Inference and Attack in Bayesian Networks

Sjoerd T. Timmer [a] John-Jules Ch. Meyer [a] Henry Prakken [ab]

Silja Renooij [a] Bart Verheij [cd]

[a] *Department of Information and Computing Sciences, Utrecht University*
[b] *Faculty of Law, University of Groningen*
[c] *Institute of Artificial Intelligence, University of Groningen*
[d] *CodeX, Stanford University*

Abstract

In legal reasoning the Bayesian network approach has gained increasingly more attention over the last years due to the increase in scientific forensic evidence. It can however be questioned how meaningful a Bayesian network is in terms that are easily comprehensible by judges and lawyers. Argumentation models, which represent arguments and defeat, are arguably closer to their natural way of arguing and therefore potentially more easy to understand for lawyers and judges. The automated extraction of rules, arguments and counter-arguments from Bayesian networks will facilitate the communication between lawyers and judges on the one hand and forensic experts on the other. In this paper we propose a method to automatically extract inference rules and undercutters from Bayesian networks from which arguments can subsequently be constructed.

1 Introduction

One of the major problems in legal reasoning about evidence is that lawyers and judges have difficulties with the interpretation of statistical evidence. The growing popularity of scientific evidence such as DNA and fingerprint matching poses a serious challenge. A need for scientifically founded methods to assess the value of probabilistic evidence has arisen. This is illustrated by some of the recent miscarriages of justice, see for instance the notorious cases of Sally Clark and Lucia de Berk.

There are a number of different modelling methods for evidential reasoning that have gained popularity over the last years. Closest to the nature of probabilistic evidence is the approach based on Bayesian networks (BN's). A Bayesian network completely defines the probability distribution over all variables of interest. That is, given a Bayesian network any probability over its variables can be calculated. Forensic scientists often use Bayesian network models to model their findings [3, 4, 8, 12].

A drawback of the use of Bayesian networks is that a BN's meaning is not very intuitive and therefore hard to understand. Supporters of this approach often argue that the structure of the graph is an intuitive representation of the relation between the variables, but in reality the meaning of the graph is often not easily explained. This is due to the fact that edges, and in particular the direction of the edges, suggest causality, but in fact have no intuitive interpretation in Bayesian networks.

An alternative way of modelling such evidence is presented by research in argumentation. Argument models are presumably more intuitive and natural to follow for lawyers and judges. Reasoning about arguments, inferences and counter arguments is something that requires less mathematical insight for judges and lawyers to understand.

In this paper[1] we will combine Bayesian networks and argumentation in order to bridge the gap between forensic and legal experts. We do so by proposing a method to extract arguments and counterarguments from a Bayesian network. This research is an explorative first step into the combination two modelling methods. The resulting understanding is necessary to facilitate reasoning with statistical evidence in law.

Inspiration for this approach was taken from Vreeswijk [14] and Williams and Williamson [15]. Some similarities and differences will be discussed in Section 5. We focus on the construction of arguments from a BN as opposed to, for instance, Lacave and Díez [6, 7] who have proposed other kinds of explanation methods for Bayesian networks.

In Section 2 the concepts from the fields of argumentation and Bayesian networks will be briefly introduced. Section 3 describes our method to extract arguments and undercutters from a BN, followed by an example in Section 4 to illustrate how this method handles different aspects of probabilistic reasoning. We will reflect on the method and compare it to the work of others in the discussion in Section 5.

2 Background

2.1 Argumentation

As mentioned, we are interested in arguments about a legal case for which we have a Bayesian network model. Various methods exist to obtain such a network but that is not the topic of this paper. In general arguments consist of inference rules which have a conclusion and one or more premises. Traditionally these inference rules are deductive, which means that the conclusion cannot be false whenever the premise is true. Starting with work of Pollock [11], defeasible reasoning has gained popularity. Defeasible arguments or inference rules are not strict in the sense that their conclusion may prove to be false in the future even when their premise is true. Their premise merely indicates the truth of their conclusion until further information is added. Rules that are not strict but defeasible are written as $r : \varphi \Rightarrow \psi$. Strict rules are written as $r : \varphi \rightarrow \psi$.

