Attaining monotonicity for Bayesian networks

Merel T. Rietbergen Linda C. van der Gaag

Department of Information and Computing Sciences, Utrecht University
P.O. Box 80.089, 3508 TB Utrecht, The Netherlands

Abstract

Many real-life Bayesian networks are expected to exhibit commonly known properties of monotonicity, in the sense that higher values for the observable variables should make higher values for the main variable of interest more likely. Yet, violations of these properties may inadvertently be introduced into a network despite careful engineering efforts. In this paper, we present a method for resolving such violations of monotonicity by varying a single parameter probability. Our method constructs intervals of numerical values to which a parameter can be varied to attain monotonicity without introducing new violations. We argue that our method has a high runtime, yet can be practically applied to at least parts of a network.

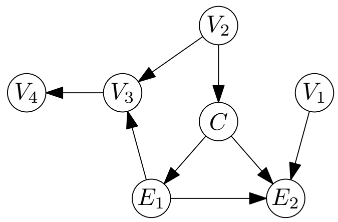
1 Introduction

Many fields of society are pervaded by problems of a scientific nature which require probabilistic reasoning over a collection of interrelated random variables for their solution. We consider for an example a physician who has to establish whether a patient has a specific disease. Upon reasoning about the uncertainties involved, the physician needs to take into consideration the diagnostic weight of the evidence presented by the patient. If the physician observes that the patient presents with worse symptoms and test results, then he will consider this patient more likely to have a worse disease than a patient presenting with less severe findings. For many real-life problems in fact, domain knowledge dictates that worse evidence makes a worse outcome more likely. This property is commonly known as monotonicity of the problem's outcome in its evidence.

When a Bayesian network is employed in a real-life domain of application, its users expect the network to exhibit commonly known properties of monotonicity; if the network violates any such property, it will not be easily accepted by its users, not even if it shows high performance otherwise. Yet, during the construction of a Bayesian network, violations of monotonicity may inadvertently be introduced despite careful engineering efforts. To attain monotonicity for the network, all such violations must be identified and resolved. Experience has shown, however, that violations of monotonicity can hardly be detected by hand. Recently, moreover, it was established that automatically verifying monotonicity is of a high computational complexity; the problem was shown to be co-NP^{PP}-complete in general [1]. In view of this unfavorable complexity result, Van der Gaag *et al.* proposed a method for verifying monotonicity which proved to be feasible for studying at least parts of a real-life network in veterinary medicine [2].

Various methods may be considered for attaining monotonicity for a Bayesian network which does not yet exhibit the required property in its evidence variables. In essence, these methods amount to changing the network's graphical structure, varying its parameter probabilities, or both. In this paper we present an initial investigation of the problem of resolving violations of monotonicity by varying the network's probabilities. We note that since the problem of verifying monotonicity in itself is already of a high computational complexity, we cannot expect to find an efficient, generally applicable method for resolving violations. We note moreover that the problem of attaining monotonicity is closely related to the problem of tuning parameters to meet some constraints on the network's output [3], which was shown to be NP^{PP}-complete in general [4]. Given these unfavorable complexity results, we initiated our investigation into the problem of attaining monotonicity for Bayesian networks by focusing on the variation of single parameter probabilities.

We present in this paper a method, called the *intersection-of-intervals method*, for attaining monotonicity for Bayesian networks by varying a single parameter probability. Our method constructs intervals of numerical values to which a parameter can be varied such that there are no more violations of monotonicity



$p(v_1) = 0.24$

$p(v_2) = 0.47$

$p(c|\bar{v}_2) = 0.37$
$p(c|v_2) = 0.86$

$p(e_1|\bar{c}) = 0.14$
$p(e_1|c) = 0.28$

$p(e_2|\bar{c}, \bar{e}_1, \bar{v}_1) = 0.99$
$p(e_2|\bar{c}, \bar{e}_1, v_1) = 0.27$
$p(e_2|\bar{c}, e_1, \bar{v}_1) = 0.46$
$p(e_2|\bar{c}, e_1, v_1) = 0.37$
$p(e_2|c, \bar{e}_1, \bar{v}_1) = 0.69$
$p(e_2|c, \bar{e}_1, v_1) = 0.95$
$p(e_2|c, e_1, \bar{v}_1) = 0.84$
$p(e_2|c, e_1, v_1) = 0.58$

$p(v_3|\bar{e}_1, \bar{v}_2) = 0.3$
$p(v_3|\bar{e}_1, v_2) = 0.06$
$p(v_3|e_1, \bar{v}_2) = 0.65$
$p(v_3|e_1, v_2) = 0.15$

$p(v_4|\bar{v}_3) = 0.4$
$p(v_4|v_3) = 0.81$

Figure 1: An example Bayesian network with two violations of monotonicity.

in the resulting modified network. Since resolving a single violation may cause new violations to arise, our method studies the effect of parameter variation for all possible combinations of evidence simultaneously rather than for just the combinations of evidence for which the property of monotonicity is violated. If the union of intervals constructed for a parameter probability is empty, then this parameter cannot, upon variation, resolve the identified violations without introducing new ones; otherwise, the parameter can be varied to any value from the union of intervals to attain monotonicity for the network at hand.

