Belief Propagation in Fuzzy Bayesian Networks: A Worked Example

Christopher Fogelberg

August 28, 2008

Abstract

Fuzzy Bayesian networks (FBN) are a model representation for machine learning techniques. They are graphical structures with variables that are simultaneously fuzzy and uncertain.

Although very similar to classic discrete or multinomial Bayesian networks (BN), and able to take advantage of existing BN techniques and algorithms, belief propagation in a FBN is different and can be too slow on some graphs. Propagating expected values effectively addresses this problem. Because the variables are simultaneously fuzzy and uncertain it can also be confusing. In this paper we summarise fuzzy Bayesian networks and provide examples of forward propagation, backward propagation and explaining away for the same classic BN and FBN.

By comparing, contrasting and interpreting classic BN and FBN side by side, belief propagation in FBN is made clearer, and the strengths of FBN over classic BN in certain situations are also highlighted.

1 Introduction

Fuzzy Bayesian networks (FBN) are a graphical machine learning model representation with variables which are simultaneously fuzzy and uncertain[3]. This paper describes the formalisation of FBN, introduces our notation, touches on the conceptual foundations and illustrates belief propagation in FBN.

By doing this it reifies FBN belief propagation and makes the formalism easier to understand. Further, it illustrates the advantages and expressivity of FBN in certain situations. We briefly extend this contrast to discuss the interpretation of fuzzy variables, and what it means for a variable to be simultaneously fuzzy and uncertain. Readers are referred to the original paper[3] and current (in progress) research[2] for a more detailed discussion of these questions.

1.1 Motivation

Bayesian networks[5; 7; 12; 14] (BN) are commonly used in machine learning. This is due to their statistical rationality, capacity for rigourous causal inference, and robustness in the face of noisy, partially missing and realistic data. They are also more easily human-interpretable than other machine learning representations such as neural networks, and experts can specify prior knowledge in a principled manner to guide the machine learning search. A wide range of search algorithms have been developed for structural and parameter inference, including structural EM[4] and MCMC. Classically, Bayesian networks use continuous (Gaussian) or multinomial variables.

Similarly, a fuzzy model has a wide range of advantages. Fuzzy models are also robust in the face of noise-corrupted data. The use of linguistic terms aids human comprehension of the learnt model, and they are particularly useful when the data is insufficient to formulate a precise model. The need to specify membership functions also forces the designer to consider the semantic interpretation of the model parameterisation and construction more explicitly.

For these reasons, FBN (which combine these advantages) may be useful. Theoretical analysis in current research[2] also indicates that fuzzy variables can be more expressive than multinomial or continuous variables. Further, FBN may be used as part of an integrated sequence of machine learning techniques that include reversible dimensionality reduction techniques such as fuzzy cover clustering algorithms. This may allow larger problems to be addressed with FBN than with classic BN.

1.2 Structure

Section 2 summarises fuzzy Bayesian networks. Section 3 presents a Bayesian network (its structure and parameters) for classic and fuzzy BN. Forward propagation on this network is shown in section 4, back propagation in section 5, and explaining away in section 6.

Section 7 summarises the differences in belief propagation over a classic BN and over a fuzzy BN, and discusses how the results can be interpreted. This highlights the strengths of FBN as a model representation.

Variables in this paper are denoted with capital letters. A variable has a *state*, which for a Boolean variable is either the *value* true or the value false. All variables are Boolean and the discrete state of a variable is denoted with a lower case letter, e.g. $A = true$ and a are equivalent, as are $A = false$ and $\neg a$. By extension, $p(a)$ denotes the probability that $A = true$, and $p(A)$ is the probability distribution over A.

2 What is a Fuzzy Bayesian Network?

A Bayesian network is a graphical model of the probabilistic dependencies among a set of variables. A FBN is a Bayesian network which has fuzzy variables. Classically, a variable's states may be either discrete (multinomial) or continuous. This section introduces the essential elements of our work, and readers are referred to the original paper[3] for a more complete presentation.