Inference rules can be combined into arguments. Where inference rules are the atomic steps of reasoning, arguments represent the broader reasoning. We will adhere to the ASPIC+ framework for argumentation [9]. We will construct arguments with both strict and defeasible rules and we use preferences to resolve conflict. The ASPIC+ framework is most suitable for this situation. We will only discuss the informal intuition of the framework and refer the interested reader to the work of Modgil and Prakken [9] for a formal account of these matters. The ASPIC+ framework takes the following as input:

- a logical language $\mathcal{L}$
- a unary negation function that maps language elements to their opposites.
- defeasible rules with a $\leq$ relation defined over them
- strict rules
- a knowledge base: $\mathcal{K} \subseteq \mathcal{L}$.

The advantage of this approach is that once the notions of inference and undercut[2] are defined the framework defines the arguments that can be constructed and how they attack and defeat (successfully attack) each other. In the ASPIC+ framework an argument is either:

- an item from the knowledge base
- the application of a strict rule to the conclusion of one or more existing arguments
- the application of a defeasible rule to the conclusion of one or more existing arguments

ASPIC+ combines inferences into arguments by connecting the conclusions of one or more arguments to the premises of an inference rule. As such, an argument can be seen as a tree of inferences where each subtree represents a subargument.

[1] This research is part of an NWO forensic science project (`http://www.ai.rug.nl/~verheij/nwofs/`), which aims to combine Bayesian networks with narrative approaches [13] and argumentation. This builds on the work of Bex [1] who combined narrative and argumentation models.

[2] Note that the framework also has a notion of undercutter that follows closely from the undercutting or inference rules.

Three modes of attack amongst arguments are distinguished: 'rebuttal', 'undermining' and 'undercutting'. A rebuttal of an argument is the contradiction of its conclusion or the conclusion of one of its subarguments. The second way to attack an argument is to show that one of the premises is false. This method of attacking an argument is called undermining. The third way in which an argument can be defeated is to directly attack the inference itself. To say that an inference is not valid (or does not hold in the specific case) voids the argument. This is called undercutting an argument. An undercutter of an argument is another argument that invalidates the application of the defeasible rule in that argument. We abuse language to refer to an inference rule which negates another inference rule as an undercutter as well, although such an inference rule is formally not an argument by itself. To express the validity or invalidity of an inference ASPIC+ assumes a function that maps defeasible inferences to names in the object language. When we write an inference rule $r : \varphi \Rightarrow \psi$, the variable r will be the name of the inference rule. This enables us to reason about undercutters. Undercutting always succeed and will therefore always result in defeat. This is not true for rebutting and undermining. A rebuttal will succeed when it is not itself rebutted by the attacked argument. A similar condition holds for underminders. Otherwise the success of the rebuttal or underminer is determined by an admissible argument ordering. A commonly used ordering that we will adhere to is the so called weakest-link principle that orders arguments on the strength of the weakest defeasible rule in the argument. The set of successful attacks defines the defeat relation. These attacks are used to apply Dung semantics [2].

2.2 Bayesian networks

In probability theory models are specified in terms of variables. In probabilistic legal reasoning, for instance, a variable could be used to model whether or not two fingerprints match. It is not uncommon to model certain aspects of the psychological state of the suspect as a probabilistic variable as well. All of the above examples are discrete variables with two possible values, but variables with three or more values are also possible. These values are mutually exclusive (at most one is true at once) and collectively exhaustive (at least one is true). For simplicity, only binary-valued variables will be considered from here on. A probability distribution over a number of variables defines the probability of every possible combination of value assignments to these variables. Without additional information it is impossible to characterize a probability distribution in any more efficient way than to enumerate all possible instantiations with their respective probabilities. However, in practice additional information is often available in the form of probabilistic independences. Two variables A and B are said to be independent when $P(A) = P(A|B)$, or equivalently if $P(B) = P(B|A)$. Two variables A and B can also be independent given a set X of other variables. This is the case when $P(A|X) = P(A|BX)$.

A Bayesian network [10] is a model that represents a probability distribution that satisfies a certain independence relation. It consists of a directed graph with one node for every variable. The nodes are connected by directed edges. To define the full joint probability distribution ($P(ABC\ldots)$) only the conditional probabilities of the variables given their parents in the graph are required:

$$P(ABC\ldots) = \prod_{X \in \{A,B,C,\ldots\}} P(X|\text{parents(X)}) \tag{1}$$

Independence between variables can be read from the graph by means of so called d-separation. Two variables X and Y are d-separated by a set of observed variables Z iff every trail from X to Y is blocked by Z. A trail is blocked by a set Z iff either (1) at least one node on the trail which does not have two incoming edges on the trail is in Z. (2) or at least one node on the trail which does have two incoming edges on the trail is in Z or has a descendant in Z. The term d-connected is used as the opposite of d-separated.