The paper is organized as follows. In Section 2, we introduce our notational conventions and review the concept of Bayesian network. In Section 3, we review the concept of monotonicity for Bayesian networks and study properties of its violations. In Section 4, we reduce the graphical structure of a network by eliminating variables for which parameter variation cannot serve to attain monotonicity. In Section 5, we detail our intersection-of-intervals method for resolving violations of monotonicity without introducing new ones; we also discuss the complexity of our method and how it can be practically applied. Finally, in Section 6, we outline our results and conclusions as well as some ideas for further study.

2 Bayesian Networks

A Bayesian network is a model of a joint probability distribution $\Pr$ over a set of random variables $\mathbf{V}$. Before briefly reviewing the concept of Bayesian network, we introduce our notational conventions. We use (indexed) upper-case letters V_i to denote individual variables from the set $\mathbf{V}$ and bold-faced upper-case letters $\mathbf{S}$ to denote (sub-)sets of variables. Each variable V_i has an associated domain of possible values, denoted $\Omega(V_i)$. While our results hold for all variables in general, for ease of presentation we address in this paper binary variables only. A variable V_i thus is taken to have the possible values v_i and $\bar{v}_i$, which are ordered $\bar{v}_i \leq v_i$; an assignment $V_i = v_i'$, for some $v_i' \in \Omega(V_i)$, will be referred to as an observation or as evidence for V_i, alternatively. The set of all joint value assignments to a set of variables $\mathbf{S}$ equals the Cartesian product of the domains of the variables involved, that is, $\Omega(\mathbf{S}) = \times_{V_i \in \mathbf{S}} \Omega(V_i)$. Elements from $\Omega(\mathbf{S})$ are denoted by bold-faced lower-case letters $\mathbf{s}$ and are ordered by the partial ordering $\preceq$ induced by the total orderings $\leq$ of the domains of the individual variables. As long as ambiguity cannot occur, we will use v_i' as a shorthand notation for $V_i = v_i'$; similarly we will write $\mathbf{s}$ to denote $\mathbf{S} = \mathbf{s}$.

A Bayesian network $B = (G, P)$ now includes a directed acyclic graph $G = (\mathbf{V}, A)$, in which each vertex models a random variable and where the set of arcs A captures the probabilistic (in)dependencies between the variables. We say that two variables are d-separated by the available evidence if every chain between the two variables contains either an observed variable with at least one emanating arc, or a variable with two incoming arcs such that neither the variable itself nor any of its descendants in the graph have been observed; two variables which are d-separated in the network's graph, are considered mutually independent given the evidence. The strengths of the relationships modeled between the variables are expressed by means of a set P of probability distributions. For each variable V_i, the set P includes the (conditional) probability distributions $p(V_i \mid \pi(V_i))$ over V_i given all value assignments to its parents $\pi(V_i)$ in the graph. The separate probabilities specified in P are called the parameters of the network; the probability distributions for a variable V_i with each other are called the conditional probability table of V_i. The network's graphical structure and associated parameter probabilities represent the unique joint probability distribution $\Pr(\mathbf{V}) = \prod_{V_i \in \mathbf{V}} p(V_i \mid \pi(V_i))$ over the variables $\mathbf{V}$.

Example 1. *Figure 1 depicts the small network that we will use for our running example throughout the paper; it includes seven variables for which twenty (conditional) probability distributions have been specified.*

3 Monotonicity in Bayesian Networks

In most real-life applications of Bayesian networks, the represented variables have different roles. In many applications in fact, a set of observable input variables $\mathbf{E}$ and a single main variable of interest C are distinguished. The concept of monotonicity in Bayesian networks has been introduced to describe properties of the (possibly indirect) relationships between these variables. The concept is defined as follows.

Definition 1. *A Bayesian network $B = (G, P)$ is* isotone in distribution *in its observable variables $\mathbf{E}$ if*

$$\mathbf{e} \preceq \mathbf{e}' \;\;\Rightarrow\;\; \Pr(C \leq c' \mid \mathbf{e}') \leq \Pr(C \leq c' \mid \mathbf{e})$$

for all $c' \in \Omega(C)$ and $\mathbf{e}, \mathbf{e}' \in \Omega(\mathbf{E})$.