It is important to note that fuzziness and probability are distinct. Previous research which has combined fuzziness and BNs has arbitrarily mixed them, using fuzziness as an approximation so that intractable problems can be tractably approximated[8; 10]. Although other mathematical research has also considered the rigourous unification of fuzziness and probability[13], ours appears to be unique in that it explicitly maintains the distinction while considering them simultaneously.

Our formalism consists of two key elements. First, there is the conceptual unification (subsection 2.1) and notation for multinomial probability distributions and fuzzy variables (subsection 2.2). Secondly, there are the fuzzy uncertainty semantics (subsection 2.3) and how this and the conceptualisation affects belief propagation (subsection 2.4).

2.1 A Conceptual Overview of FBN

The two most important ideas in FBN are simple:

- A variable in the network may be simultaneously fuzzy and uncertain.
- Uncertainty and fuzziness are not conflated.

Importantly, a FBN may have the same structure and parameters as a classic BN. In fact, a classic BN that has been learnt can have fuzzy beliefs propagated over its variables without modifying the structure or conditional distributions[1].

[1]Depending on the nature of the relationship between the underlying variable and the fuzzified state though, the previously learnt conditional distributions may not be optimal.

A fuzzy BN differs from a classic BN only in the type of its variables, and in how belief propagation is conducted. This means that algorithms such as *structural EM*[4], *MCMC*[6] and the junction-tree algorithm[9] for belief propagation in graphs which aren't polytrees can be used, and they only need to be slightly modified to take FBN belief propagation into account. We do not consider network inference in this paper. Belief propagation is summarised in subsection 2.4, and sections 4–6 show some examples. Subsection 2.2 introduces the notation we use to express the formalisation and describe belief propagation.

2.2 Some Notation

To facilitate clear presentation, we need to be able to denote the state of a fuzzy, observed variable differently than the state of a discrete, uncertain variable or the state of a fuzzy, uncertain variable.

2.2.1 Fuzzy States

A fuzzy state is denoted with square brackets surrounding its *components*. Each component is one of the *fuzzy values*[2], that the variable can take on and is subscripted by the variable's membership in that fuzzy value. For example, $C = [f_{0.3}, t_{0.7}]$ says that C has 0.3 membership in false and 0.7 membership in true. This might denote that it was cloudy for 0.7 (70%) of the day, and not cloudy for 30% of the day.

If we use lower case letters to denote the discrete states true and false then $c = [t_1]$ and $\neg c = [f_1]$. The types of relationships shown in equation 1 also hold amongst fuzzy states.

$$[f_{0.3}, t_{0.6}, f_{0.1}] = [f_{0.4}, t_{0.6}] = [t_{0.6}, f_{0.4}] \tag{1}$$

2.2.2 Probability Distributions

We consider BN with multinomial variables, and denote probability distributions (PD) with curly brackets. Each element of the *range* of a PD (usually the same range as the variable) has its probability subscripted. For example, a sample from the probability distribution $\{lo_{0.4}, mid_{0.5}, hi_{0.1}\}$ is *lo* 40% of the time, and *mid* or *hi* 50% and 10% of the time, respectively.

[2]Classically referred to as *fuzzy sets*, and analogous to the *values* that a multinomial variable may take on, e.g. *hi*, *mid*, *lo*, or *true* and *false*.

2.3 Fuzzy Probability Distributions and Mixed States

We also need to be able to denote a *fuzzy probability distribution* and sample from it, and a variable which is fuzzy, partially uncertain and partially *observed*. The notation given in subsection 2.2 is extended to denote this.