Whenever two variables are d-separated according to the graph they must be probabilistically independent. This is known as the d-separation criterion and it justifies the above factorisation of the probability distribution.

3 Extracting rules and undercutters from a Bayesian network

In this Section we will describe a method to extract inference rules and undercutters of those rules from a Bayesian network. In Section 4 we will show what kind of arguments can be built from these inferences. One of the difficulties with argumentation is that it is hard to capture the strength of an inference rule. More specifically, it is hard to decide what rules (and therefore which arguments) must have precedence. We will use a probabilistic method to allow for a numerical valuation of inferential strength. From this numerical strength an ordering of inference rules is easy to obtain.

3.1 Inference rules and strength

We will consider every pair of value assignments as a candidate for a rule. The first value assignment is the premise, the second the conclusion. We immediately exclude all rules for which the variables of the premise and the conclusion are the same or d-separated. For simplicity we will consider inference rules with a single premise from here onwards.

Since we assumed that nodes are binary we can refer to the positive outcome of a variable A as a and to the negative outcome as $\neg a$. An inference rule is then a statement of the form $\varphi \Rightarrow \psi$, where φ and ψ are propositions derived from two different variables in the network. So, both φ and ψ can be statements like 'variable A has value a' or 'variable B has value $\neg b$'. We will use the shorthand notation $\neg\varphi$ to denote the other value of the same variable.

We are interested in the strength of inferences so we need a measure of inferential strength that is based on the probability distribution defined by the Bayesian network. Note that this measure of strength is based on the probabilities encoded in the network and not on the independence relation that is modelled by the BN graph. Consider an inference rule $\varphi \Rightarrow \psi$.We will adhere to the following definition of inferential strength $s(r)$ of rule $r : \varphi \Rightarrow \psi$:

$$s(r) = \frac{P(\psi|\varphi)}{P(\psi|\neg\varphi)} \tag{2}$$

If we interpret probabilities as degrees of belief then this expresses the factor by which our belief in the consequent grows when the evidence is changed from false to true. Many alternative measures of strength are imaginable and we will briefly discuss a number of them in Section 5.

3.2 Undercutters of rules

The rules that have a high strength according to the definition above are still defeasible in the sense that other observations may weaken or even completely nullify the positive effect of φ on ψ. How then can these inferences be attacked? We will say that a value assignment ρ undercuts the rule $\varphi \Rightarrow \psi$ when the measure of strength no longer yields a strength above one when both probabilities are conditioned on ρ as well. For instance, for the measure of strength as described above, we consider the fraction

$$u(r, \rho) = \frac{P(\psi|\rho\varphi)}{P(\psi|\rho\neg\varphi)} \tag{3}$$

When this fraction is less than one we will conclude that ρ undercuts the inference from φ to ψ. This then yields a method to extract undercutters of inference rules. Undercutters, as opposed to the inference rules, do not need a strength since they always succeed.

3.3 Extraction algorithm

Given a Bayesian network we are now interested in inference rules (and ultimately arguments) that follow from that network. We have identified what an inference rule is and how it probabilistically relates to the model. We will now describe how these inference rules and undercutters can be extracted from a Bayesian network. The process is algorithmically split in two phases. In the first phase all candidates are enumerated

and in the second phase the ones with an insufficient strength (less than or equal to one with our choice of $s(r)$, since at a value of one the influence turns from positive to negative) are discarded. The enumeration of candidates takes place by combining every possible variable value as a premise with every possible variable value as a conclusion. Of course, pairs of value assignments to the same variable are not considered and neither are value assignments to nodes that are d-separated (and therefore probabilistically independent). To summarize, the general structure of the algorithm to extract inference is presented as pseudo-code in Figure 1a.

input : Bayesian network G with nodes N
output: defeasible inference rules R
for $N_1 \in N$ **do**
  for $N_2 \in N$, $N_1 \neq N_2$ **do**
    if *not* `d-separated(`N_1,N_2`)` **then**
      for $v_1 \in$ `values(`N_1`)` **do**
        for $v_2 \in$ `values(`N_2`)` **do**
          $r \leftarrow (v_1 \Rightarrow v_2)$;
          if $s(r) > 1$ **then**
            add this rule r to R

(a) Pseudo-code to extract inference rules.

input : defeasible rules R
output: Undercutters U
for $r : \varphi \Rightarrow \psi \in R$ **do**
  for $\rho \in$
  `d-connected-variables(`$\psi|\varphi$`)` **do**
    if $u(r, \rho) \leq 1$ **then**
      add undercutter $\rho \rightarrow \neg r$ to U

(b) Peudo-code to extract undercutters of rules.