Note that for a binary variable of interest C, the property of isotonicity in distribution implies that, for all $\mathbf{e}, \mathbf{e}' \in \Omega(\mathbf{E})$, if $\mathbf{e} \preceq \mathbf{e}'$, then $\Pr(\bar{c} \mid \mathbf{e}') \leq \Pr(\bar{c} \mid \mathbf{e})$ and, hence, $\Pr(c \mid \mathbf{e}) \leq \Pr(c \mid \mathbf{e}')$. The results presented in the sequel are readily extended to hold also for the reverse property of antitonicity, which states that $\mathbf{e} \preceq \mathbf{e}'$ implies $\Pr(C \leq c' \mid \mathbf{e}') \geq \Pr(C \leq c' \mid \mathbf{e})$ for all $c' \in \Omega(C)$ and $\mathbf{e}, \mathbf{e}' \in \Omega(\mathbf{E})$. Without loss of generality therefore, we will use the term monotonicity to refer to the property of isotonicity in distribution.

If a Bayesian network does not exhibit monotonicity in its observable variables, then there must be one or more pairs of joint value assignments $\mathbf{e}, \mathbf{e}' \in \Omega(\mathbf{E})$ with $\mathbf{e} \preceq \mathbf{e}'$ for which $\Pr(\bar{c} \mid \mathbf{e}') > \Pr(\bar{c} \mid \mathbf{e})$. In their work on identifying such violations of monotonicity [2], Van der Gaag *et al.* showed that it suffices to consider only pairs of assignments $\mathbf{e}, \mathbf{e}' \in \Omega(\mathbf{E})$ that differ in the value for just a single observable variable $E_i \in \mathbf{E}$. In the sequel, we will build upon this property and write $\mathbf{e}_i^-$ to denote the joint value assignment to $\mathbf{E} \setminus \{E_i\} = \mathbf{E}_i^-$ that is shared by such a pair $\mathbf{e}, \mathbf{e}'$. Studying monotonicity in a Bayesian network now amounts to establishing whether the property $\Pr\left(\bar{c} \mid e_i, \mathbf{e}_i^-\right) \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_i^-\right)$ holds for all $E_i \in \mathbf{E}$ and all value assignments $\mathbf{e}_i^-$ to $\mathbf{E}_i^-$; we will use $viol(\mathbf{e}_i^-)$ to denote a violation of the property for a specific variable V_i and evidence $\mathbf{e}_i^-$.

Example 2. *We consider again the example network B from Figure 1. In this network, C is the main variable of interest and E_1, E_2 are the observable variables; the other variables are intermediate and cannot be observed. To establish whether B is monotone in its observable variables, we compute from the network the conditional probabilities of $\bar{c}$ given all possible combinations of values for E_1 and E_2 (rounded to three decimal places):*

$$\Pr(\bar{c} \mid e_1, e_2) = 0.158 \qquad \Pr(\bar{c} \mid \bar{e}_1, e_2) = 0.463$$
$$\Pr(\bar{c} \mid e_1, \bar{e}_2) = 0.457 \qquad \Pr(\bar{c} \mid \bar{e}_1, \bar{e}_2) = 0.370$$

From the computed probabilities we observe that B does not meet the property of monotonicity. In fact, the computed probabilities reveal the violations $viol(\bar{e}_1)$ and $viol(\bar{e}_2)$.

If domain knowledge dictates that a Bayesian network should be monotone in its observable variables, all violations must be identified and resolved. Various methods may be considered for attaining monotonicity for a network, which are all based upon changing the graphical structure, varying the parameters or both. In this paper we address the problem of resolving violations of monotonicity by varying a single parameter probability. Let $p(u \mid \boldsymbol{\pi})$ be a parameter probability for some variable U, where $u \in \Omega(U)$ and $\boldsymbol{\pi} \in \Omega(\pi(U))$ is a joint value assignment to the parents of U in the network's graph. Varying the parameter $p(u \mid \boldsymbol{\pi})$ is said to resolve the violation $viol(\mathbf{e}_i^-)$, if there exists a numerical value $x \in [0, 1]$ for which

$$\Pr\left(\bar{c} \mid e_i, \mathbf{e}_i^-\right)\left(p(u \mid \boldsymbol{\pi}) = x\right) \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_i^-\right)\left(p(u \mid \boldsymbol{\pi}) = x\right),$$

where $\Pr(\mathbf{V})\big(p(u \mid \boldsymbol{\pi}) = x\big)$ indicates the probability distribution over the variables $\mathbf{V}$ as established from the network after changing the numerical value of the parameter $p(u \mid \boldsymbol{\pi})$ to x. A parameter which serves to simultaneously resolve all violations of monotonicity for a network without introducing any new ones, will be termed a *resolvent parameter*. We note that a Bayesian network which includes one or more violations of monotonicity may or may not have such a resolvent parameter.

