Modeling Causal Reinforcement and Undermining for Efficient CPT Elicitation

Yang Xiang and Ning Jia

***Abstract*— Representation of uncertain knowledge using a Bayesian network requires acquisition of a conditional probability table (CPT) for each variable. The CPT can be acquired by data mining or elicitation. When data are insufficient to support mining, causal modeling, such as the noisy-OR, aids elicitation by reducing the number of probability parameters to be acquired from human experts. Multiple causes can reinforce each other in producing the effect or can undermine the impact of each other. Most existing causal models do not consider causal interactions from the perspective of reinforcement or undermining. Our analysis shows that none can represent both interactions. Except the RNOR, other models also limit parameters to probabilities of single-cause events. We present the first general causal model, the *non-impeding noisy-AND tree*, that allows encoding of both reinforcement and undermining. It supports efficient CPT acquisition by elicitating a partial ordering of causes in terms of a tree topology, plus necessary numerical parameters. It also allows incorporation of probabilities for multi-cause events.**

Keywords: knowledge engineering, probabilistic reasoning, uncertainty, knowledge acquisition, knowledge modeling, elicitation methods.

I. Introduction

A Bayesian network (BN) [10] encodes probabilistic knowledge about a problem domain through a dependence structure in the form of a directed acyclic graph and CPTs associated with nodes of the graph. To construct a BN for a given domain, these CPTs must be acquired. The complexity is linear on the number of variables but exponential on the maximum number of parents of a variable. When a variable has many parents, direct acquisition of its CPT is costly. The CPT may be acquired through data mining. However, a given problem domain may have insufficient amount of data to support mining, but has human experts for elicitation. In such a case, how to elicitate the CPT efficiently becomes a practical need in knowledge engineering.

The most widely used model to support such elicitation is the noisy-OR model pioneered by Good [3] and further studied by Pearl [10]. The model assumes that a number of binary causes can produce an effect. Their interaction is expressed by a logic OR gate. Each cause may fail to produce the effect and this uncertainty is represented by a probabilistic inhibitor conjuncted with the cause. The noisy-OR gate encodes the assumption that causes fail independently, from which the number of probability parameters to be assessed per CPT is reduced to linear on the number of parent variables.

Henrion [7] added to the noisy-OR model a leaky probability, extended the model from binary to multi-valued variables, and introduced the noisy-MAX model. A leaky probability captures the probability of occurrence of the effect when all explicitly represented causes are absent. Diez [1] and Srinivas [15] also studied generalization of the noisy-OR model. Diez [1] introduced the noisy-MIN models. Heckerman discussed the noisy-ADD model [5][1]. Heckerman and Breese [6] analyzed a collection of causal independence relations that allows efficient acquisition of CPTs. In particular, they considered amechanistic, decomposable, multiply decomposable, and temporal relations, which are generalizations of models such as the noisy-OR. Galan and Diez [2] introduced the noisy-AND model. Recently, Lemmer and Gossink [9] proposed the recursive noisy-OR (RNOR) model. Instead of allowing only probability parameters of the effect given each single-cause as the input, the RNOR model allows probability parameters of the effect given subsets of causes. A method based on the RNOR, tailored to an application, was used by Kuter et al. [8] to handle inhibition.

When multiple causes are present, they may reinforce each other. That is, the effect is more likely to occur when more causes are active. Alternatively, multiple causes may undermine the impact of each other. That is, the effect becomes less likely when more causes are present. Unlike the RNOR [9], previous works do not consider causal interactions among variables from the perspective of reinforcement or undermining. Our analysis shows that previously proposed causal models, including the noisy-OR, the noisy-MAX, the noisy-AND, the noisy-MIN, the noisy-ADD and the RNOR, are limited to represent only one type of causal interaction, and cannot express both. Furthermore, except the RNOR, other models limit input parameters to probabilities of single-cause events.

In this work, we present a new causal model, termed the non-impeding noisy-AND tree, or simply NIN-AND tree, that can represent both types of causal interactions among a set of causes, some of which are reinforcing and others are undermining. Like the RNOR, probabilities for multi-cause events can be incorporated as model parameters if so desired. The NIN-AND tree degenerates to the noisy-OR and the noisy-MAX in the binary case, when its topology and input events are restricted, but is more expressive than each in general.

In Section II, we introduce the terminology and define formally reinforcement and undermining. Section III presents building blocks for modeling reinforcement and undermining. Section IV analyzes limitations of alternative causal models in representing these causal interactions. Section V proposes the NIN-AND tree model. How to use it to obtain causal probabilities is described in Section VI. We present, in Section VII, how to use the NIN-AND tree when default assumptions do not hold. We analyze the complexity of elicitation in Section VIII and demonstrate elicitation of CPTs with the NIN-AND tree in Section IX.

Y. Xiang is with the University of Guelph. N. Jia is with Desire2Learn Inc.

[1] It was referred to as the noisy-addition there. We will use the name noisy-ADD to be consistent with the naming of other alternatives.

II. Background

A. Uncertain Causes and Causal Events

We aim to acquire efficiently the CPT of a variable x conditioned on a set of variables Y based on their causal relation. The causes that we consider are uncertain causes. Following Lemmer and Gossink [9], an *uncertain cause* is a cause that can produce an effect but does not always do so. We denote a set of binary cause variables as $X = \{c_1, ..., c_n\}$ and their effect variable (binary) as e. For each c_i, we denote $c_i = true$ by c_i^+ and $c_i = false$ by c_i^-. Similarly, we denote $e = true$ by e^+.

In this work, we assume that the causal relation between a cause and an effect has been ascertained. We investigate how to model such relations to support elicitation of probability parameters. For how to ascertain the causal relation, see, for example, Spirtes, Glymour and Scheines [13], [14], Shafer [12], and Pearl [11].

We refer to the event that a cause c_i caused an effect e to occur successfully as a *causal event*. We denote this causal event by $e^+ \leftarrow \{c_i^+\}$ or simply $e^+ \leftarrow c_i^+$. The event is uncertain since e may be false when c_i is true. Its probability is denoted $P(e^+ \leftarrow c_i^+)$ where $0 < P(e^+ \leftarrow c_i^+) < 1$. The negation of the causal event, that c_i failed to cause e, is denoted as $e^+ \not\leftarrow c_i^+$. The probability of the *causal failure event* is $P(e^+ \not\leftarrow c_i^+) = 1 - P(e^+ \leftarrow c_i^+)$.

We denote the causal event that a set $X = \{c_1, ..., c_n\}$ of causes caused e by $e^+ \leftarrow \{c_1^+, ..., c_n^+\}$, or simply $e^+ \leftarrow c_1^+, ..., c_n^+$, or the vector notation $e^+ \leftarrow \underline{x}^+$. When the cause set is indexed, such as $W_i = \{c_1, ..., c_n\}$, the causal event can be denoted as $e^+ \leftarrow \underline{w}_i^+$. We allow broad interpretations of a causal event by a set of causes, as will be seen in later sections. For instance, we are not limited to the interpretation in [9]: *the effect is caused by at least one of the causes*. The probability of the event $e^+ \leftarrow \underline{w}_i^+$ is $P(e^+ \leftarrow \underline{w}_i^+) = P(e^+ \leftarrow c_1^+, ..., c_n^+)$.

Pearl [10] regards a cause as an event whose occurrence always results in an effect, unless it is blocked by an inhibitor. He encodes the causal uncertainty through the uncertain inhibitor. The conjunction of a certain cause and an inhibitor in his formulation is equivalent to an uncertain cause.

When modeling a domain with a BN, the set of all causes of an effect variable e is its parent variables. Denote the set of *all causes* of e by C. To capture causes that we do not wish to represent explicitly, we include a leaky variable in C.

