WRDC-TR-90-6002

Graphical Inference in Qualitative Probabilistic Networks

Michael P. Wellman
AI Technology Office
WRDC/TXI

June 1990

Technology Exploitation Directorate
Wright Research and Development Center
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio 45433

Graphical Inference in Qualitative Probabilistic Networks

Michael P. Wellman
AI Technology Office
WRDC/TXI
Wright-Patterson AFB, Ohio 45433
(513) 255-5800
wellman@wrdc.af.mil

Abstract

Qualitative probabilistic networks (QPNs) are abstractions of influence diagrams that encode constraints on the probabilistic relation among variables rather than precise numeric distributions. *Qualitative relations* express monotonicity constraints on direct probabilistic relations between variables, or on interactions among the direct relations. Like their numeric counterpart, QPNs facilitate graphical inference: methods for deriving qualitative relations of interest via graphical transformations of the network model. However, query processing in QPNs exhibits computational properties quite different from basic influence diagrams. In particular, the potential for information loss due to the incomplete specification of probabilities poses the new challenge of minimizing ambiguity. Analysis of the properties of QPN transformations reveals several characteristics of admissible graphical inference procedures.

Extended version (with proofs) of an article to appear in *Networks*, 1990.

1 Qualitative Probabilistic Networks

The modeling advantages of influence diagrams are largely attributable to the graphical nature of the representation, which highlights the important relationships among the domain variables. As in the related belief-network representation [5], the structure of an influence diagram reveals the dependencies (and implicitly, the independencies) operating among the variables of interest. Furthermore, transformation-based algorithms for evaluating influence diagrams—such as Shachter's [8]—preserve the model's structural properties at each stage of the computation. Each transformation step corresponds to an intuitive operation on the variables ("averaging out" or "belief revision"). Such *graphical inference* is attractive because the operators may be arranged flexibly to answer a variety of queries [9].

These desirable properties of influence diagram models emerge from the representation's structural features alone and do not depend on the precise specification of conditional probabilities. Given this observation, it is natural to investigate other decision model representations that share the structural features of influence diagrams but support a different class of inferences about the relationships among decisions and events. In particular, we are interested in finding representations that are more robust than numerical decision models, yet still provide decision-theoretic justification for some choices among competing plans.

Qualitative probabilistic networks (QPNs) [12] are influence diagrams with the numeric conditional probability tables replaced by qualitative probabilistic relations among the variables. These *qualitative relations* can be viewed as constraints on the conditional probability tables, expressing inequalities that must be satisfied by the various elements.[1]

The node types and topological dependence properties of QPNs and influence diagrams are identical. Because qualitative relations are weaker than numeric conditional probability tables, QPNs are abstractions of influence diagrams. While an influence diagram represents a particular joint distribution among its random variables, a QPN represents a family of such distributions satisfying its constraints. Without a unique joint distribution, of course, it is not generally possible to determine the optimal decision. Rather, the aim of inference in QPNs is to derive constraints on the *form* of the optimal decision *policy*.

2 Qualitative Relations

The constraints on probabilistic relations expressible in QPNs are not arbitrary. In fact, QPNs permit only very regular patterns of inequalities to be asserted about conditional probabilities. All qualitative relations take the form of conditions on the comparative probability distribution for a variable given various values for its predecessors in the network. Let F_c denote the cumulative probability distribution for c as a function of its predecessors.

[1] The term "table" here is somewhat misleading because QPN variables need not be discrete.

This distribution is typically represented in an influence diagram by a conditional probability table. In QPNs, qualitative relations constrain the value of F_c without specifying it precisely. Relations of one type, called *qualitative influences*, restrict the relative values of F_c upon variation of one of c's predecessors. *Qualitative synergies* constrain the behavior of this distribution upon changes of two predecessors at once. Both types of relation can be interpreted as probabilistic versions of monotonicity conditions on the partial derivatives, first- and second-order, respectively.

Specifically, a qualitative influence from a to c with sign $+$ $(-)$ means that the probability distribution for c given a is nondecreasing (nonincreasing) in a—all else equal—in the sense of first-order stochastic dominance. The notation $S^\delta(a, c)$, $\delta \in \{+, -, 0, ?\}$, is used to denote such an influence.

Definition 1 (qualitative influences) *$S^\delta(a, c)$ holds in a qualitative probabilistic network G if and only if (iff), for all values c_0 of c, x of c's predecessors in G other than a, and $a_1 > a_2$ of a:*

$$F_c(c_0|a_1x) \;\; R_\delta \;\; F_c(c_0|a_2x),$$

where R_δ is $\leq$, $\geq$, or $=$ as δ is $+$, $-$, or 0, and $R_?$ is the complete relation (thus $S^?$ always holds).

Such a complex specification of a rather straightforward concept is necessary for completeness. The condition that the first cumulative distribution be no greater than the second for all values of c is a requirement of first-order stochastic dominance. (Lower values for the cumulative distribution correspond to higher probabilities for the larger values.) Quantification over all values for c's predecessors (represented by the variable x, typically a vector) realizes the "all else equal" part of the definition. Fixing all variables but one corresponds to the usual interpretation of partial derivatives.