4 Reducing the graphical structure

Identifying the parameters that upon variation can resolve all violations of monotonicity in a Bayesian network carries a considerable computational burden. To restrict the number of computations involved, we begin by preprocessing a network by eliminating variables for which we know that they cannot have resolvent parameters; that is, we eliminate variables for which it can be decided beforehand that their parameter probabilities cannot affect a probability of interest upon variation, for example because the probability of interest is shielded from their influence by the available evidence. We say that the probability of interest is algebraically independent of the parameters of these variables and use $\nsim$ to denote such independence.

By simple inspection of the graphical structure of a network, some variables with non-influential parameter probabilities can be feasibly identified without any reference to the parameters' numerical values. For this purpose, we exploit the concept of sensitivity set which was introduced before in sensitivity analysis of Bayesian networks [5]. The sensitivity set for a variable of interest C given observed variables $\mathbf{E}$ is the set of all variables for which the probability of interest is algebraically dependent of its parameter probabilities. This set is obtained as follows. From the graph G of a Bayesian network, we construct a new graph G^* by adding an auxiliary parent X_i to every vertex $V_i \in \mathbf{V}$; this parent X_i in essence represents the conditional probability table of V_i. The sensitivity set for C given $\mathbf{E}$, denoted $Sen(C, \mathbf{E})$, now is the set of all variables $V_i \in \mathbf{V}$ for which X_i and C are not d-separated by $\mathbf{E}$. If X_i and C are d-separated by $\mathbf{E}$, then the probability $\Pr(c \mid \mathbf{e})$ is algebraically independent of $p(V_i \mid \pi(V_i))$ for any $\mathbf{e} \in \Omega(\mathbf{E})$ [5].

Building upon the concept of sensitivity set, we have the following result for the problem of attaining monotonicity for a Bayesian network.

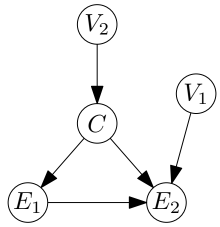
Proposition 1. *Let B be a Bayesian network with a main variable of interest C such that B is not monotone in its observable variables $\mathbf{E}$. Let $V_i \notin Sen(C, \mathbf{E})$ and let p be a parameter from the conditional probability table of V_i. Then, varying p cannot resolve any violation of monotonicity in B.*

Proof. Let $p = p(v_i' \mid \boldsymbol{\pi})$ be an arbitrary parameter probability of the variable V_i in B. Since $V_i \notin Sen(C, \mathbf{E})$, we have that its auxiliary parent X_i is d-separated from the variable of interest C by the observed variables $\mathbf{E}$ and that $\Pr(c' \mid \mathbf{e}) \nsim p(v_i' \mid \boldsymbol{\pi})$ for all $c' \in \Omega(C)$ and all $\mathbf{e} \in \Omega(\mathbf{E})$. The probabilities $\Pr(c \mid \mathbf{e})$ and $\Pr(\bar{c} \mid \mathbf{e})$ for the variable of interest C thus are constants in terms of the parameter p for all $\mathbf{e} \in \Omega(\mathbf{E})$. We conclude that varying p cannot resolve any violation of monotonicity in B. □

From Proposition 1 we have that varying a parameter from the conditional probability table of a variable which is not included in the sensitivity set under consideration cannot be used to attain monotonicity; we would like to note that this property holds for Bayesian networks in general and is not restricted to binary networks. For resolving violations of monotonicity by parameter variation, therefore, it suffices to consider the parameter probabilities of the variables from the sensitivity set only. Varying such a parameter probability may serve to attain monotonicity of the network at hand, yet is not guaranteed to do so. The parameter probabilities of the variables from the sensitivity set thus need further investigation.

Building upon the concept of sensitivity set, we now preprocess a Bayesian network under consideration by restricting it to the part which is relevant for studying monotonicity. For this purpose, we cannot simply remove all variables which are not included in the sensitivity set of the network's variable of interest, since some variables from the set $\mathbf{E} \setminus Sen(C, \mathbf{E})$ may still be needed to incorporate evidence into the computation of the probability distribution over the variable of interest. We therefore retain the variables from $Sen(C, \mathbf{E}) \cup \mathbf{E}$ and remove all other variables along with their incident arcs. If upon doing so the graphical structure of the network falls apart into multiple connected components, then the network is further restricted by removing all variables that are not included in the same component as the variable of interest C. In the remainder of the paper, we assume that a Bayesian network has been preprocessed as described above.

