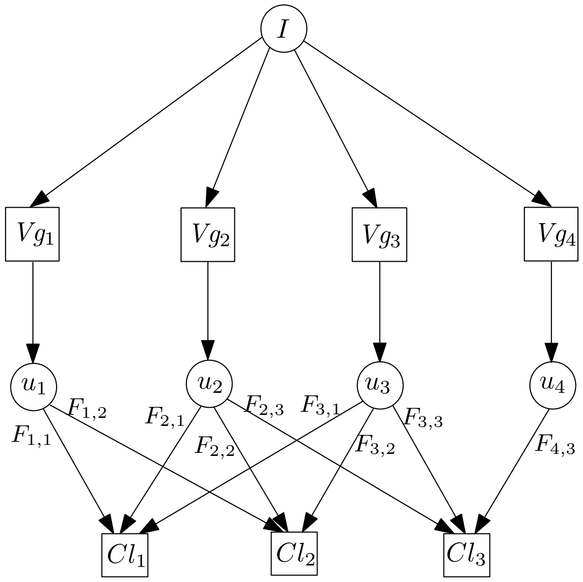

Figure 5: The literal-clause construction

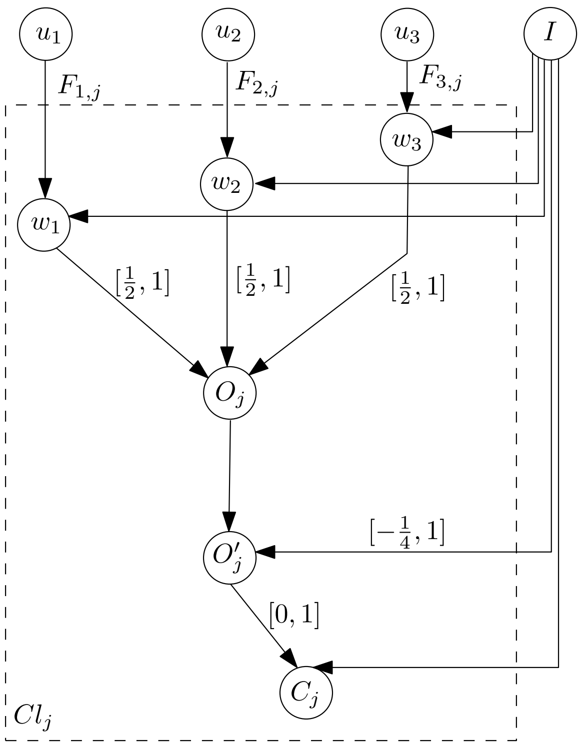

Figure 6: The clause construction

The use of these intermediate variables allows us to generalise these results to more restricted cases. The interval of these edges is $[\frac{1}{2}, 1]$, leading to a value of $[1, 1]$ in Y if and only if all clauses C_j have a value of $[1, 1]$ (see figure 7). The influence interval in Y has a value between $[\frac{3}{4}, 1]$ and $[\frac{2^{k+1}-1}{2^{k+1}}, 1]$, where k is the number of clauses, if one or more clauses had a value of $[0, 1]$. Note that we can easily transform the interval in Y to a subset of $[-1, 1]$ in Y' by consecutively adding it to $[-1, 1]$ and $[-1 + \frac{1}{2^{k+1}}, 1]$. This would result in a true subset of $[-1, 1]$ if and only Y was equal to $[1, 1]$.

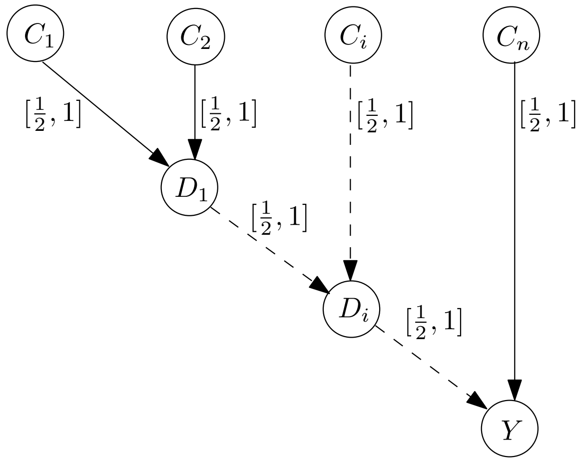

Figure 7: Connecting the clauses

4.2 *NP*-hardness proof

Using the construct presented in the previous section, the computational complexity of the iPieqnetd can be established as follows.