Figure 1: pseudo code for the extraction of inference rules and undercutters.

We do something very similar for undercutters. For every rule we try to find undercutters by considering every possible single variable assignment as a candidate to be an undercutter. So, for every rule $\varphi \Rightarrow \psi$ all possible value assignments ρ to other variables are considered as a candidate undercutter. Here again, we discard the undercutters that are probabilistically independent of the conclusion ψ given the premise. In pseudo-code the algorithm to extract undercutters is given in Figure 1b.

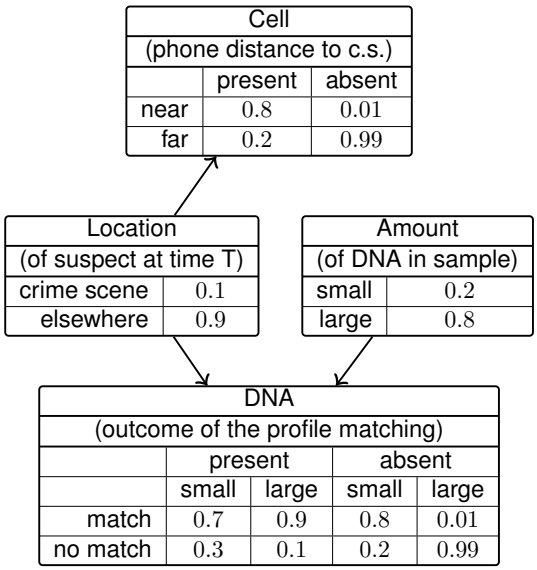
3.4 Argument construction

With the extracted inference rules and undercutters, we proceed by constructing arguments using the ASPIC+ framework. This means that we have to define the different elements of the argumentation framework.

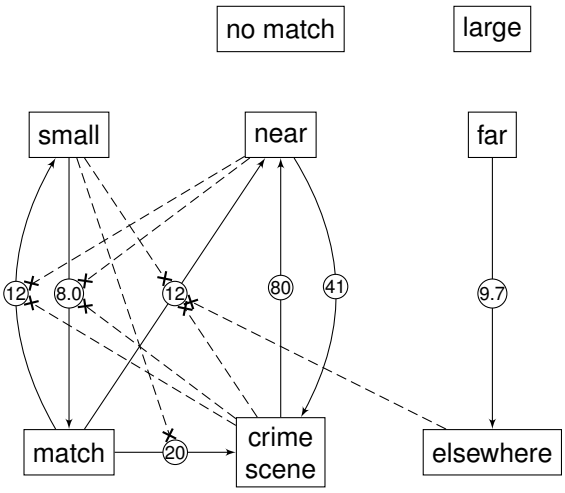
- for the logical language $\mathcal{L}$ we take all value assignments to a node. The language will also contain an element r for every extracted defeasible rule that represents the validity of that rule.
- the negation function maps every value assignment onto the other value assignment of the same node.
- the extracted inference rules are used as the defeasible rules. The ordering follows naturally from the strength of the rules.
- undercutters are captured in strict rules.
- the knowledge base contains the set of observed variable values. To be able to speculate about variables that are not observed we add both values of these nodes to the knowledge base as well. In other contexts it may be more appropriate to model these differently. For instance as a social kind of 'ordinary' knowledge or as defeasible rule without a premise.

4 Example case

To show the result of the described method we present a small fictive case. The network is shown in Figure 2 together with the extracted arguments. It represents part of a fictive case about the location of a suspect and two pieces of evidence related to this location. We suppose that the outcomes of the DNA matching test and the cell phone localization are observed as well as the amount of DNA in the sample, but to illustrate the approach we add these one by one to show how the resulting arguments change. Let us first



(a) Example Bayesian network representing the suspect's presence at the crime scene, two possible pieces of evidence (a DNA test and the location of the suspect's cell phone at the time of the crime) and the amount of DNA that was recovered from the DNA trace.