Example 3. *We consider again the example network from Figure 1. Figure 2 depicts the restricted network that results after preprocessing. Since the variables V_3, V_4 are not included in the sensitivity set for the variable of interest C given the observed variables E_1, E_2, we removed these variables and their incident arcs from the network's graphical structure. We observe that the thus restricted network still includes the violations of monotonicity from Example 2.*



$p(v_1) = 0.24$	$p(e_2 \vert \bar{c}, \bar{e}_1, \bar{v}_1) = 0.99$
	$p(e_2 \vert \bar{c}, \bar{e}_1, v_1) = 0.27$
$p(v_2) = 0.47$	$p(e_2 \vert \bar{c}, e_1, \bar{v}_1) = 0.46$
	$p(e_2 \vert \bar{c}, e_1, v_1) = 0.37$
$p(c \vert \bar{v}_2) = 0.37$	$p(e_2 \vert c, \bar{e}_1, \bar{v}_1) = 0.69$
$p(c \vert v_2) = 0.86$	$p(e_2 \vert c, \bar{e}_1, v_1) = 0.95$
$p(e_1 \vert \bar{c}) = 0.14$	$p(e_2 \vert c, e_1, \bar{v}_1) = 0.84$
$p(e_1 \vert c) = 0.28$	$p(e_2 \vert c, e_1, v_1) = 0.58$

Figure 2: The restricted Bayesian network constructed from the network from Figure 1.

5 The intersection-of-intervals method

In the previous section we showed that a Bayesian network can be restricted, based upon graphical considerations only, to the part that is relevant for studying violations of monotonicity. For each of the remaining variables we must now determine whether varying a parameter from its conditional probability table can result in monotonicity. For this purpose, we introduce a method which determines whether a specific parameter can be varied to a numerical value for which there are no violations of monotonicity in the resulting network. Using this method we can then decide whether there exists a parameter that can be varied to attain monotonicity. Our method has been designed to apply to arbitrary networks in general [6]. In this paper, however, we discuss a slightly simplified variant of the method applicable to binary networks only.

5.1 The method

We consider a (restricted) Bayesian network $B = (G, P)$ with a variable of interest C and a set of observable variables $\mathbf{E}$. Suppose that there is a single violation of monotonicity $viol(\mathbf{e}_i^-)$ in B. To resolve this violation by parameter variation, we must change the value of a parameter $p \in P$ to a numerical value x in the unit interval $[0, 1]$ such that $\Pr\left(\bar{c} \mid e_i, \mathbf{e}_i^-\right)\left(p = x\right) \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_i^-\right)\left(p = x\right)$. More generally, however, a network may contain multiple violations of monotonicity. Also, resolving one such violation may cause other, new violations to arise. To attain monotonicity for the network B, we must therefore vary a parameter $p \in P$ to a numerical value $x \in [0, 1]$ such that the system of inequalities

$$\Pr\left(\bar{c} \mid e_i, \mathbf{e}_i^-\right)\left(p = x\right) \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_i^-\right)\left(p = x\right)$$

holds for all $\mathbf{e}_i^- \in \Omega(\mathbf{E}_i^-)$ and all $E_i \in \Omega(\mathbf{E})$. This observation gives rise to the following method, called the intersection-of-intervals method. The method determines whether the value of a specific parameter $p \in P$ can be changed to a new value such that there are no violations of monotonicity in the resulting network; it thus determines whether p is a resolvent parameter for B. More specifically, the intersection-of-intervals method first determines for each $\mathbf{e}_i^-$ separately, the union of intervals of values x for the parameter p for which the above inequality holds; we call this union of intervals the solution space for $\mathbf{e}_i^-$. We can now only attain monotonicity by varying p to a value x, if x is included in the intersection of the solution spaces for all $\mathbf{e}_i^-$ and every $E_i \in \Omega(\mathbf{E})$.

Method 1 (Intersection-of-intervals method)**.** *Let C, $\mathbf{E}$ and p be as before, and for every $i \in \{1, 2 \ldots, |\mathbf{E}|\}$, let E_i and $\mathbf{E}_i^-$ be as defined above. Let $\mathbf{e}_{ij}^-$ be the j-th element of an ordering of the domain $\Omega(\mathbf{E}_i^-)$ of $\mathbf{E}_i^-$. Now, let $I = [0, 1]$ and $i = 1$. While $i \leq |\mathbf{E}|$ and $I \neq \varnothing$, repeat the following steps:*

1. *Let $I_i = [0, 1]$ and $j = 1$. While $j \leq |\Omega(\mathbf{E}_i^-)|$ and $I_i \neq \varnothing$, repeat the following steps:*
   (a) *Compute I_{ij}, which is the union of all intervals of values x for p for which*
   $$0 \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_{ij}^-\right)\left(p = x\right) - \Pr\left(\bar{c} \mid e_i, \mathbf{e}_{ij}^-\right)\left(p = x\right).$$
   (b) *Compute $I_i = I_i \cap I_{ij}$ and $j = j + 1$.*
2. *Compute $I = I \cap I_i$ and $i = i + 1$.*

When applied for a specific parameter p, the intersection-of-intervals method returns a union of intervals I of values for p for which there are no violations of monotonicity in the resulting network B. More specifically, we have the following property for the union of intervals I.