2.3.1 Fuzzy Probability Distributions and Sampling

A fuzzy probability distribution is an uncertain component of a fuzzy state. Just as we annotate an observed (certain, definite) component of a fuzzy value with the state's μ in that value (e.g. $f_{0.3}$) we annotate a component probability distribution with a μ value, and this represents the state's certain membership that probability distribution, e.g. $[\ldots, \alpha = \{f_{0.4}, t_{0.6}\}_{0.5}, \ldots]$

As discussed in our current research[2], there are several ways of sampling from a probability distribution over a fuzzy variable. With FBN, we have adopted the convention that a sample from a FPD is only and entirely one value, and each value is drawn with its associated probability. For example $f_{0.5}$ will be drawn as a sample from the FPD α 40% of the time, and $t_{0.5}$ will be drawn 60% of the time. These are the only samples which could possibly be drawn from α.

Note that this approach relaxes the tight interdependencies amongst the fuzzy values. Consider a variable with membership in the fuzzy values *hi*, *mid* and *lo*. Usually its membership in *mid* would also define its membership in *hi* and *lo*, and vice versa. However this is not necessarily true of a fuzzily uncertain variable. We do not consider the theoretical implications further in this paper, but refer interested readers to our unpublished note[2] which indicates that it increases the expressive power of a variable in certain situations.

2.3.2 Mixed States

Mixed states combine the notation for FPD and fuzzy values and represent variables which are partially observed and partially uncertain. For example, a sample from the partially observed state $C = [f_{0.3}.t_{0.6}, \{f_{0.2}, t_{0.8}\}_{0.1}]$ will be $[f_{0.4}, t_{0.6}]$ with probability 0.1, and $[f_{0.3}, t_{0.7}]$ with probability 0.8. If C denotes a cloudy sky this might denote having observed the sky for 70% of the day, or an incomplete and uncertain weather forecast. We further discuss the interpretation of fuzzy states in section 7.1.

2.3.3 Implicit Assumptions in Fuzzy Uncertainty

To summarise, note that we have made three assumptions for FBN, and two are relevant to FPD. Firstly, a variable's membership in its set of fuzzy values always sums to 1, and the sampling strategy we have adopted maintains this. Secondly, FPD samples are *monolithic*; a sample from a FPD will only be of one value.

2.4 Belief Propagation in FBN

Belief propagation in FBN is very similar to belief propagation in classic BN, and the examples provided in sections 4–6 will be clearer than a formal explanation. However, this section highlights the third key assumption we have made to make FBN inference tractable, and explores the consequences of that assumption for belief propagation to variables with multiple parents.

The third assumption is that components within a variable don't interact during belief propagation. Thus if $C = [f_{0.3}, t_{0.7}]$ then the first component ($f_{0.3}$) is propagated through the conditional distributions of C's children exactly as in classic BN belief propagation without regard to the remainder of C's components, and C's child will have μ 0.3 in the result of propagating $f_{0.3}$.

The second component is propagated in a similarly independent way, which means that all variables which only have C as a parent will have two components in their updated state: one with membership 0.3 and propagated according to the C-value f, and one with membership 0.7 and propagated according to the C-value *true*.

If a variable has more than one parent, and if its parents each have more than one component, then we need to combine the parent's components. The selected way of doing this is with the Cartesian product of the sets of parent components, and μ for each member of the Cartesian product can be calculated using the product t-norm[1].

However, representing a variable's state with multiple independent components and combining them with the Cartesian product means that probabilistic propagation is often intractable. For example, if C and a coparent each have two components then their child will have four. If the child's coparent also has four components then the grandchildren will have 16, and so forth.

Fogelberg et al. [3] outlines four ways of addressing this problem. Propagating the fuzzy expected value, which can be calculated using *fuzzy integration*, means that FBN propagation is usually only a small factor less efficient than classic BN propagation. Further, propagating the fuzzy expected value does not appear to have any disadvantages when compared to classic probabilistic propagation, and the graphs that it is markedly worse for are often intractable as classic BN anyway. For that reason it is adopted in this paper.