(b) The inference rules and their strength extracted from the example BN. Every inference rule is displayed as an arrow with a small circle halfway that states the strength. undercutters are displayed as a dashed, cross-tipped arrow pointing to the circle of the inference. Only inferences with a strength greater than five are shown to prevent visual clutter. As a result several rules, such as for example *no_match*⇒*large*, are not visible in the graph.

Figure 2: The example network and the extracted inference rules visually represented.

assume that *DNA=match* and *Amount=small* were observed. This evidence is entered by adding it to the knowledge base $\mathcal{K}$. Some arguments can already be built with this knowledge. For instance, with the rule *DNA=match*⇒*Location=crime scene* we can derive an argument with the conclusion *Location=crime scene*. This argument can be undercut by an argument for *Amount=small*. Both arguments and their undercutting attack are shown in Figure 3.

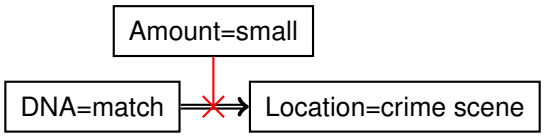


Figure 3: The upper argument undercuts the lower.

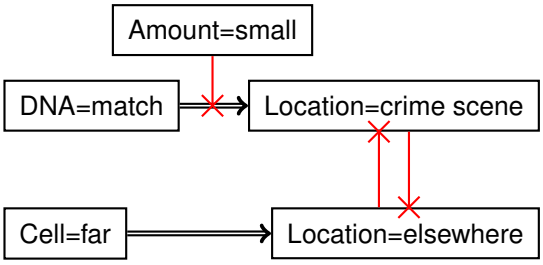


Figure 4: A rebuttal is added for the location of the suspect.

Let us now add the evidence for the cell phone location. We suppose that the cell was observed by a GSM tower *far* away from the crime scene. A third argument can thus be constructed from *Cell=far* to *Location=elsewhere*. This is Visualised in Figure 4. Since *Location=elsewhere* and *Location=crime scene* are both values of the same node they negate each other and will therefore also rebut each other. As we have described, the ASPIC+ framework resolves this bidirectional attack by comparing the arguments on their weakest inference rule. So, besides for the selection of rules, the strengths of inferences are used again to determine argument defeat. In this particular case the argument for *Location=crime scene* defeats the argument for *Location=elsewhere*. The resulting defeat relation between the arguments is shown in

Figure 5. Note that here we have drawn all arguments including the trivial sub-arguments (B and D) for *DNA=match* and *Cell=far*. These arguments do not play a role in the defeat relation but we show them for completeness because without them the arguments C and E could not exist.

To these arguments and the defeat relation we can apply any Dung-style semantics [2]. The set $\{A, B, D, E\}$ is a grounded, stable and preferred extension of the argument framework. This means that argument E, although it defeats C, will be defended by A. We can also see that the sub-argument B is by itself not incompatible with A and therefore the conclusion E can be accepted without disregarding any evidence. The posterior probability $P(\text{absent}|\text{match}, \text{small}, \text{far})$, which we can calculate with a traditional Bayesian network approach, is 0.98. The outcome of the argumentation approach is thus in accordance with what was to be expected from the Bayesian network.

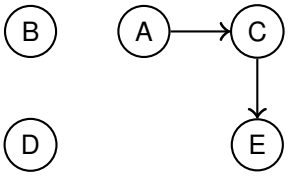


Figure 5: The abstract argument graph with the defeat relation. A is the argument for *Amount=small*, C is the argument for *Location=crime scene* and E is de argument for *Location=elsewhere*. Arguments B and D are the sub-arguments for *DNA=match* and *Cell=far*.

5 Conclusions

We have described how inference rules can be extracted from a Bayesian network and combined into arguments using APIC+. This is an important step in the connection between two of the approaches of reasoning with evidence. From the Bayesian approach we inherit the possibility to express properties in a numerical way and from the argumentation approach we inherit the notion of argument and attack.

We used one particular measure of strength to illustrate the approach but any other measure of strength could easily be substituted. Many alternatives to this measure of strength exist. Which measure is most appropriate must be decided by further research. Possibilities for this choice include $P(\psi|\varphi)$, $\frac{P(\varphi|\psi)}{P(\varphi|\neg\psi)}$, $P(\psi|\varphi) - P(\psi|\neg\varphi)$ and $\frac{P(\psi|\varphi)}{P(\psi)}$, but many more are imaginable and all have certain advantages and disadvantages.