Proposition 2. *Let B be a Bayesian network as before and let p be a parameter in B. Let I be the union of intervals of numerical values which results from applying the intersection-of-intervals method for p. Then, $I \neq \varnothing$ if and only if p is a resolvent parameter for B.*

Proof. We first assume that $I \neq \varnothing$. Then there must be some value $x \in I$ such that $x \in I_{ij}$ for all $i \in \{1, 2, \ldots, |\mathbf{E}|\}$, $j \in \{1, 2, \ldots, |\Omega(\mathbf{E}_i^-)|\}$. So, for every $E_i \in \mathbf{E}$ and each $\mathbf{e}_i^- \in \Omega(\mathbf{E}_i^-)$, the property $\Pr\left(\bar{c} \mid e_i, \mathbf{e}_i^-\right)\left(p = x\right) \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_i^-\right)\left(p = x\right)$ must hold. We conclude that varying p to x resolves all violations without introducing any new ones, which means that p is a resolvent parameter for B.

We now assume that p is a resolvent parameter for B. Then there must be some value $x \in [0, 1]$ for which for every $E_i \in \mathbf{E}$ and each $\mathbf{e}_i^- \in \Omega(\mathbf{E}_i^-)$ the property $\Pr\left(\bar{c} \mid e_i, \mathbf{e}_i^-\right)\left(p = x\right) \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_i^-\right)\left(p = x\right)$ holds. So, $x \in I_{ij}$ for all $i \in \{1, 2, \ldots, |\mathbf{E}|\}$, $j \in \{1, 2, \ldots, |\Omega(\mathbf{E}_i^-)|\}$. From the intersections performed, we have that the union of intervals I resulting from the method contains every value x' which is contained in all I_{ij} for all $i \in \{1, 2, \ldots, |\mathbf{E}|\}$, $j \in \{1, 2, \ldots, |\Omega(\mathbf{E}_i^-)|\}$. We conclude that $x \in I$, and hence $I \neq \varnothing$. □

In practical applications, to attain monotonicity for a network through a resolvent parameter p we must choose a numerical value from the union of intervals I which results from applying the intersection-of-intervals method. To this end, heuristics may be used. An example of such a heuristic would be to choose the value from I which is closest to the original value of p, thereby enforcing the smallest possible change in p; other examples of heuristics for parameter tuning are detailed in [7]. We would further like to note that it may be impractical to try and attain monotonicity for an entire network by parameter variation. As proposed by Van der Gaag *et al.* [8], it may be possible to resolve the violations of monotonicity that have arisen in a fixed context of values for some of the observable variables. Note that doing so need not resolve all monotonicity violations in the network; in fact, it may even introduce new violations for other contexts.

5.2 Application and complexity of the intersection-of-intervals method

From a computational point of view, the most expensive step in each iteration within the intersection-of-intervals method is step 1a. This step serves to compute, for a given observable variable $E_i \in \mathbf{E}$, the solution space I_{ij} for some assignment $\mathbf{e}_{ij}^-$ to $\mathbf{E}_i^-$. This space consists of all intervals of numerical values x for the parameter p under study for which there is no violation $viol(\mathbf{e}_{ij}^-)$ in the network, that is, for which

$$0 \leq \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_{ij}^-\right)\left(p = x\right) - \Pr\left(\bar{c} \mid e_i, \mathbf{e}_{ij}^-\right)\left(p = x\right). \tag{1}$$

The endpoints of the separate intervals in the solution space I_{ij} are the values $x \in [0, 1]$ for which

$$0 = \Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_{ij}^-\right)\left(p = x\right) - \Pr\left(\bar{c} \mid e_i, \mathbf{e}_{ij}^-\right)\left(p = x\right). \tag{2}$$

For computing these endpoints, we observe from previous studies of sensitivity functions that

$$\Pr\left(\bar{c} \mid \bar{e}_i, \mathbf{e}_{ij}^-\right)\left(p = x\right) = \frac{\alpha x + \beta}{\gamma x + \delta} \qquad \text{and} \qquad \Pr\left(\bar{c} \mid e_i, \mathbf{e}_{ij}^-\right)\left(p = x\right) = \frac{\alpha' x + \beta'}{\gamma' x + \delta'}$$

where $\alpha, \alpha', \beta, \beta', \gamma, \gamma', \delta$ and δ' are constants that are built from the parameter probabilities from the network other than p [5]. Using this notation, we now rewrite equality (2) to

$$0 = \frac{\alpha x + \beta}{\gamma x + \delta} - \frac{\alpha' x + \beta'}{\gamma' x + \delta'}.$$

This equality holds only if the following quadratic equation in x holds:

$$0 = (\alpha x + \beta)(\gamma' x + \delta') - (\alpha' x + \beta')(\gamma x + \delta). \tag{3}$$

Computing the intervals for the solution space I_{ij} thus amounts to solving the above quadratic equation to find the endpoints of the separate intervals, and determining on which side of these endpoints inequality (1) holds to find the intervals themselves. We note that for this purpose, we need to obtain the constants $\alpha, \alpha', \beta, \beta', \gamma, \gamma', \delta$ and δ'; an algorithm for computing these constants is readily available, however [9].