Probabilities of causal events can be used to acquire the CPT $P(e|C)$. For example, if $C = \{c_1, c_2, c_3, c_4\}$, then we have $P(e^+|c_1^+, c_2^+, c_3^+, c_4^+) = P(e^+ \leftarrow c_1^+, c_2^+, c_3^+, c_4^+)$ and $P(e^+|c_1^+, c_2^-, c_3^+, c_4^+) = P(e^+ \leftarrow c_1^+, c_3^+, c_4^+)$. Note that only cause variables of value *true* are included in the causal probability. Under the leaky variable assumption, we have $P(e^+|c_1^-, c_2^-, c_3^-, c_4^-) = P(e^+ \leftarrow \emptyset) = 0$. Note $P(e^+ \leftarrow \emptyset) \neq P(e^+)$ in general.

B. Reinforcement and Undermining

When multiple causes are present, they may reinforce each other in producing the effect. That is, their combined influence is greater than that from only some of them. Alternatively, multiple causes may undermine each other in producing the effect. Below, we define reinforcement and undermining formally.

Definition 1: Let $R = \{W_1, W_2, ...\}$ be a partition of a set X of causes, R' be a proper subset of R, and Y be the union of elements in R'. Sets of causes in R are said to reinforce each other, if for every subset $R' \subset R$, it holds that

$$P(e^+ \leftarrow \underline{y}^+) \leq P(e^+ \leftarrow \underline{x}^+).$$

Sets of causes in R are said to undermine each other, if for every subset $R' \subset R$, it holds that

$$P(e^+ \leftarrow \underline{y}^+) > P(e^+ \leftarrow \underline{x}^+).$$

Intuitively, sets of causes $W_1, W_2, ...$ reinforce each other if collectively they are at least as effective in causing the effect as some acting by themselves. If collectively they are less effective, then they must be undermining each other.

Lemmer and Gossink [9] classify causal interactions into *positive causality* versus *inhibition*. The dividing line between positive causality and inhibition is drawn when causes collectively are more likely to produce the effect than any proper subset. They further classify positive causality into three subclasses: *synergy*, *noisy-OR*, and *interference*. Synergy and interference are defined according to whether causes collectively are more likely or less likely to produce the effect than in a noisy-OR model. We interpretate the relation between the two classifications in Fig. 1, where the shaded area represents positive causality and the white area corresponds to inhibition.

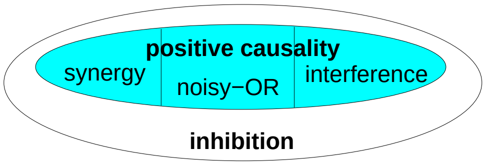


Fig. 1. Relation between two classifications of causal interactions. Each point corresponds to a pattern of causal interaction between an effect and a set of causes.

We believe that collection being stronger than parts in causal strength is a more fundamental dividing line (used by the positive causality versus inhibition classification). In comparison, the difference between synergy and interference is less fundamental. After all, they all satisfy the property of collection being stronger than parts and only differ in the degree of strength.

The classification of reinforcement versus undermining defined above is consistent with the positive causality versus inhibition classification, but is more general. In Definition 1, when each W_i is a singleton, reinforcement becomes positive causality and undermining becomes inhibition. In other words, the positive causality versus inhibition classification is a special case of the reinforcement versus undermining classification. The generality of reinforcement and undermining allows modeling of reinforcement of sets of causes when the causes in some set are undermining. It also allows modeling of undermining of sets of causes when the causes in some set are reinforcing. Such modeling expressiveness is not possible under the positive causality versus inhibition definition. This will become clear in Section V.

III. Building Blocks for Reinforcement and Undermining

In this section, we present the building blocks for modeling reinforcement and those for modeling undermining. For each type of causal interaction, we propose default assumptions and encode these assumptions graphically by a noisy logic gate. We then show that these assumptions lead to the intended causal interaction (i.e., reinforcing or undermining), thus establishing the noisy gate as the building block of the causal interaction.

First, we consider reinforcing interactions. We assume that these causes satisfy failure conjunction and failure independence.

Definition 2: Sets of causes $W_1, ..., W_m$, where W_i and W_j are disjoint for distinct i and j, satisfy `failure conjunction` if the following equation holds,

$$(e^+ \not\leftarrow \underline{w}_1^+, ..., \underline{w}_m^+) = (e^+ \not\leftarrow \underline{w}_1^+) \wedge ... \wedge (e^+ \not\leftarrow \underline{w}_m^+). \quad (1)$$

Intuitively, these sets of causes collectively fail to produce the effect when each set of causes has failed to produce the effect.

Definition 3: Sets of causes $W_1, ..., W_m$, where W_i and W_j are disjoint for distinct i and j, satisfy `failure independence` if failure events $e^+ \not\leftarrow \underline{w}_1^+$, ..., $e^+ \not\leftarrow \underline{w}_m^+$ are independent of each other. That is, the following equation holds,

$$\begin{aligned} &P((e^+ \not\leftarrow \underline{w}_1^+) \wedge ... \wedge (e^+ \not\leftarrow \underline{w}_m^+)) \\ &= P(e^+ \not\leftarrow \underline{w}_1^+) \ ... \ P(e^+ \not\leftarrow \underline{w}_m^+). \quad (2) \end{aligned}$$

To graphically model causal interactions that satisfy the above conditions, we introduce a non-impeding noisy-AND gate, or NIN-AND gate. Its inputs are either all causal events, e.g., $e^+ \leftarrow c_i^+, c_j^+$, or all causal failure events, e.g., $e^+ \not\leftarrow c_j^+, c_k^+$. These events are uncertain and hence the name noisy. Its output event is the conjunction (AND) of the input events. The output event is independent of whether the input events involve all causes of the effect e (non-impeding). This non-impeding property is in contrast with the noisy-AND model [2], which we will elaborate in Section IV-C. We refer to NIN-AND gates whose inputs are causal events as *direct* NIN-AND gates and those whose inputs are causal failure events as *dual* NIN-AND gates. As a convention, we require each NIN-AND gate to have at least two input events.

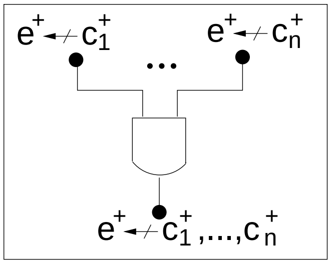

$e^+ \not\leftarrow c_1^+$
...
$e^+ \not\leftarrow c_n^+$
$e^+ \not\leftarrow c_1^+, ..., c_n^+$

Fig. 2. A dual NIN-AND gate for reinforcement.

We model causal interactions that satisfy failure conjunction and failure independence graphically with a dual NIN-AND gate. Fig. 2 illustrates such a gate, where each $W_i = \{c_i\}$ is a singleton and $m = n$. Note that failure conjunction is expressed by the AND gate and failure independence is expressed by the lack of direct connection between individual failure events. Note that probabilities in Eqn (2) are associated with the input or output events but these probabilities are not themselves the input or output of the gate. The following proposition establishes that a dual NIN-AND gate models reinforcement.

Proposition 1: Let $R = \{W_1, W_2, ...\}$ be a partition of a set X of uncertain causes of an effect e and sets in R satisfy Eqns (1) and (2). Then, interaction among sets of causes in R is reinforcing.