Theorem 1. *The* iPieqnetd *problem is NP-hard.*

Proof. To prove *NP*-hardness, we construct a transformation from the 3sat problem. Let (U, C) be an instance of this problem, and let $Q_{(U,C)}$ be the interval-based qualitative probabilistic network constructed from this instance, as described in the previous section. When the node $I \in Q$ is instantiated with $[1, 1]$, then I has an influence of $[1, 1]$ on Y (and therefore an influence on Y' which is a true subset of $[-1, 1]$) if and only if all nodes C_j have a value of $[1, 1]$, i.e. there exists an ordering of the operators in the 'variable-gadget' such that at least one literal in C is true. We conclude that (U, C) has a solution with at least one true literal in each clause, if and only if the iPieqnetd problem has a solution for network $Q_{(U,C)}$, instantiation $I = [1, 1]$ and node Y'. Since $Q_{(U,C)}$ can be computed from (U, C) in polynomial time, we have a polynomial-time transformation from 3SAT to iPIEQNETD, which proves *NP*-hardness of iPIEQNETD. □

4.3 On the possible membership of *NP*

Although iPIEQNETD has been shown to be *NP*-hard, membership of the *NP*-class (and, as a consequence, *NP*-completeness) is not trivial to prove. To prove membership of *NP*, one has to prove that if the instance is solvable, then there exists a certificate, polynomial in length with respect to the input of the problem, that can be used to verify this claim. A trivial certificate could be a formula, using the $\oplus$- and $\otimes$-operators, influences, and parentheses, describing how the influence of the a certain node can be calculated from the instantiated node and the characteristics of the network. Unfortunately, such a certificate can grow exponentially large.

In special cases, this certificate might also be described by an ordering of the nodes in the network and for each node an ordering of the inputs, which would be a polynomial certificate. For example, the variable gadget from figure 4 can be described with the ordering I, A, B, C, D plus an ordering on the incoming trails in D. All possible outcomes of the propagation algorithm can be calculated using this description. Note, however, that there are other propagation sequences possible that cannot be described in this way. For example, the algorithm might first explore the trail $A \to B \to D \to C$, and after that the trail $A \to C \to D \to B$. Then, the influence in D is dependent on the information in C, but C is visited by D following the first trail, and D is visited by C if the second trail is explored. Nevertheless, this cannot lead to other outcomes than $[-1, \frac{1}{2}]$, $[-\frac{1}{2}, 1]$, or $[-1, 1]$.

However, it is doubtful that such a description exists for all possible networks. For example, even if we can make a topological sort of a certain network, starting with a certain instantiation node X and ending with another node Y, it is still not the case that a propagation sequence that follows only the direction of the arcs until all incoming trails of Y have been calculated always yields better results than a propagation sequence that doesn't have this property. This makes it plausible that there exist networks where an optimal solution (i.e., a propagation sequence that leads to the smallest possible subset of $[-1, 1]$ at the node we are interested in) cannot be described using such a polynomial certificate.