Example 4. *We consider again the example (restricted) network from Figure 2, and recall that it includes the two violations of monotonicity* $viol(\bar{e}_1)$ *and* $viol(\bar{e}_2)$*. We demonstrate the application of our intersection-of-intervals method for the parameter* $p = p(\bar{e}_2 \mid \bar{c}, \bar{e}_1, \bar{v}_1)$*. Let* $\mathbf{e}_{11}^- = \bar{e}_2, \mathbf{e}_{12}^- = e_2$ *and* $\mathbf{e}_{21}^- = \bar{e}_1, \mathbf{e}_{22}^- = e_1$*. In step 1a of the method, the following solution spaces are constructed:*

$$I_{11} = [0.114, \infty) \qquad I_{21} = [0.095, \infty)$$
$$I_{12} = (\infty, 0.852] \qquad I_{22} = \mathbb{R}$$

Note that these solution spaces consist of a single interval; this property does not hold in general, however. The method now computes

$$I_1 = I_1 \cap I_{11} = [0, 1] \cap [0.114, \infty) = [0.114, 1]$$

and subsequently establishes

$$I_1 = I_1 \cap I_{12} = [0.114, 1] \cap (\infty, 0.852] = [0.114, 0.852].$$

Having computed the solution space I_1 *for all* $\mathbf{e}_1^-$*, the method intersects the initial interval* $I = [0, 1]$ *with* I_1 *to obtain the new interval* $I = [0.114, 0.852]$*. The method then proceeds with the computation of* I_2*:*

$$I_2 = I_2 \cap I_{21} = [0, 1] \cap [0.095, \infty) = [0.095, 1]$$

and subsequently finds

$$I_2 = I_2 \cap I_{22} = [0.095, 1] \cap \mathbb{R} = [0.095, 1].$$

It now intersects the overall interval $I = [0.114, 0.852]$ *found so far with the solution space* I_2 *for all* $\mathbf{e}_2^-$*, which results in* $I = [0.114, 0.852]$*. We conclude that monotonicity can be attained for the example network by varying the parameter* $p(\bar{e}_2 \mid \bar{c}, \bar{e}_1, \bar{v}_1)$ *to any value from the interval* $[0.114, 0.852]$*.*

From applying the intersection-of-intervals method to the other parameters of the network, we find that monotonicity can be attained by varying the parameter $p(\bar{v}_1)$ *to a value from* $[0, 0.694]$*, by varying* $p(\bar{e}_1 \mid c)$ *to* 0*, or by varying* $p(\bar{e}_2 \mid c, \bar{e}_1, \bar{v}_1)$ *to a value from the interval* $[0, 0.212]$*; for all other parameters, the intersection-of-intervals method results in* $I = \varnothing$*, meaning that these parameters are not resolvent. To achieve monotonicity with the smallest change in parameter value, we must vary* $p(\bar{v}_1)$ *from* 0.760 *to* 0.694*.*

To address the computational complexity of the intersection-of-intervals method, we focus first on a single iteration of its step 1. We observe that, given quadratic equation (3) for a single parameter p and a single joint value assignment $\mathbf{e}_{ij}^-$, the actual intervals of the solution space I_{ij} can be computed in step 1a in constant time. For establishing the constants of this quadratic equation, an algorithm is available that takes polynomial time in the number of variables involved for networks with reasonable restrictions on their topology [9]; the runtime complexity of this algorithm is $O(n^2)$ where $n = |\mathbf{V}|$. Now, for a single variable E_i, step 1a of the intersection-of-intervals method needs to be performed a number of times at most equal to the size of $\Omega(\mathbf{E}_i^-)$, that is, 2^{m-1} where $m = |\mathbf{E}|$. Within a single iteration of step 1, therefore, our method can take $O(n^2 \cdot 2^{m-1})$ time for performing step 1a. To establish the amount of time spent on step 1b, we observe that, in general, the intersection of a union of k intervals with a union of l intervals can be computed in $O(k + l)$ time. In the jth execution of step 1b, our method takes the intersection of a union of at most j intervals with a union of at most 2 intervals, which can be done in $O(j)$ time. Since within a single iteration of step 1, step 1b is performed at most 2^{m-1} times, our method spends at most $O(4^m)$ time on this step.