Proof:

Partition R into $\{U, V\}$, where $U \subset R$ and $V = R \setminus U$. From Eqns (1) and (2), we have $P(e^+ \not\leftarrow \underline{x}^+) = P(e^+ \not\leftarrow \underline{u}^+)P(e^+ \not\leftarrow \underline{v}^+)$. Because each cause c_i in X is an uncertain cause and $0 < P(e^+ \not\leftarrow c_i^+) < 1$, we have $0 < P(e^+ \not\leftarrow \underline{u}^+) < 1$ and $0 < P(e^+ \not\leftarrow \underline{v}^+) < 1$. Hence, $P(e^+ \not\leftarrow \underline{x}^+) < P(e^+ \not\leftarrow \underline{u}^+)$, which implies $P(e^+ \leftarrow \underline{x}^+) > P(e^+ \leftarrow \underline{u}^+)$. $\square$

Unlike the graphical model in [10] which represents success events directly, the dual NIN-AND gate represents failure events, which corresponds to failure independence more directly. Furthermore, the following shows that noisy-OR gates are special cases of dual NIN-AND gates, in the same sense that positive causality is a special case of reinforcement.

Proposition 2: A noisy-OR gate is equivalent to a dual NIN-AND gate with only single-cause input events.

Proof:

The proposition follows from the equivalent expression of Eqn (1) as follows:

$$(e^+ \leftarrow \underline{w}_1^+, ..., \underline{w}_m^+) = (e^+ \leftarrow \underline{w}_1^+) \vee ... \vee (e^+ \leftarrow \underline{w}_m^+).$$

Alternatively, when each set W_i of causes is a singleton, Eqn (2) can be written as

$$P(e^+ \leftarrow c_1^+, ..., c_n^+) = 1 - \prod_{i=1}^{n} (1 - P(e^+ \leftarrow c_i^+)), \quad (3)$$

which characterizes the noisy-OR gate (see [10]). $\square$

We refer to the dual NIN-AND gate in Fig. 2 as the default model for reinforcement. The default model represents only one pattern of reinforcement among sets of causes. We consider other patterns of reinforcement in Section VII.

Next, we consider undermining interactions. As this is the causal interaction less studied, we motivate with an example in family relation: *A man who lives with his wife only is likely happy, and so is with his mother only. When he lives with both, he is likely miserable.* A recent web article by Elizabeth Graham [4] reported that out of 17 women in a study group, only two had a good in-law relationship. The focus of the article was not on the man in the middle, but it is not difficult to realize the stress that the man is under, trying to mediate the relationship. Here, happiness of the man is the effect and living alone (such as being an orphan) is assumed unhappy. Activating a single-cause, living with one of the women, increases the chance of being happy. The probability is reduced when both causes are active, compared to the case of only one. Formally, we assume that these causes satisfy success conjunction and success independence.

Definition 4: Sets of causes $W_1, ..., W_m$, where W_i and W_j are disjoint for distinct i and j, satisfy `success conjunction` if the following relation holds,

$$e^+ \leftarrow \underline{w}_1^+, ..., \underline{w}_m^+ = (e^+ \leftarrow \underline{w}_1^+) \wedge ... \wedge (e^+ \leftarrow \underline{w}_m^+). \quad (4)$$

This assumption states that a successful multi-cause event requires each cause to be effective. Because it is not immediately intuitive, we come back to elaborate after Proposition 3 below.

Definition 5: Sets of causes $W_1, ..., W_m$, where W_i and W_j are disjoint for distinct i and j, `succeed independently` if success events $e^+ \leftarrow \underline{w}_1^+$, ..., $e^+ \leftarrow \underline{w}_m^+$ are independent of each other. That is, the following relation holds,

$$\begin{aligned} &P((e^+ \leftarrow \underline{w}_1^+) \wedge ... \wedge (e^+ \leftarrow \underline{w}_m^+)) \\ &= P(e^+ \leftarrow \underline{w}_1^+) \ ... \ P(e^+ \leftarrow \underline{w}_m^+). \quad (5) \end{aligned}$$

We model causes that satisfy success conjunction and success independence graphically with a direct NIN-AND gate. Fig. 3 illustrates such a gate, where each $W_i = \{c_i\}$ is a singleton and

$m = n$. Note that success conjunction is expressed by the AND gate and success independence is expressed by the lack of direct connection between individual success events.

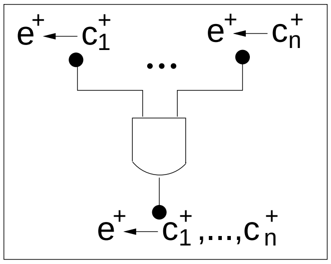


Fig. 3. A direct NIN-AND gate for undermining.

The following proposition establishes that a direct NIN-AND gate models the undermining interaction, whose proof is straightforward.

Proposition 3: Let $R = \{W_1, W_2, ...\}$ be a partition of a set X of uncertain causes of effect e and sets in R satisfy Eqns (4) and (5). Then, interaction among sets of causes in R is undermining.

Again, the direct NIN-AND gate in Fig. 3 is the default model for undermining and represents only one pattern of undermining among sets of causes. We consider representation of other patterns of undermining in Section VII.

Having established that causes that satisfy success conjunction and success independence are undermining each other, we elaborate the intuition behind. Success conjunction says that when sets of causes succeed in causing the effect in an undermining way, each set of causes must have been effective. If any set of causes has occurred but has failed to be effective, it would not undermine the other sets of causes. In the in-law example, wife has her way to make husband happy (her way to manage the family matter) and mother often has a different way. When both are trying to impact how the family matter is handled (for instance, how rooms should be decorated, how kids should be educated, etc.), idea of neither can be implemented smoothly and completely, and the man has to resolve the conflict, which reduces the chance of his being happy. As for success independence, given that living with mother only has made the man happy, it does not change the tendency that the man can be happy after marrying his wife and living separately from his mother. An additional example on undermining is given in Section V.

IV. Limitations of Alternative Causal Models

Before presenting the new causal model for reinforcement and undermining, we analyze alternative causal models and reveal their limitations in modeling the two types of causal interactions. As we have defined reinforcement and undermining under the binary context, the following analysis is restricted to such a context as appropriate.

A. The Noisy-OR Model

First, we analyze the noisy-OR model [3], [10]. Proposition 1 shows that when uncertain causes satisfies Eqns (1) and (2), their interaction is reinforcing. Proposition 2 shows that a noisy-OR gate is equivalent to a dual NIN-AND gate of single-cause input events, since they both satisfy Eqns (1) and (2). Therefore, Definition 1, Propositions 1 and 2 collectively imply that the noisy-OR model can represent only reinforcement but not undermining.

B. The Noisy-MAX Model

In the binary case, with two inputs and with uncertainty ignored, the output of a noisy-MAX model [7] is true if at least one of the two inputs is true, and is false if both inputs are false. Hence, the noisy-MAX model behaves the same as the noisy-OR model when variables are binary. From the conclusion drawn in Section IV-A on the noisy-OR, when the domain is binary, the noisy-MAX model represents only reinforcing interactions.

C. The Noisy-AND Model

The behavior of a noisy-AND model [2] is characterized into two cases. In the first case, the set C of *all causes* are true. In the second case, some causes are false.

$$P(e^+ \leftarrow c_1^+, ..., c_n^+) =$$
$$\begin{cases} P(e^+ \leftarrow c_1^+) \cdots P(e^+ \leftarrow c_n^+) & : \{c_1, ...c_n\} = C \\ 0 & : \{c_1, ...c_n\} \subset C \end{cases}$$

From Definition 1, the noisy-AND model represents only reinforcing interactions.

It can be seen that the behavior of the noisy-AND in the first case is consistent with Eqns (4) and (5). However, absence of any cause in C obstructs the causal influence of the remaining causes, forces the causal probability to zero, and prevents Eqns (4) and (5) from being applicable to the second case. We refer to this as the *impeding* behavior of the noisy-AND. On the other hand, the direct NIN-AND gate in Fig. 3 allows Eqns (4) and (5) to be extended to the second case, hence, the name *non-impeding* noisy-AND. This difference between impeding and non-impeding is significant. As the result, the noisy-AND represents reinforcement while the direct NIN-AND represents undermining.