The qualitative synergy relation is intended to capture the intuition that an increase in one variable has greater effect at higher levels of the other. In other words, the combined effect of increasing the variables is greater than taking the two effects independently. One way to define this probabilistically is demonstrated by the following qualitative synergy (Y^δ) condition.

Definition 2 (qualitative synergies) *$Y^\delta(\{a, b\}, c)$ holds in a QPN G iff for all values c_0 of chance node c, x of c's predecessors in G other than a and b, $a_1 > a_2$ of a, and $b_1 > b_2$ of b:*

$$[F_c(c_0|a_1b_1x) - F_c(c_0|a_2b_1x)] \;\; R_\delta \;\; [F_c(c_0|a_1b_2x) - F_c(c_0|a_2b_2x)].$$

Qualitative synergy on value, $Y^\delta(\{a, b\}, v)$ holds iff the utility function u satisfies the following (with the same conditions on a, b, and x):

$$[u(a_2, b_1, x) - u(a_1, b_1, x)] \;\; R_\delta \;\; [u(a_2, b_2, x) - u(a_1, b_2, x)]. \tag{1}$$

A QPN is represented by a directed acyclic graph, with nodes representing variables and signed edges and hyper-edges representing qualitative relations. A link from a to b with sign δ denotes an assertion that $S^{\delta}(a,b)$. By convention, S^0 assertions are implicit in the absence of a link.[2] It follows from the probabilistic definitions that any variable without an influence link to some target is also non-synergistic (Y^0) with any other variable on that target. However, the absence of a synergy hyper-edge for variables with nonzero influences to a target does not imply anything about their interaction or non-interaction. Lacking an explicit synergy, we adopt the conservative default, $Y^?$.

The probabilistic definitions of the qualitative relations were chosen for two primary properties. First, the relations are preserved by certain graphical inference operations. These are discussed in some depth in Section 4 below. Second, qualitative relations involving decision variables and the value node constrain the form of the optimal decision policy. The example of Section 3 demonstrates these decision-theoretic implications of QPNs. See [12] for further motivation and discussion of the mathematical properties of qualitative influences and synergies.

3 Example: A Generic Decision Model

A significant advantage of the qualitative formalism is that the relations are valid for a wide range of contexts and interpretations of the variables. In support of this assertion, Figure 1 presents the QPN representation of a generic decision problem.

generic.ps

Figure 1: The generic decision model.

The model of Figure 1 includes two decision variables, representing the choices of observations and actions. These variables may be propositional (whether to perform the action), or may admit a range of values on an ordered scale. The "act" has both costs and benefits, each positively related to the degree to which the act is performed. For example, both the therapeutic results (benefits) and undesirable side effects (costs) would increase in the sense of S^+ with the aggressiveness of a medical treatment (act). By definition, costs negatively influence the value node whereas the influence of benefits is positive.

There is also an unknown "state of nature," θ, which affects the valuation of costs and benefits. Although it makes no commitment about the direct effect of θ on value ($S^?(\theta, v)$), the generic decision model requires that the state of nature be positively synergistic with

[2]See Pearl [5] for a detailed treatment of independence implicit in the topology of a directed probabilistic network.

both costs and benefits. The synergy with benefits implies that the beneficial effects of the action are enhanced for higher values of θ. For example, if θ represents underlying disease severity and the *benefits* of therapy reflect degree of cure, then the synergy captures the idea that a cure has greater value to patients whose diseases are more severe. The positive synergy with costs rules out the possibility that θ exacerbates the negative effects of the actions. In the therapy example, this implies that disease severity is at worst neutral with respect to the side effects of the treatment.

Although the state θ is not directly observable by the decision maker, there is another variable, *info*, which is related to θ by a positive influence. The dashed line from the decision variable *observe* indicates that if the observation is performed, the value of *info* is accessible to the decision maker for use in choosing a value for the action. The act of observing has costs, but no benefits other than the value of information revealed in this way.[3]

The generic decision model is clearly inadequate to justify a particular choice for the decision variable *act*. However, the model is strong enough to provide useful constraint on the optimal action policy as a function of the information available. Using the graphical inference operations described in Section 4, the query algorithm transforms the original QPN into the reduced model of Figure 2.

reduced-generic.ps

Figure 2: The generic decision model reduced to three variables.

As expected, the model fails to establish a nondegenerate relationship between *act* and value. The useful result of Figure 2 is the positive synergy $Y^+(\{\mathit{info}, \mathit{act}\}, v)$. Positive synergy imposes a constraint on the utility function (1) that is sufficient to establish the *monotone decision property* [7] on *act* and *info*. This property dictates that the optimal value of *act* is a nondecreasing function of *info*. Thus, if the observation is performed and the information is available, the decision maker should raise the level of *act* when the information indicates the expected benefits are greater (and costs are no worse). If the decision maker's policy is to choose $\mathit{act} = \mathit{act}_1$ upon observing $\mathit{info} = \mathit{info}_1$, then consistency requires that it choose a value for act no greater than act_1 when it observes $\mathit{info}_2 < \mathit{info}_1$. If *act* is a propositional variable, the synergy result implies that the optimal policy is to take the action iff *info* exceeds some threshold value.