Consider the variable $X = [\{f_{0.4}, t_{0.6}\}_{0.2}, \{f_{0.8}, t_{0.2}\}_{0.5}, \{f_{0.1}, t_{0.9}\}_{0.3}]$. The fuzzy expected value of X is easily calculated analytically, and this is shown in equation 2.

$$\begin{aligned} X &= [\{f_{0.9}, t_{0.1}\}_{0.2}, \{f_{0.8}, t_{0.2}\}_{0.5}, \{f_{0.1}, t_{0.9}\}_{0.3}] \\ &\rightarrow [f_{0.9\times0.2+0.8\times0.5+0.1\times0.3}, t_{0.6\times0.2+0.2\times0.5+0.9\times0.3}] \\ X &\rightarrow [f_{0.61}, t_{0.39}] \end{aligned} \tag{2}$$

3 An Example Network

Figure 1 shows the structure of the BN we will be using. It is a classic example, and describes the relationships between the sky (C, cloudy or not), whether or not it rains (R), whether or not the sprinkler is turned on (S) and the state of the grass (wet, or not, W).

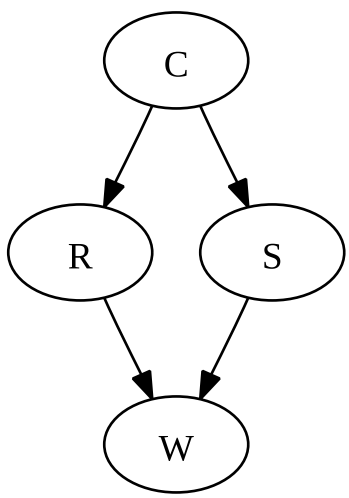

Figure 1: The cloud-sprinkler-rain-grass network.

The parameters for this BN are shown in tables 1–4.

Table 1:

$\to C$	$p(\neg c)$	$p(c)$
	0.5	0.5

Table 2:

$C \to S$	$p(\neg s)$	$p(s)$
$\neg c$	0.5	0.5
c	0.9	0.1

Table 3:

$C \to R$	$p(\neg r)$	$p(r)$
$\neg c$	0.8	0.2
c	0.2	0.8

Table 4:

$R, S \to W$		$p(\neg w)$	$p(w)$
$\neg r$	$\neg s$	1.00	0.00
$\neg r$	s	0.1	0.9
r	$\neg s$	0.1	0.9
r	s	0.01	0.99

4 Forward Propagation

Forward belief propagation in a BN is updating the probability distributions from parents to children, i.e. in the direction of the edges. This means that it only requires relatively straightforward applications of conditional probability calculations. Subsection 4.1 describes forward propagation in a classic BN, subsection 4.2 describes forward propagation in a FBN, and subsection 4.3 analyses the differences between classic and FBN forward propagation.

4.1 Forward Propagation in Classic Bayesian Networks

Assume we we are certain that the sky is cloudy (C is observed to be in the state c, perhaps we looked up). We want to know the posterior probability distribution for the grass, $p(W|c)$.

The effect of c on the updated conditional probabilities of R and S are just read off the respective tables (2 and 3):

- $S = \{f_{0.9}, t_{0.1}\}$
- $R = \{f_{0.2}, t_{0.8}\}$

Where we have used the notation described in subsection 2.2.

Given the updated uncertain beliefs about the states of R and S we can also calculate our updated belief about W. This is done by taking each conditional distribution in table 4, weighting it by the updated distributions over R and S and summing them.

Doing this, we find that:

$$\begin{aligned} p(W|c) &= 0.18 \times \{f_{1.00}, t_{0.00}\} + 0.72 \times \{f_{0.10}, t_{0.90}\} + \\ &\qquad 0.02 \times \{f_{0.10}, t_{0.90}\} + 0.08 \times \{f_{0.01}, t_{0.99}\} \\ &= \{f_{0.2548}, t_{0.7452}\} \end{aligned} \tag{3}$$

If we observed $\neg c$ then we would calculate $p(W|\neg c) = \{f_{0.451}, t_{0.549}\}$ in the same manner.

4.2 Forward Propagation in Fuzzy Bayesian Networks

Fuzzy BN propagation is described in [3] and readers are referred to that paper for another discussion of the details. In this subsection we illustrate an example of forward propagation which is analogous to that presented in subsection 4.1.