D. The Noisy-MIN Model

In the binary case, with two inputs and with uncertainty ignored, the output of a noisy-MIN model [1] is false if at least one of the two inputs is false, and is true if both inputs are true. Hence, the binary noisy-MIN model behaves the same as the noisy-AND model. According to the above analysis, when the domain is binary, the noisy-MIN model represents only reinforcing interactions.

E. The Recursive Noisy-OR Model

Lemmer and Gossink [9] proposed the RNOR model. To acquire the effect probability due to a set of causes, the RNOR model can combine causal probabilities due to subsets of causes, where each subset can contain multiple causes. According to the RNOR, for a set of causes $X = \{c_0, ..., c_{n-1}\}$, if $P(e^+ \leftarrow c_0^+, ..., c_{n-1}^+)$ is not directly assessed by the expert, it is evaluated as

$$P(e^+ \leftarrow c_0^+, ..., c_{n-1}^+) =$$
$$1 - \prod_{i=0}^{n-1} \frac{1 - P(e^+ \leftarrow \underline{x}^+ \setminus c_i^+)}{1 - P(e^+ \leftarrow \underline{x}^+ \setminus \{c_i^+, c_{(i+1)\%n}^+\})}. \quad (6)$$

Because Eqn (6) is derived from a rewriting of Eqn (3) observed by the noisy-OR model, the RNOR inherits the assumptions of the noisy-OR. Therefore, according to the analysis of the noisy-OR model in Section IV-A, the RNOR model can only represent reinforcement, as acknowledged in [9]. As shown by Lemmer and Gossink, Eqn (6) produces a numerical value in the range $[0, 1]$ as long as causes in X are reinforcing. However, if they are undermining, the result from the equation may not be a valid probability (the numerical value may be outside the range $[0, 1]$).

F. The Noisy-ADD Model

The noisy-ADD [5] can only represent reinforcement. Consider a noisy-adder with two binary causes c_1 and c_2 whose domains are $\{0, 1\}$. It has the following DAG model

$$c_1 \longrightarrow i_1 \longrightarrow e \longleftarrow i_2 \longleftarrow c_2$$

where i_1 and i_2 are intermediate variables and the effect $e = i_1 + i_2 \in \{0, 1, 2\}$. The model assumes $P(i_j = 0|c_j = 0) = 1$ and $0 < P(i_j = 1|c_j = 1) < 1$ for $j = 1, 2$. For simplicity, we assume $P(i_1 = 1|c_1 = 1) = P(i_2 = 1|c_2 = 1)$ and denote their value by q. Note that

$$P(e = 1|c_1 = 1, c_2 = 0) = q \quad and \quad P(e = 2|c_1 = 1, c_2 = 0) = 0.$$

To assess expressiveness of the model, we first consider the effect $e = 1$ and then consider the disjunction $e = 1$ or $e = 2$. Denote $P(e = 1|c_1 = 1, c_2 = 1)$ by r and first we compare r with q. A simple derivation shows $r = 2q(1 - q)$. If $q < 0.5$, then $r > q$. If $q > 0.5$, then $r < q$. By Definition 1, if a causal model is reinforcing, then no matter what value $P(e^+ \leftarrow \underline{y}^+)$ is, the relation $P(e^+ \leftarrow \underline{y}^+) \leq P(e^+ \leftarrow \underline{x}^+)$ must hold and its negation must hold for undermining. Being unable to maintain the inequality across the entire range of values for $P(e^+ \leftarrow \underline{y}^+)$ implies that the noisy-ADD is unable to represent either reinforcement or undermining if we focus on effect $e = 1$.

Next, we focus on the disjunctive effect $e \in \{1, 2\}$, i.e., $e \neq 0$. Denote $P(e \neq 0|c_1 = 1, c_2 = 1)$ by t and it can be shown that $t = q(2 - q)$. Since $P(e \neq 0|c_1 = 1, c_2 = 0) = P(e = 1|c_1 = 1, c_2 = 0) = q$, we compare t with q. From $t = q(2 - q) > q$ for $0 < q < 1$, we conclude that noisy-ADD can only represent reinforcement if we focus on disjunctive effect $e \in \{1, 2\}$.

G. Other Models

For models considered in [6], the amechanistic model has essentially a star topology and other three models (decomposable, multiply decomposable and temporal) are essentially binary trees. When the binary tree is instantiated according to the noisy-OR, the noisy-AND, the noisy-MAX, the noisy-MIN, or the noisy-ADD, it inherits limitations of these models as discussed above. In these models, each root node must also be a single-cause variable and multi-cause event probability parameters are not considered.

H. Summary

TABLE I

SUMMARY OF PROPERTIES OF ALTERNATIVE CAUSAL MODELS.

Model	Reinforcement	Undermining	Multi-cause input
noisy-OR	yes	no	no
noisy-AND	yes	no	no
noisy-MAX	yes	no	no
noisy-MIN	yes	no	no
noisy-ADD	yes	no	no
RNOR	yes	no	yes
NIN-AND tree	yes	yes	yes

To summarize, Table I compares the properties of alternative causal models with the NIN-AND tree model to be presented and analyzed below. Note that none of the alternative models can incorporate probability parameters for multi-cause events, except the RNOR and the NIN-AND tree.

Note also that although the noisy-MAX and the noisy-MIN are applicable to multi-valued variables, their inability to model both reinforcement and undermining in the binary case implies that their inability in general. Therefore, our analysis of these two models based on binary variables is conclusive in general.

V. THE NIN-AND TREE

A. A Motivating Example

Consider two sets X and Y of causes that reinforce each other. It is possible that causes within X undermine each other, and so do causes within Y. In general, such interplay of causal interactions of different natures can form a hierarchy. In this section, we present a graphical representation to model such a hierarchy. It is based on the direct and dual NIN-AND gates and has a tree topology. We term it the *non-impeding noisy-AND tree* or NIN-AND tree. We assume that a human expert assesses reinforcing and undermining interactions among causes according to some partial order and is able to articulate the hierarchy.

For example, consider a patient who is in the process to recover from a disease D. Taking medicine M helps recovery and so does regular exercise. The patient's normal diet contains minerals that facilitate recovery but taking with medicine M reduces the effectiveness of both. The causes and effect involved are as follows:

- e^+ : Recovery from disease D within a particular time period.
- c_1^+: Taking medicine M.
- c_2^+: Regular exercise.
- c_3^+: Patient's taking his/her normal diet.

For the purpose of prognosis, one needs to acquire $P(e^+|c_1^+, c_2^+, c_3^+) = P(e^+ \leftarrow c_1^+, c_2^+, c_3^+)$. To ease the task, a physician may consider first the undermining interaction between c_1 and c_3. (S)he then considers the reinforcing interaction between sets $\{c_1, c_3\}$ and $\{c_2\}$. Thus, the physician has articulated an order for stepwise evaluation. The physician also assesses

$$P(e^+ \leftarrow c_1^+) = 0.85, P(e^+ \leftarrow c_2^+) = 0.8, P(e^+ \leftarrow c_3^+) = 0.7.$$

If this is all the information that the physician can provide, the causal interaction can be modeled as the NIN-AND tree in Fig. 4.

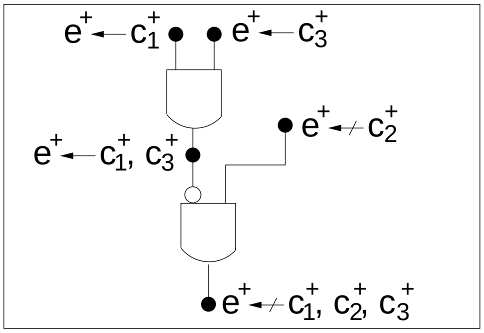

$e^+ \leftarrow c_1^+$
$e^+ \leftarrow c_3^+$
$e^+ \not\leftarrow c_2^+$
$e^+ \leftarrow c_1^+, c_3^+$
$e^+ \not\leftarrow c_1^+, c_2^+, c_3^+$

Fig. 4. The NIN-AND tree model for the recovery example.