5.1 Related work

The idea to use Bayesian networks to assist the process of argument selection is not new. Williams and Williamson [15] have also proposed a method to select strong probabilistic arguments. Vreeswijk [14] has proposed a method to extract arguments from Bayesian networks. What our approach shares with the mentioned methods is that we take node values as the premises and conclusions of the inference rules. However, we introduce a method for extracting inference rules from Bayesian networks, which eliminates a number of shortcomings in Vreeswijk's approach. For instance, Vreeswijk's method only extracts inference rules in the direction of the arc in the network, which is not sufficient because the arcs in a Bayesian network do not necessarily bear inferential information.

One of the key differences with the work of Williams and Williamson is that they consider every inference rule to be either strong or unacceptable. They just check that $P(\psi|\varphi)$ is greater than $P(\psi|\neg\varphi)$. In our method inference rules can be compared on the basis of their strengths. In addition, it seems from their examples that they only consider pairs of variables that have a parent-child relation in the graph, but it is not specified how they select the candidate rules for inferences. They also have an ambiguity in the definition of the extraction rule. The already extracted rules can prevent certain other rules from being discovered. Therefore the order in which they are added is important, but this order is left unspecified.

A similar approach to ours has been taken by Keppens [5] who proposes a method to construct so called Argument Diagrams from Bayesian Networks. A key difference is that Argument Diagrams express only one view on the case whereas an Argumentation System can express argument for different views on the case.

5.2 Future research

The approach should be extended to handle rules with more than one premise. Very often it is the combination of factors that causes the effect while neither of the individual factors could produce the same effect.

The influence of the choice of Dung semantics should be investigated since this will certainly make a difference in the final results.

Currently, the strengths of rules are computed from the prior network and therefore do not include the effects of evidence on strength. As a result, some arguments may remain undetected. In the near future we plan to extend our method to include the updating of strengths upon entering evidence.

References

[1] F. J. Bex. *Arguments, Stories and Criminal Evidence*, volume 92 of *Law and Philosophy Library*. Spinger, 2011.

[2] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 2005.

[3] N. E. Fenton, M. Neil, and D. A. Lagnado. A general structure for legal arguments about evidence using Bayesian Networks. *Cognitive Science*, 37(1):61–102, 2013.

[4] A. B. Hepler, A. P. Dawid, and V. Leucari. Object-oriented graphical representations of complex patterns of evidence. *Law, Probability & Risk*, 6(1-4):275–293, 2007.

[5] J. Keppens. Argument diagram extraction from evidential Bayesian Networks. *Artificial Intelligence and Law*, 20(2):109–143, 2012.

[6] C. Lacave and F. J. Díez. A review of explanation methods for Bayesian Networks. *Knowledge Engineering Review*, 17(2):107–127, 2002.

[7] C. Lacave, M. Luque, and F. Diez. Explanation of Bayesian Networks and influence diagrams in elvira. *Systems, Man, and Cybernetics, Part B*, 37(4):952–965, 2007.

[8] K. B. Laskey and S. M. Mahoney. Network fragments: Representing knowledge for constructing probabilistic models. In *Uncertainty in Artificial Intelligence*, pages 334–341. Morgan Kaufmann, 1997.

[9] S. Modgil and H. Prakken. A general account of argumentation with preferences. *AI Journal*, 195:361–397, 2013.

[10] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[11] J. L. Pollock. Defeasible reasoning. *Cognitive Science*, 11:481–518, 1987.

[12] Q. Shen, J. Keppens, C. Aitken, and B. Schafer. A scenario-driven decision support system for serious crime investigation. *Law, Probability & Risk*, 5(2):87–117, 2006.

[13] C. Vlek, H. Prakken, S. Renooij, and B. Verheij. Modeling crime scenarios in a bayesian network. In *ICAIL*, 2013. to appear.

[14] G. A. Vreeswijk. Argumentation in Bayesian belief networks. In I. Rahwan, P. Moraïtis, and C. Reed, editors, *Argumentation in Multi-Agent Systems*, volume 3366 of *Lecture Notes in Computer Science*, pages 111–129. Springer Berlin / Heidelberg, 2005.

[15] M. Williams and J. Williamson. Combining argumentation and Bayesian Nets for breast cancer prognosis. *Journal of Logic, Language and Information*, 15(1–2):155–178, 2006.