Assume $C = [f_{0.4}, t_{0.6}]$, where we continue to use the notation developed in [3]. Note that if $C = [t_1]$ or $C = [f_1]$ were observed then propagation in an FBN would give identical results to propagation in a classic BN in which c and $\neg c$, respectively, were observed. Given $C = [f_{0.4}, t_{0.6}]$, and by using component-wise propagation, we find that:

- $R = [\{f_{0.8}, t_{0.2}\}_{0.4}, \{f_{0.2}, t_{0.8}\}_{0.6}] \rightarrow [f_{0.44}, t_{0.56}]$
- $S = [\{f_{0.5}, t_{0.5}\}_{0.4}, \{f_{0.9}, t_{0.1}\}_{0.6}] \rightarrow [f_{0.74}, t_{0.26}]$

Where → denotes fuzzy integration. Using table 4 and the product t-norm (to find μ for each member of the Cartesian product of the parents), W is calculated in the same way:

$$\begin{aligned} W &\rightarrow 0.3256 \times \{f_1, t_0\} + 0.1144 \times \{f_{0.1}, t_{0.9}\} + \\ &\quad 0.4144 \times \{f_{0.1}, t_{0.9}\} + 0.1456 \times \{f_{0.01}, t_{0.99}\} \\ &\rightarrow [f_{0.3799}, t_{0.6201}] \end{aligned} \tag{4}$$

(5)

4.3 Comparing Classic and Fuzzy Forward Propagation

Forward propagation in a classic BN and a FBN are very similar. The key difference is that the state of a variable in a FBN often has multiple components. Each of these components is propagated independently of the others in the variable, and when a variable has multiple parents, each of which has multiple components, then the parents' components are combined using the Cartesian product set for calculation of the updated beliefs. The child's degree of membership in each member of the Cartesian product is calculated using the product t-norm.

5 Back Propagation

Through the application of Bayes's law, BN also support back propagation, from children to ancestors. How this is calculated is outlined in the following subsection, which presents an example of back propagation in a classic BN. Subsection 5.2 describes back propagation in a FBN and subsection 5.3 compares and contrasts classic and FBN back propagation.

5.1 Back Propagation in Classic Bayesian Networks

By applying Bayes's law, Bayesian networks also support back propagation, from children to ancestors. For example, assume we have observed that the grass is wet (w). What does this tell us about the sprinkler, the rain, and the cloudiness of the sky?

To calculate this we can use Bayes's Rule[3], which shows that:

[3]Or a message passing algorithm which also uses Bayes's Rule but which does not force us to calculate the full joint.

$$p(r|w) = \frac{p(r,w)}{p(w)} = \frac{p(w|r) \cdot p(r)}{p(w)} \tag{6}$$

And similarly for $p(\neg r|w)$, $p(s|w)$ and $p(\neg s|w)$.

Working with equation 6, we can find $p(r)$ and $p(w)$ by integrating over their respective ancestors. In subsection 4.1 we almost did this for w, by calculating $p(w|c)$ and $p(w|\neg c)$ instead. Thus we can complete the integration by summing over C now:

$$p(w) = 0.5 \times p(w|c) + 0.5 \times p(w|\neg c) = 0.6471 \tag{7}$$

By the same method, $p(R) = \{f_{0.5}, t_{0.5}\}$.

Finally, by summing over S in $p(w|S,r)$, where we need to use $p(S|r)$, we can calculate that $p(R|w) = \{f_{0.2921}, t_{0.7079}\}$. A similar set of calculations for S shows that $p(S|w) = \{f_{0.5702}, t_{0.4298}\}$.