From the upper direct NIN-AND gate and Eqn (5), $P(e^+ \leftarrow c_1^+, c_3^+) = 0.595$ is derived: a result of undermining. Its output is negated (shown by the white oval) before entering the lower dual NIN-AND gate and the corresponding event has probability $P(e^+ \not\leftarrow c_1^+, c_3^+) = 0.405$. From the lower dual NIN-AND gate and Eqn (2), the following are derived:

$$P(e^+ \not\leftarrow c_1^+, c_2^+, c_3^+) = P(e^+ \not\leftarrow c_1^+, c_3^+) P(e^+ \not\leftarrow c_2^+) = 0.081,$$

and $P(e^+ \leftarrow c_1^+, c_2^+, c_3^+) = 0.919$. It is a result of reinforcement. The CPT obtained is shown below, where $P(e^+|c_1^-, c_2^-, c_3^-)$ is obtained based on the leaky cause assumption, $P(e^+|c_1^-, c_2^-, c_3^+)$ by direct elicitation, $P(e^+|c_1^-, c_2^+, c_3^+)$ through a dual NIN-AND gate, and $P(e^+|c_1^+, c_2^-, c_3^+)$ through a direct NIN-AND gate.

$$\begin{array}{rcr}
P(e^+|c_1^-, c_2^-, c_3^-) & = & 0 \\
P(e^+|c_1^-, c_2^-, c_3^+) & = & 0.7 \\
P(e^+|c_1^-, c_2^+, c_3^-) & = & 0.8 \\
P(e^+|c_1^-, c_2^+, c_3^+) & = & 0.94 \\
P(e^+|c_1^+, c_2^-, c_3^-) & = & 0.85 \\
P(e^+|c_1^+, c_2^-, c_3^+) & = & 0.595 \\
P(e^+|c_1^+, c_2^+, c_3^-) & = & 0.97 \\
P(e^+|c_1^+, c_2^+, c_3^+) & = & 0.919
\end{array}$$

B. The Definition

The following defines the NIN-AND tree formally. In the definition, we have extended the set operators ∪ and ∩ to vectors according to the obvious interpretation.

Definition 6: Let e be an effect and $X = \{c_1, ..., c_n\}$ be a set of uncertain causes that is known to have occurred. A `NIN-AND tree` for modeling the causal interaction among elements of X is a directed tree where the following holds:

1) There are two types of nodes on the tree. An `event` node is shown as a black oval and a `gate` node is shown as a NIN-AND gate. Each event node has at most one incoming link and at most one outgoing link. Each gate has at least two incoming links and exactly one outgoing link.
2) Every link connects an event node with a gate node. There are two types of links: `forward` links and `negation` links. Each link is directed from its `tail` node to its `head` node consistently along the input-to-output stream of gates. A forward link is shown as a line and is implicitly directed. A negation link is shown as a line with a white oval at the head and is explicitly directed.
3) All terminal nodes are event nodes and each is labeled by a causal event in the form $e^+ \leftarrow \underline{y}^+$ or $e^+ \not\leftarrow \underline{y}^+$. Exactly one terminal node, called the `leaf`, is connected to the output of a gate and has $\underline{y}^+ = \underline{x}^+$. Each other terminal node is connected to the input of a gate and is a `root`. For each root, $\underline{y}^+$ is a proper subset of $\underline{x}^+$. With i indexing roots, it holds that $\bigcup_i \underline{y}_i^+ = \underline{x}^+$. For every two roots labeled by $\underline{y}_j^+$ and $\underline{y}_k^+$, it holds that $\underline{y}_j^+ \cap \underline{y}_k^+ = \emptyset$.
4) Multiple inputs of a gate g must be in one of the following cases:
   a) Each is either connected by a forward link to a node labeled with $e^+ \leftarrow \underline{y}^+$, or by a negation link to a node labeled with $e^+ \not\leftarrow \underline{y}^+$. The output of g is connected by a forward link to a node labeled with $e^+ \leftarrow \cup_i \underline{y}_i^+$.
   b) Each is either connected by a forward link to a node labeled with $e^+ \not\leftarrow \underline{y}^+$, or by a negation link to a node labeled with $e^+ \leftarrow \underline{y}^+$. The output of g is connected by a forward link to a node labeled with $e^+ \not\leftarrow \cup_i \underline{y}_i^+$.

Degree restriction in Condition 1 ensures that an event represents the output of no more than one gate and is connected to the input of no more than one gate. Condition 4 (a) corresponds to a direct NIN-AND gate (e.g., the upper gate in Fig. 4). Condition 4 (b) corresponds to a dual NIN-AND gate (e.g., the lower gate in Fig. 4). Semantically, 4 (a) corresponds to sets of undermining causes and 4 (b) corresponds to sets of reinforcing causes.

The NIN-AND tree model is more general than the noisy-OR (as well as the binary noisy-MAX) and has the latter as its special case. For instance, Fig. 2 corresponds to the noisy-OR and is a special case of the NIN-AND tree, where each root node is labeled by $e^+ \not\leftarrow c_i^+$ (single-cause) and there are no negation links. We summarize this in the following proposition.

Proposition 4: The NIN-AND tree degenerates to the noisy-OR, if it has a single NIN-AND-gate, has no negation links, and labels every event node by a causal failure $e^+ \not\leftarrow c_i^+$.

VI. NIN-AND Tree Evaluation

A NIN-AND tree can be used to evaluate $P(e^+ \leftarrow \underline{x}^+)$ given $P(e^+ \leftarrow \underline{y}^+)$ or $P(e^+ \not\leftarrow \underline{y}^+)$ for each root node. The computation can be performed recursively by decomposing the noisy-AND tree into subtrees. The following lemma shows that such decomposition is valid.

Lemma 1: Let T be a NIN-AND tree, the leaf of T be v, and the gate connected to v be g. Let v and g be deleted from T, as well as the links incoming to g. In the remaining graph, each component is either an isolated event node or a NIN-AND tree.

Proof:

It suffices to show that if a component resultant from the deletion is not an isolated event node, then it is a well-defined NIN-AND tree. We show that such a component has a unique leaf node and it is labeled according to condition 3 of Definition 6.

Let w be an event node connected to the input of g in T, and w is connected to the output of a gate g_w. Because T is a tree, after deletion, the component with w is also a tree which we denote by T_w. Since each gate in T has out-degree 1, w is the unique leaf of T_w. Let $\underline{y}_i^+$ be the causes in the label of a root node in T_w. From condition 4 of Definition 6, it follows that the causes in the label of node w is $\cup_i \underline{y}_i^+$, where the index i is over all root nodes in T_w. □

A NIN-AND tree can be evaluated according to the following recursive algorithm.

Algorithm 1: GetCausalEventProb(T)

```
Input: A NIN-AND tree $T$ where probability for each root node is specified.
denote leaf of $T$ by $v$ and gate connected to $v$ by $g$;
for each node $w$ directly connected to input of $g$, do
    if probability $P(w)$ for event at $w$ is not specified,
        denote the sub-NIN-AND-tree with $w$ as the leaf by $T_w$;
        $P(w) = GetCausalEventProb(T_w)$;
    if $(w, g)$ is a forward link, $P'(w) = P(w)$;
    else $P'(w) = 1 - P(w)$;
$P(v) = \prod_w P'(w)$;
return $P(v)$;
```

Consider an execution of GetCausalEventProb(T) where T is shown in Fig. 4 and probabilities for root nodes are specified as

$$P(e^+ \leftarrow c_1^+) = 0.85, P(e^+ \not\leftarrow c_2^+) = 0.2, P(e^+ \leftarrow c_3^+) = 0.7.$$

Leaf v is labeled $e^+ \not\leftarrow c_1^+, c_2^+, c_3^+$ and g is the lower NIN-AND gate in the figure. The *for* loop iterates through the two input nodes of v. Let w (line 2) be the node labeled $e^+ \not\leftarrow c_2^+$. Since $P(w)$ has been specified, lines 3 through 5 are skipped. Since w connects to g by a forward link, we have $P'(e^+ \not\leftarrow c_2^+) = 0.2$ (line 6).