In the overall loop of our method, step 1 is performed at most m times, each time taking $O(n^2 \cdot 2^{m-1})$ computations for step 1a and $O(4^m)$ time for step 1b; the method thus can take $O(n^2 \cdot m \cdot 2^m + m \cdot 4^m)$ time for step 1. We further find that in the ith iteration of the overall loop, step 2 can be executed in $O(i \cdot 2^m)$ time, since then a union of at most $(i-1) \cdot 2^{m-1} + 1$ intervals is intersected with a union of at most $2^{m-1} + 1$ intervals. As our method performs at most m iterations of the loop, it can spend $O(m^2 \cdot 2^m)$ time on step 2. Note that the time spent on this step is dominated by the time spent on step 1.

We conclude that the method has a runtime complexity of $O(n^2 \cdot m \cdot 2^m + m \cdot 4^m)$. For small sets of observable variables whose size can be considered constant with respect to n, the algorithm can run in $O(n^2)$ time; for sets of observable variables of size $O(n)$, however, the runtime complexity can increase to $O(n \cdot 4^n)$. Note that, in practice, the overall runtime is further increased by the observation that the number of parameters to be investigated for a given network can be exponential in n.

6 Conclusions

When a Bayesian network is employed in a real-life application, its users expect it to exhibit commonly acknowledged properties of monotonicity. In this paper, we studied the problem of attaining monotonicity for a network in which these properties are violated. We restricted our investigations to the problem of deciding whether changing the value of a single parameter probability to a new value can result in a network that does have the required properties. By building upon the previously known concept of sensitivity set, we showed that we can efficiently restrict a network to a part which is relevant for studying monotonicity. We further presented a method, called the intersection-of-intervals method, to compute, for a specific parameter of the restricted network, a union of intervals of numerical values to which this parameter can be varied in order to attain monotonicity. We showed that application of our method for every parameter in the restricted network can have a highly unfavorable runtime complexity, but argued that an efficient, generally applicable method could not be expected given the already high complexity of deciding monotonicity for a network.

Application of the intersection-of-intervals method to a Bayesian network can yield one or more parameters which each individually can be varied to attain monotonicity. We can then choose, for example, to change the parameter requiring the smallest amount of variation. Yet, the method may also uncover the impossibility of attaining monotonicity for a network by varying a single parameter. For such networks, it would be interesting to investigate attaining monotonicity by varying multiple parameters. We surmise that our method can be used to obtain a sequence of parameters which can be varied one after the other in order to attain monotonicity, although the result may not be optimal. Another option would be to vary several parameters simultaneously. We expect that for this option a fairly different method would be required, but that the results would be promising if such a method were to be found. Instead of attempting to attain monotonicity by varying one or more parameter probabilities from a Bayesian network, it may also be possible to do so by applying changes to its graphical structure. We hope to be able in the near future to report results from our further investigations.

References

[1] L.C. van der Gaag, H.L. Bodlaender, A. Feelders. Monotonicity in Bayesian networks. In: M. Chickering, J. Halpern (editors). *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (UAI)*, AUAI Press, Arlington, Virginia, 2004, pp. 569 – 576.

[2] L.C. van der Gaag, S. Renooij, P.L. Geenen. Lattices for studying monotonicity of Bayesian networks. In: M. Studený, J. Vomlel (editors). *Proceedings of the Third European Workshop on Probabilistic Graphical Models*, Prague, Czech Republic, 2006, pp. 99 – 106.

[3] H. Chan, A. Darwiche. When do numbers really matter? *Journal of Artificial Intelligence Research*, vol. 17, 2002, pp. 265 – 287.

[4] J.H.P. Kwisthout. *The Computational Complexity of Probabilistic Networks*. Ph.D. thesis, Utrecht University, 2009.

[5] V.M.H. Coupé, L.C. van der Gaag. Properties of sensitivity analysis of Bayesian belief networks. *Annals of Mathematics and Artificial Intelligence*, vol. 36, 2002, pp. 323 – 356.

[6] M.T. Rietbergen. *Restoring Monotonicity in Bayesian Networks*. Master's thesis, Utrecht University, 2010.

[7] H. Chan, A. Darwiche. A distance measure for bounding probabilistic belief change. *International Journal of Approximate Reasoning*, vol. 38, 2005, pp. 149 – 174.

[8] L.C. van der Gaag, H. Tabachneck-Schijf, P. Geenen. Verifying monotonicity of Bayesian networks with domain experts. *International Journal of Approximate Reasoning*, vol. 50, 2009, pp. 429 – 436.

[9] L.C. van der Gaag, U. Kjærulff. Making sensitivity analysis computationally efficient. In: G. Boutilier, M. Goldszmidt (editors). *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)*, Morgan Kaufmann Publishers, San Francisco, 2000, pp. 317 – 325.