By using the same method and carrying on our calculations, we find that:

$$p(C|w) = \{f_{0.4242}, t_{0.5758}\} \tag{8}$$

5.2 Back Propagation in Fuzzy Bayesian Networks

Similarly, as in a classic BN, we can also do back chaining in a FBN. Assume $W = [f_{0.1}, t_{0.9}]$. Using this we find:

- $R = [\{f_{0.8813}, t_{0.1187}\}_{0.1}, \{f_{0.2921}, t_{0.7079}\}_{0.9}] \rightarrow [f_{0.3510}, t_{0.6490}]$
- $S = [\{f_{0.9379}, t_{0.0621}\}_{0.1}, \{f_{0.5702}, t_{0.4298}\}_{0.9}] \rightarrow [f_{0.6070}, t_{0.3930}]$

Just as in FBN forward propagation, we use the product t-norm with the Cartesian product to find the expected state of C.

$$\begin{aligned} C \;\rightarrow\;& 0.2131 \times \{f_{0.6897}, t_{0.3103}\} + 0.1379 \times \{f_{0.9524}, t_{0.0476}\} + \\ & \quad 0.3939 \times \{f_{0.1220}, t_{0.8780}\} + 0.2551 \times \{f_{0.5556}, t_{0.4444}\} \\ \rightarrow\;& [f_{4681}, t_{5319}] \end{aligned} \tag{9}$$

5.3 Comparing Classic and Fuzzy Back Propagation

As in forward propagation, components within a variable are propagated independently, and the Cartesian product and product t-norm is used to combine different parents' components.

As is reasonable and expected, because the state in the classic BN (w) and the FBN ("0.9 w") were similar, the results are as well. However, their interpretation is somewhat different.

In the classic BN there is approximately a 42% chance that it was cloudy today, whatever it means for the weather to be cloudy. Does a single passing cloud on an otherwise sunny day make the day cloudy? Is it some proportion of the day? The FBN expected state is still vague, but less so. We would expect the cloudiness (whether that is measured by time or heaviness) to have been about half as much as it could be at most.

Through this analysis we begin to see how FBN may guide a more nuanced interpretation.

6 Explaining Away

Finally, we consider the phenomena of "explaining away", in which the posterior probability of a variable is affected because one of its co-parents was observed[11].

Although it is initially counter-intuitive (why is the probability of rain reduced when I learn that the sprinkler has been turned on?) careful thought and further analysis shows that the rain and sprinkler are indirectly related.

6.1 Explaining Away in Classic Bayesian Networks

For example, if we know that the grass is wet, but that it rained, then the posterior probability that the sprinkler was on is reduced. This is because the rain "explains away" the wetness of the grass:

$$p(S|r,w) = \{f_{0.8055}, t_{0.1945}\} \tag{10}$$

Thus, despite the grass completely soaked, we expect that the sprinkler was mostly off, rather than mostly on.

6.2 Explaining Away in Fuzzy Bayesian Networks

Finally, we present the phenomena of explaining away in a FBN. Assume we know that the grass is completely soaked ($W = [t_1]$) but that $R = [f_{0.25}, t_{0.75}]$. What is the expected state of the sprinkler?

Using the product t-norm on the Cartesian product of the parents and performing propagation for each pair of components, we find:

$$
\begin{aligned}
S &\rightarrow 0 \times \{f_{0.9325}, t_{0.0675}\} + 0.25 \times \{f_0, t_1\} + \\
&\quad\; 0 \times \{f_{0.9785}, t_{0.0215}\} + 0.75 \times \{f_{0.8055}, t_{0.1945}\} \\
&\rightarrow [f_{0.6041}, t_{0.3959}]
\end{aligned} \tag{11}
$$

6.3 Comparing Classic and Explaining Away

Because fuzzy states are not as absolute as discrete states, the magnitude of explaining away is somewhat reduced; however, we can see that it still occurs as we expect. Even though w was observed, the distribution over S was much more biased towards the discrete Boolean value $\neg s$ than s. This is because the state of R was more like r than $\neg r$.

7 Discussion

Having presented the bare numerical results of classic and FBN propagation it is now possible to describe and interpret them in more detail.