Next, let w be the node labeled $e^+ \leftarrow c_1^+, c_3^+$. Because $P(w)$ is not specified, the recursive call GetCausalEventProb(T_w) in line 5 is made, where T_w is the sub-NIN-AND-tree with w as the leaf and with the upper NIN-AND gate only.

In the recursive call, leaf v is labeled $e^+ \leftarrow c_1^+, c_3^+$. The *for* loop iterates through two root nodes w_1 labeled $e^+ \leftarrow c_1^+$ and w_2 labeled $e^+ \leftarrow c_3^+$. In the first iteration over w_1, since $P(w_1)$ is specified and w_1 is connected to the NIN-AND gate by a forward link, we have $P'(w_1) = P(w_1)$ in line 6. Similarly, we have $P'(w_2) = P(w_2)$ in the second iteration. After the *for* loop is completed, $P(e^+ \leftarrow c_1^+, c_3^+) = P'(w_1) * P'(w_2) = 0.85 * 0.7 = 0.595$ is computed at line 8. GetCausalEventProb(T_w) now completes with the value returned.

This brings line 5 in the original GetCausalEventProb(T) to completion. Since the node w, labeled by $e^+ \leftarrow c_1^+, c_3^+$, is connected to the lower NIN-AND gate by a negation link, $P'(w) = 1 - P(w) = 0.405$ is computed in line 7. The *for* loop is now completed. In line 8, $P(v) = 0.2 * 0.405 = 0.081$ is computed. The final result $P(e^+ \not\leftarrow c_1^+, c_2^+, c_3^+) = 0.081$ is then returned.

The following theorem establishes the soundness of GetCausalEventProb. We define the *depth* of a NIN-AND tree to be the maximum number of gate nodes contained in a path from a root to the leaf.

Theorem 1: Let T be a NIN-AND tree where probability for each root node is specified in the range $(0, 1)$ and $P(v)$ be returned by $GetCausalEventProb(T)$. Then $P(v)$ is a probability in the range $(0, 1)$ and it combines given probabilities according to reinforcement with failure conjunction and independence, or undermining with success conjunction and independence, specified by the topology of T.

Proof:

According to Lemma 1, T_w in $GetCausalEventProb$ is a valid NIN-AND tree. Hence, algorithm $GetCausalEventProb$ is well defined.

$GetCausalEventProb$ evaluates first the output event for each gate node whose inputs are root events. If causes in root event labels are reinforcing, evaluation is performed according to Eqn (2). Otherwise (undermining), evaluation is performed according to Eqn (5). Hence, the evaluation result is a probability in the range $(0, 1)$ and reflects the effect of reinforcement, due to Proposition 1, or undermining, due to Proposition 3.

After the evaluation, root nodes no longer participate in further evaluations and can be deleted from the tree. The remining subtree is still a valid NIN-AND tree with the depth reduced by one. $GetCausalEventProb$ repeats the above computation until the depth is reduced to zero. The statement is true for the evaluation at each depth and hence the theorem holds. □

Note that the topology of T is a crucial piece of knowledge. For the recovery example, suppose the physician articulates a different order, which is shown in Fig. 5. In this scenario, the physician feels that the reinforcing interaction between c_1 and c_2 should be considered first. The undermining interaction between sets $\{c_1, c_2\}$ and $\{c_3\}$ should then be considered. Applying GetCausalEventProb, we obtain $P(e^+ \not\leftarrow c_1^+, c_2^+) = 0.03$ and $P(e^+ \leftarrow c_1^+, c_2^+, c_3^+) = 0.679$.

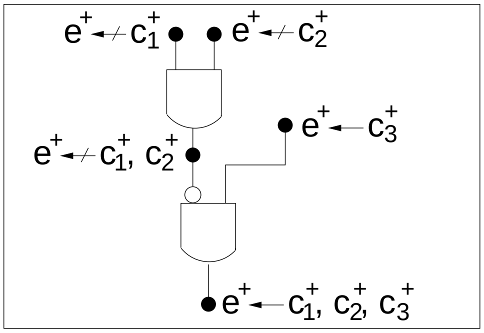

$e^+ \not\leftarrow c_1^+$
$e^+ \not\leftarrow c_2^+$
$e^+ \leftarrow c_3^+$
$e^+ \not\leftarrow c_1^+, c_2^+$
$e^+ \leftarrow c_1^+, c_2^+, c_3^+$

Fig. 5. The alternative NIN-AND tree model for the recovery example.

VII. Relaxing Default Assumptions

A NIN-AND tree assumes, by default, failure conjunction and independence for sets of reinforcing causes, and success conjunction and independence for sets of undermining causes. For some sets of causes, the default assumptions may not be consistent with the expert's knowledge. As the result, the expert may disagree with the probability of the output event of a gate node. When this occurs, the NIN-AND tree model allows coherent incorporation of the new information by deleting the corresponding gate node from the tree and replacing the evaluated probability value by a directly assessed value. In particular, let g be the gate in question and its output be connected to event node v. If the expert disagrees with the event probability computed for node v, the entire subtree with v as the leaf can be discarded by deleting the link (g, v). Node v remains in the resultant new NIN-AND tree as a root node. The expert can then assess a proper event probability for v.

For instance, with the NIN-AND tree in Fig. 4, suppose that the expert disagrees with $P(e^+ \leftarrow c_1^+, c_3^+) = 0.595$. Instead, (s)he feels that 0.4 is more appropriate. Note that this assignment is consistent with the undermining nature of the interaction between c_1 and c_3, but the degree of undermining is different from what the default assumptions dictate. Then root nodes labeled by $e^+ \leftarrow c_1^+$ and $e^+ \leftarrow c_3^+$ can be removed, as well as the gate connected to them. As the result, node $e^+ \leftarrow c_1^+, c_3^+$ becomes a root node and $P(e^+ \leftarrow c_1^+, c_3^+) = 0.4$ is assigned to it. Applying GetCausalEventProb to the new NIN-AND tree, $P(e^+ \leftarrow c_1^+, c_2^+, c_3^+) = 0.88$ is obtained.

This flexibility of the NIN-AND tree allows it to be used interactively, increasing its expressive power as a tool for probability elicitation: An expert can start by articulating a NIN-AND tree where each root is labeled by a single-cause c_i. The default assumptions on failure and success independence now allow computation of the probability for each non-root causal event. This can be viewed as the first approximation of the expert's subjective belief. The expert can then examine each computed event probability and decide if it is consistent with his/her belief.

Upon identification of disagreement over a node v connected to the output of a gate g, the expert can trace backward to input events connected to g. The expert will decide whether (s)he disagrees with the probabilities of any input events. If no such disagreement is identified, then the expert must be disagreeing with the degree of reinforcement or undermining implied by the default assumptions on event conjunction and independence. (S)he can then assess a probability for the output event as illustrated above.

On the other hand, if disagreement with the probability of an input event connected to g is identified, the processing continues by tracing further back from the event node v towards root nodes.

During the above processing, the expert's attention shifts from the leaf to roots and assessment at each step is local (relative to the output and inputs of a single gate). Therefore, a software with a well-designed GUI (allowing the expert to draw the NIN-AND tree topology, to indicate a point of disagreement, and to replace a probability) is sufficient to aid the elicitation. A knowledge engineer is not required after the initial training on the operation of the software.