5 Operator variants

In order to be able to represent every possible 3SAT instance, a relaxation scheme must be able to generate a variable gadget, and retain enough information to discriminate between the cases where zero, one, two or three literals in each clause are true. Furthermore, the relaxation scheme must be able to represent the instantiations $[1, 1]$ (or $\top$) and $[-1, -1]$ (or $\perp$), and the uninstantiated case $[0, 0]$. With a relaxation scheme that effectively divides the interval $[-1, 1]$ in discrete blocks with size of a multitude of $\frac{1}{4}$, (such as $R_{\frac{1}{4}}(a, b) = [\frac{\lfloor 4a \rfloor}{4}, \frac{\lceil 4b \rceil}{4}]$) the proof construction is essentially the same as in the general case discussed in section 3. This relaxation scheme does not have any effect on the intervals we used in the variable gadget and the clause construction of $Q_{(U,C)}$, the network constructed in the *NP*-hardness proof of the general case used only intervals (a, b) for which $R_{\frac{1}{4}}(a, b) = (a, b)$. Furthermore, when connecting the clauses, the possible influences in Y are relaxed to $[0, 1]$, $[\frac{1}{4}, 1]$, $[\frac{1}{2}, 1]$, $[\frac{3}{4}, 1]$, and $[1, 1]$, so we can onstruct Y' by consecutively adding the interval in Y to $[-1, 1]$ and $[-\frac{3}{4}, 1]$. Thus, the problem - which we will denote as RELAXED-PIEQNETD - remains *NP*-hard for relaxation scheme $R_{\frac{1}{4}}$.

The non-associativity of the $\oplus_e$- and $\oplus_r$-operators defined in (Renooij and van der Gaag, 1999) suggest hardness of the inference problem as well. Although $\oplus_e$ and $\oplus_r$ are not associative, they cannot produce results that can be regarded as opposites. For example, the expression $(+ + \oplus_e + \oplus_e -)$ can lead to a positive influence of unknown strength ('$+_?$') when evaluated as $((+ + \oplus_e +) \oplus_e -)$ or an unknown influence

('?') when evaluated as $(+ + \oplus_e (+ \oplus_e -))$, but never to a negative influence. A transformation from a 3SAT variant might not succeed because of this reason. However, it might be possible to construct a transformation from RELAXED-PIEQNETD, which is subject of ongoing research.

6 Conclusion

In this paper, we addressed the computational complexity of inference in enhanced Qualitative Probabilistic Networks. As a first step, we have "embedded" both standard and enhanced QPNs in the interval-model using relaxation schemes, and we showed that inference in this general interval-model is *NP*-hard and remains *NP*-hard for relaxation scheme $R_{\frac{1}{4}}(a, b)$. We believe, that the hardness of inference is due to the fact that reasoning in QPNs is underdefined: The outcome of the inference process depends on choices during evaluation. Nevertheless, further research needs to be conducted in order to determine where exactly the *NP/P* border lies, in other words: which enhancements to the standard qualitative model allow for polynomial-time inference, and which enhancements lead to intractable results. Furthermore, a definition of transitive and compositional combinations of qualitative influences in which the outcome is independent of the order of the influence propagation might reduce the computational complexity of inference and facilitate the use of qualitative models to design, validate, analyse, and simulate probabilistic networks.

Acknowledgements

This research has been (partly) supported by the Netherlands Organisation for Scientific Research (NWO).

The authors wish to thank Hans Bodlaender and Silja Renooij for their insightful comments on this subject.

References

G.F. Cooper. 1990. The Computational Complexity of Probabilistic Inference using Bayesian Belief Networks. *Artificial Intelligence*, 42(2):393–405.

M.J. Druzdzel and M. Henrion. 1993a. Belief Propagation in Qualitative Probabilistic Networks. In N. Piera Carrete and M.G. Singh, editors, *Qualitative Reasoning and Decision Technologies*, CIMNE: Barcelona.

M.J. Druzdzel and M. Henrion. 1993b. Intercausal reasoning with uninstantiated ancestor nodes. In *Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence*, pages 317–325.

J. Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, Palo Alto.

S. Renooij and L.C. van der Gaag. 1999. Enhancing QPNs for trade-off resolution. In *Proceedings of the Fifteenth Conference on Uncertainy in Artificial Intelligence*, pages 559–566.

S. Renooij and L.C. van der Gaag. 2002. From Qualitative to Quantitative Probabilistic Networks. In *Proceedings of the Eighteenth Conference in Uncertainty in Artificial Intelligence*, pages 422–429.

L.C. van der Gaag, S. Renooij, and P.L. Geenen. 2006. Lattices for verifying monotonicity of Bayesian networks. In *Proceedings of the Third European Workshop on Probabilistic Graphical Models (to appear).*

M.P. Wellman. 1990. Fundamental Concepts of Qualitative Probabilistic Networks. *Artificial Intelligence*, 44(3):257–303.