7.1 Interpretations

The first important factor to consider and which has been largely implicit and ignored in our calculations is a semantic definition of $C = [\alpha_f, (1-\alpha)_t]$, and likewise for the other variables.

There appear to be two intuitive semantic mappings of C. The first is that it represents the proportion of the day that was cloudy. In this case $C = [f_1, t_0]$ is a day with constantly and completely blue skies, and $C = [f_0, t_1]$ is a day that is constantly cloudy.

The second is that it represents the heaviness and storminess of the cloud, with $C = [f_1, t_0]$ representing entirely blue skies and $C = [f_0, t_1]$ representing raging thunderstorms, a cyclone, or similar right now, or over the past 24 hours, or since sunrise, or since today began, or similar (we should definite both dimensions carefully for our model to be most useful).

Similarly, R's state (W's state) could represent the heaviness of the rain (wetness of the grass) or the proportion of the day that it rained for (that the grass was wet for). These are all important design decisions that need to be considered explicitly but which are easy to forget with a Boolean or multinomial model.

One approach which is prompted by this analysis is as follows. Rather than having just one C variable, split it into two: C_P, which denotes the proportion of today which was cloudy, and C_H, which denotes the heaviness of the cloud. A possible structure for this model is shown in figure 2. Such a formulation is a much more nuanced representation of the system, and would not have been considered if not prompted by the FBN analysis.

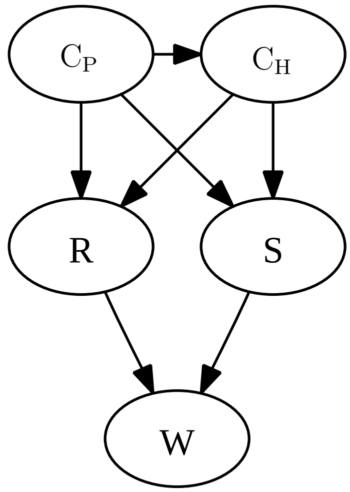

Figure 2: A revised cloud-sprinkler-rain-grass network, suggested by FBN-inspired analysis.

Whether or not this model is better or necessary depends on the application, but it is an advantage of FBN that it forces us to consider the meaning of our model in such detail.

7.2 Comparison

How do the results of classic and fuzzy propagation compare? Two important factors must be briefly noted.

7.2.1 Types of State

Firstly, the results of the FBN propagation are expected values. These are different from a probability distribution, the maximum likelihood state or the raw fuzzy state, made up of fuzzy probability distributions. As noted in [3], using the expected value does not bias further belief propagation. Further, the expected value can be meaningfully compared with observations when scoring and searching over models.

In contrast, the result of classic BN propagation is a probability distribution. This can be trivially converted into the expected value and offers the same advantages. However a probability distribution does not have any further advantages, unless the distribution itself is necessary.

Further, an equivalent distribution over a discrete variable can be constructed from a fuzzy expected value just by discarding the fuzzy information. For example, if $C = [f_{0.6179}, t_{0.3821}]$ then $C = \{f_{0.6179}, t_{0.3821}\}$ could be an appropriate discrete distribution over it.

Although the probabilistic and fuzzy states of C are superficially and notationally very similar it is important to note that they have different semantics. However, we can directly compare the expected value with observations, or the constructed distribution with other information, and so check that the results match our expectations.

7.2.2 Observations

In addition, the results are also not directly comparable because the observations when propagating over a classic BN and when propagating over a FBN were different. For example, in forward propagation over a classic BN, the state c (equivalently, $C = [1_t]$) was observed, but for the FBN the state $[0.4_f, 0.6_t]$ was observed. Had the fuzzy values been discrete, with complete and sole membership in just one value (true or false) then the results and necessary calculations would have been identical.