It is possible that as the expert traces disagreements, deletes subtrees, and replaces event probabilities, an initially deep NIN-AND tree becomes shallow in the end. Many root node labels now consist of a subset of causes, instead of a single-cause initially. The resultant NIN-AND tree becomes topologically different. This does not mean that the initial NIN-AND tree was wrong: The initial tree has served its useful role in elicitation and has adapted more faithfully to the expert's subjective knowledge.

VIII. Complexity Analysis

Elicitation of a CPT using the NIN-AND tree model involves the specification of the tree topology and the numerical parameters. The complexity of topology specification can be measured by the number of nodes in the NIN-AND tree. Because each gate node is connected to a unique event node through a forward link, only event nodes need to be counted. For instance, the NIN-AND tree in Fig. 5 has five event nodes and two gate nodes. By pairing the two gate nodes with their output event nodes, the equivalent number of nodes to be specified is counted as five.

Given an arbitrary NIN-AND tree of n root event nodes, the total number of event nodes of the tree can be increased as follows: Select a gate node of $k \geq 3$ incoming links. Replace the gate node and the event nodes connected to it with a subtree of two or more gate nodes. For instance, a gate node connected to the input event nodes $e^+ \leftarrow c_1^+$, $e^+ \leftarrow c_2^+$, $e^+ \leftarrow c_3^+$ and the output event node $e^+ \leftarrow c_1^+, c_2^+, c_3^+$ can be replaced with the subtree in Fig. 5. The number of event nodes contributed to the total counting is thus increased from four to five. Note that this modification involves changing the labels of some event nodes and modifying some forward links into negation links. Since the focus here is the counting of the total number of event nodes, these issues are not of the concern.

If the above replacement is performed repeatedly, eventually there will be no gate node left of $k \geq 3$ incoming links. At this point, the total number of event nodes reaches its upper bound. In the resultant tree, each gate node has exactly two input events. Hence, the number of total event nodes is upper bounded by $2n-1$, yielding the complexity of $O(n)$ for topology specification.

Next, we consider the specification of numerical parameters. Given the topology of the NIN-AND tree, only the probability for each root event node needs to be directly assessed, again yielding the complexity of $O(n)$.

Suppose that the number of parent variables involved in the CPT is n. The above deals with the acquisition of a single probability $P(e^+|c_1^+, c_2^+, ..., c_n^+) = P(e^+ \leftarrow c_1^+, c_2^+, ..., c_n^+)$ of the CPT where all causes are turned on. For each other probability in the CPT, some causes are turned off, e.g., c_1 is turned off in probability $P(e^+|c_1^-, c_2^+, ..., c_n^+) = P(e^+ \leftarrow c_2^+, ..., c_n^+)$. The only necessary modification to the NIN-AND tree is to remove the root node labeled by $e^+ \leftarrow c_1^+$ (or $e^+ \not\leftarrow c_1^+$) and then $P(e^+ \leftarrow c_2^+, ..., c_n^+)$ can be computed in the same fashion. Note that if the removal of the root node reduces the incoming links of the corresponding gate node to one, the gate node can be removed as well. Since no new probability parameters need to be assessed, the acquisition of the entire CPT has the complexity of $O(n)$.

The preceding presentation assumes no modification to the computed probabilities of intermediate events. If the human expert does not agree with some of the default assumptions encoded in the NIN-AND tree, some probabilities of intermediate events need to be directly assessed as outlined in Section VII. The number of such probabilities are upper-bounded by the maximum possible number of event nodes in the NIN-AND tree, namely, $2n-1$. Hence, the complexity is $O(n)$. If α probabilities in the CPT requires such modification, then the complexity of acquisition of the CPT becomes $O(\alpha\ n)$. The value of α is expected to be a small positive integer in practice.

In the worst case, the human expert can do away from the causal model and directly assess each probability value of the CPT. The complexity would be $O(2^n)$. Therefore, the NIN-AND tree can be viewed as a flexible tool that provides a range of acquisition complexity from $O(n)$ to $O(\alpha\ n)$ to $O(2^n)$. In one extreme, the expert only has to assess the causal interactions qualitatively as reinforcing or undermining and the n single-cause event probabilities. The default assumptions for reinforcement and undermining will filling in the gap between the $O(n)$ assessments and necessary $O(2^n)$ probability values in the CPT. As the experience of expert becomes richer and the time of the expert becomes more available, more default patterns for reinforcement and undermining embedded in the evaluated probability values can be replaced by directly assessed probability values, making them more accurate.

IX. CPT Elicitation With the NIN-AND tree

We demonstrate how to use the NIN-AND tree to elicit CPTs in BNs with a more challenging example. It is feasible (though undesirable) to directly elicitate the CPT for previous examples, but it is much less so for this example. Consider an effect (child) variable e with a set of seven causes (parents) in a BN: $c_1, ..., c_7$. Suppose that a human expert identifies the following three subsets of causes and the interaction within each subset:

- Subset s_1: c_1 and c_2 are undermining each other.
- Subset s_2: c_3, c_4 and c_5 are reinforcing each other.
- Subset s_3: c_6 and c_7 are reinforcing each other.

The expert assesses that the interaction between subsets s_1 and s_2 is also undermining and, together as a group, they reinforce s_3. These assessments produce the NIN-AND tree in Fig. 6. Suppose that the following probabilities for single-cause events are also assessed:

$$P(e^+ \leftarrow c_1^+) = 0.65, P(e^+ \leftarrow c_2^+) = 0.35, P(e^+ \leftarrow c_3^+) = 0.8,$$
$$P(e^+ \leftarrow c_4^+) = 0.3, P(e^+ \leftarrow c_5^+) = 0.6,$$
$$P(e^+ \leftarrow c_6^+) = 0.75, P(e^+ \leftarrow c_7^+) = 0.55.$$

To evaluate $P(e^+|c_1^+, ..., c_7^+)$, GetCausalEventProb is applied to obtain

$$P(e^+|c_1^+, ..., c_7^+) = P(e^+ \leftarrow c_1^+, ..., c_7^+) = 0.912.$$

Note that the noisy-OR model would produce 0.999. To evaluate $P(e^+|c_1^+, c_2^+, c_3^-, c_4^+, c_5^+, c_6^+, c_7^+)$, eliminate node $e^+ \not\leftarrow c_3^+$ from

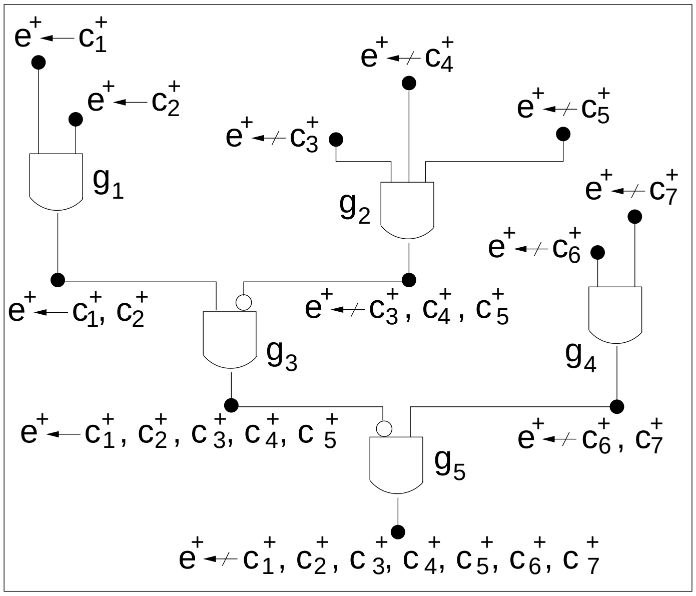

$e^+ \leftarrow c_1^+$
$e^+ \leftarrow c_2^+$
g_1
$e^+ \not\leftarrow c_4^+$
$e^+ \not\leftarrow c_3^+$
$e^+ \not\leftarrow c_5^+$
g_2
$e^+ \not\leftarrow c_7^+$
$e^+ \not\leftarrow c_6^+$
$e^+ \leftarrow c_1^+, c_2^+$
$e^+ \not\leftarrow c_3^+, c_4^+, c_5^+$
g_3
g_4
$e^+ \leftarrow c_1^+, c_2^+, c_3^+, c_4^+, c_5^+$
$e^+ \not\leftarrow c_6^+, c_7^+$
g_5
$e^+ \not\leftarrow c_1^+, c_2^+, c_3^+, c_4^+, c_5^+, c_6^+, c_7^+$

Fig. 6. An example NIN-AND tree.