However, reviewing the results shows that they can all be easily and reasonably interpreted. Consider $S = [0.6041_f, 0.3959_t]$, the expected value of S, given that the grass is soaked and that it has rained a lot. This state suggests that we can expect the sprinkler to have been turned on roughly 40% of the time. Although this might not be what a person would have done it probably accurately reflects a weather-sensitive automatic system's behaviour. Further, in many situations this expected value is more informative for us than knowing that there was a 20% chance the sprinkler was on today (whatever it means for the sprinkler to be on) even though it rained (all day, was a thunder storm) and the grass is wet. Because there is no gradation or subtlety in discrete values their interpretation can be more difficult and requires more forethought and knowledge of the system's design.

7.3 Final Words

This paper has described the formalisation of FBN and touched on some of the conceptual foundations. It has illustrated belief propagation over a FBN in a range of situations; it has also analysed the differences in belief propagation and interpretation with classic BN. These differences suggest advantages that FBN may offer in certain situations and for doing certain types of model design.

References

[1] F. Bobillo and U. Straccia. A fuzzy description logic with product t-norm. In *Proceedings of the 16th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2007)*, pages 652–657, London (United Kingtom), July 2007.

[2] Christopher Fogelberg. Fuzziness, probability and fuzzy probability: A conceptual analysis. Current theoretical research., August 2008.

[3] Christopher Fogelberg, Vasile Palade, and Phil Assheton. Belief propagation in fuzzy Bayesian networks. In Ioannis Hatzilygeroudis, editor, *1st International Workshop on Combinations of Intelligent Methods and Applications(CIMA) at ECAI'08*, pages 19–24, University of Patras, Greece, 22 July 2008.

[4] Nir Friedman, Kevin Murphy, and Stuart Russell. Learning the structure of dynamic probabilistic networks. In *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, volume 14, pages 139–147, San Francisco, CA, 1998. Morgan Kaufmann. URL `citeseer.ist.psu.edu/friedman98learning.html`.

[5] A. J. Hartemink, D. K. Gifford, T. S. Jaakkola, and R. A. Young. Combining location and expression data for principled discovery of genetic regulatory network models. *Pacific Symposium on Biocomputing*, pages 437–449, 2002.

[6] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[7] David Heckerman. A tutorial on learning with Bayesian networks. Technical report, Microsoft Research, Redmond, Washington, 1995. URL `http://citeseer.ist.psu.edu/41127.html`.

[8] Xing-Chen Heng and Zheng Qin. FPBN: A new formalism for evaluating hybrid Bayesian networks using fuzzy sets and partial least-squares. In De-Shuang Huang, Xiao-Ping Zhang, and Guang-Bin Huang, editors, *ICIC (2)*, volume 3645 of *Lecture Notes in Computer Science*, pages 209–217, Hefei, China, August 23–26 2005. Springer. ISBN 3-540-28227-0.

[9] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003. URL `http://www.cambridge.org/0521642981`. Available from `http://www.inference.phy.cam.ac.uk/mackay/itila/`.

[10] Heping Pan and Lin Liu. Fuzzy Bayesian networks - a general formalism for representation, inference and learning with hybrid Bayesian networks. *IJPRAI*, 14(7):941–962, 2000.

[11] Michael P. Wellman and Max Henrion. Explaining "explaining away". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15 (3):287–292, March 1993. URL `http://ai.eecs.umich.edu/people/wellman/pubs/pami93.pdf`.

[12] Jing Yu, V. Anne Smith, Paul P. Wang, Alexander J. Hartemink, and Erich D. Jarvis. Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, 20 (18):3594–3603, 2004.

[13] Lotfi A. Zadeh. Generalized theory of uncertainty (GTU) — principal concepts and ideas. *Computational Statistics & Data Analysis*, 51(1): 15–46, 2006.

[14] Yu Zhang, Zhingdong Deng, Hongshan Jiang, and Peifa Jia. Dynamic Bayesian network (DBN) with structure expectation maximization (SEM) for modeling of gene network from time series gene expression data. In Hamid R. Arabnia and Homayoun Valafar, editors, *BIOCOMP*, pages 41–47. CSREA Press, 2006. ISBN 1-60132-002-7.