Fig. 6 and modify output labels for g_2, g_3 and g_5. The evaluation gives

$$P(e^+|c_1^+, c_2^+, c_3^-, c_4^+, c_5^+, c_6^+, c_7^+)$$
$$= P(e^+ \leftarrow c_1^+, c_2^+, c_4^+, c_5^+, c_6^+, c_7^+) = 0.906.$$

The same NIN-AND tree has been used to assess both probabilities above. This is desirable but not required. That is, the NIN-AND tree model does not require that different causal probabilities to be assessed using the same tree. If the expert feels that a particular combination of a subset of causes follows a different pattern of interaction, a distinct NIN-AND tree can be used for acquiring the corresponding causal probability. Commonly, it is expected that one tree topology is used for assessment of all probabilities in a CPT. If the expert is happy with the result, the complexity of his/her assessment task is $O(n)$, where n is the number of causes.

For the sake of demonstration, suppose that the expert believes that 0.906 is too high for $P(e^+|c_1^+, c_2^+, c_3^-, c_4^+, c_5^+, c_6^+, c_7^+)$. (S)he attributes it to probability $P(e^+ \not\leftarrow c_6^+, c_7^+) = 0.113$ of the output event at gate g_4 as too low. Instead, (s)he believes that 0.2 is a better assessment. In response, the subtree with g_4 as the leaf is removed and probability 0.2 is assigned to the root event node $e^+ \not\leftarrow c_6^+, c_7^+$. GetCausalEventProb then evaluates $P(e^+|c_1^+, c_2^+, c_3^-, c_4^+, c_5^+, c_6^+, c_7^+)$ to be 0.833.

X. Conclusions

Causal interactions may be reinforcing or undermining. A causal model to aid CPT elicitation during engineering of Bayesian networks needs to be expressive enough so that both types of interactions can be encoded. We have shown that existing causal models can only model one type of interactions. We present the first general causal model, the NIN-AND tree, that supports the expression of both reinforcement and undermining.

Existing causal models, except the RNOR, limit model parameters to probabilities of single-cause events. As the RNOR, the NIN-AND tree allows integration of probability parameters of both single-cause events and multi-cause events.

In addition to numerical probability parameters, the NIN-AND tree model requires elicitation of the tree topology, which is equivalent to a partial ordering on how causal interactions of causes should be integrated. Given that interactions of opposite natures are considered and the role of each cause relative to each type of interactions must be specified, this appears to be a necessary but not expensive overhead.

As any parameterized models, not all possible causal interactions can be expressed by the NIN-AND tree. For instance, it is conceivable that causes c_1 and c_2 reinforce each other, and c_1 and c_3 undermine each other. Using the NIN-AND tree, either c_3 is modeled as undermining the reinforcing group c_1 and c_2, or c_2 is modeled as reinforcing the undermining group c_1 and c_3. To model the above interaction exactly as stated, (1) additional information must be specified on how the two groups (with a shared cause) interact; and (2) a multiply connected structure is needed, which is no longer a tree.

In conclusion, the NIN-AND tree provides a simple yet expressive model for efficient CPT elicitation in engineering probabilistic graphical models. It is worth noting that Pearl [11] analyzed causation using functional causal models and our work is consistent with his functional approach. In particular, we have proposed the NIN-AND tree as a useful boolean functional model. Our ongoing efforts include extending the NIN-AND tree from binary to multi-valued variables and exploring the model for efficiency gain in probabilistic inference.

Acknowledgements

The financial support from National Sciences and Engineering Research Council (NSERC) of Canada through Discovery Grant to the first author is acknowledged. We thank anonymous reviewers for their helpful comments.

References

[1] F.J. Diez. Parameter adjustment in Bayes networks: The generalized noisy or-gate. In D. Heckerman and A. Mamdani, editors, *Proc. 9th Conf. on Uncertainty in Artificial Intelligence*, pages 99–105. Morgan Kaufmann, 1993.

[2] S.F. Galan and F.J. Diez. Modeling dynamic causal interactiosn with Bayesian networks: temporal noisy gates. In *Proc. 2nd Inter. Workshop on Causal Networks*, pages 1–5, 2000.

[3] I. Good. A causal calculus (i). *British Journal of Philosophy of Science*, 11:305–318, 1961.

[4] E. Graham. How to live in harmony with your mother-in-law or daughter-in-law. `http://www.marriagemissions.com/family_issues/how_to_live.php`.

[5] D. Heckerman. Causal independence for knowledge acquisition and inference. In D. Heckerman and A. Mamdani, editors, *Proc. 9th Conf. on Uncertainty in Artificial Intelligence*, pages 122–127. Morgan Kaufmann, 1993.

[6] D. Heckerman and J.S. Breese. Causal independence for probabilistic assessment and inference using Bayesian networks. *IEEE Trans. on System, Man and Cybernetics*, 26(6):826–831, 1996.

[7] M. Henrion. Some practical issues in constructing belief networks. In L.N. Kanal, T.S. Levitt, and J.F. Lemmer, editors, *Uncertainty in Artificial Intelligence 3*, pages 161–173. Elsevier Science Publishers, 1989.

[8] U. Kuter, D. Nau, D. Gossink, and J.F. Lemmer. Interactive course-of-action planning using causal models. In *Proc. 3rd Inter. Conf. Knowledge Systems for Coalition Operations*, pages 37–51, 2004.

[9] J.F. Lemmer and D.E. Gossink. Recursive noisy OR - a rule for estimating complex probabilistic interactions. *IEEE SMC, Part B*, 34(6):2252–2261, 2004.

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

[11] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

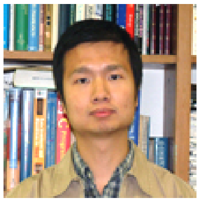
[12] G. Shafer. *The Art of Causal Conjecture*. MIT press, 1997.

[13] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search (Springer Lecture Notes in Statistics)*. New York: Springer-Verlag, 1993.

[14] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search (Springer Lecture Notes in Statistics, 2nd ed.)*. Cambridge, MA: MIT Press, 2000.

[15] S. Srinivas. A generalization of noisy-or model. In D. Heckerman and A. Mamdani, editors, *Proc. 9th Conf. on Uncertainty in Artificial Intelligence*, pages 208–215. Morgan Kaufmann, 1993.

Yang Xiang is a Professor of Computing & Information Science at the University of Guelph, Canada, where he also directs the Intelligent Decision Support System Lab. He received his Ph.D. from the University of British Columbia, Canada. His research interests include probabilistic and decision-theoretic graphical models, multiagent graphical models, collaborative design, decision making and planing, distributed constraint satisfaction, knowledge discovery and data mining, diagnosis and scheduling. He published a monograph on multiagent probabilistic reasoning using graphical models and developed WebWeavr, a comprehensive Java toolkit for decision support with graphical models.



Ning Jia received his Bachelor degree of Information Science from Peking University, China in 1998 and Master degree of Computer Science from University of Guelph, Canada in 2005. Since 2005, he has been working for Desire2Learn Inc.