[3] The dashed line is an auxiliary notational element not formally part of the qualitative probabilistic network. This convention runs counter to the usual approach in influence diagrams, which include informational links as part of the model. In an influence diagram, performing the observation would actually change the state of *info*. Since QPNs were designed to be used within the context of a general-purpose planning program [11], I chose to maintain a uniform interpretation for observable variables and relegate the task of maintaining observability constraints to the planner.

Although the details of any particular decision situation will provide a stronger basis for choosing a decision policy, the generic analysis above serves to bound the strategy space at a very high level of abstraction. Such bounds are useful for restricting the search for optimal policies, even with highly incomplete problem specifications. For example, we have applied the generic decision model to justify rules for critiquing decision trees constructed by human analysts using a general decision-making tool [13]. Because the constraints are justified by a generic model, the critiquer needs only a qualitative understanding of the domain-specific actions and events included in the model.

The fact that we can constrain the form of optimal decision policies at a high level of abstraction is a feature unique to qualitative decision models. Of course, the literature is full of generic decision-theoretic results derived from abstract mathematical models. The contribution here is a qualitative decision-modeling formalism that permits such results to be derived automatically via a uniform inference procedure.

4 Graphical QPN Operations

The key idea behind Shachter's algorithm for influence diagram evaluation [8] is that of truth-preserving graphical manipulations. For numeric influence diagrams, truth-preserving means that the joint distribution represented by the revised model is implied by the structures in the original model. In the qualitative context, truth-preserving means that qualitative relations in the original network entail qualitative relations on the joint distribution of variables in the modified network. In other words, any joint probability distribution consistent with the original QPN remains consistent after performing the truth-preserving operation.

Inference in QPNs—as in influence diagrams—is accomplished by applying sequences of two truth-preserving operations: node reduction and link reversal. A chance node is eligible for reduction iff it has at most one direct successor. Decision and value nodes can be reduced simply by splicing them out and deleting any dangling links. A link between chance nodes is eligible for reversal as long as there is no other directed path between the two nodes (otherwise, the reversal would introduce a cycle). We define these operations by specifying the qualitative relations holding in the modified QPN as a function of those in the original model.

Suppose node c is eligible for reduction in G and let G' be the QPN obtained by reducing c. The nodes of G' are just the nodes of G minus c. If c has no successors in G, then G' maintains all qualitative relations of the original save those incident on c. Otherwise, c has exactly one successor; call it d. In this case, all qualitative relations except those incident on c or d are retained in G'. Let $\delta_{a,c}$ represent the direction of influence between variables a and c in G, and similarly $\delta_{\{a,b\},c}$ the synergy of a and b on c. The modified influence of a variable a on d in G' is $\delta'_{a,d}$, where

$$\delta'_{a,d} = \delta_{a,d} \oplus (\delta_{a,c} \otimes \delta_{c,d}). \tag{2}$$

The operators $\oplus$ and $\otimes$ denote sign addition and multiplication, respectively. Thus, the influence update equation (2) sanctions chaining of qualitative influences, as long as the sign obtained from a traversal of the path through c agrees with the existing link, if any, from a to d. If the two paths disagree (or one is "?"), $\delta'_{a,d}$ is assigned the "?" sign. If a and d were not directly linked in G ($\delta_{a,d} = 0$, the identity element for $\oplus$), then G' will include an influence determined solely by the path through c.

The procedure for updating synergy links after reduction is similar, although there are more factors to consider. For every pair of variables a and b that have influences on c or d in G, the modified synergies are computed as follows:

$$\delta'_{\{a,b\},d} = \delta_{\{a,b\},d} \oplus (\delta_{\{a,b\},c} \otimes \delta_{c,d}) \oplus (\delta_{b,c} \otimes \delta_{\{a,c\},d}) \oplus (\delta_{a,c} \otimes \delta_{\{b,c\},d}). \tag{3}$$

Figure 3 illustrates the situation before the update. In typical applications of reduction, not all of the links shown are actually present in G. In such cases, some of the terms in the above equation simply drop out.

update.ps

Figure 3: A fragment of QPN G before performing a QPN transformation. The relations on d holding in G' after reducing c from the network are given by the update equations (2) and (3). Reversing the link from c to d results in G'', described by equations (4–6).

The second operation on QPNs is link reversal. Suppose we wish to reverse the influence link from c to d in G, obtaining the modified QPN G''. G'' retains all the variables of G, as well as all of the qualitative relations not incident on c or d. Using Bayes's theorem, it is possible to show that the qualitative relation of d on c in the reversed network has the same sign as the original link:[4]

$$\delta''_{d,c} = \delta_{c,d}. \tag{4}$$

(Since QPNs are acyclic, G'' cannot contain a link from c to d.) For any variable a that had a link to c or d in G, the new qualitative influences are as follows.

$$\begin{aligned} \delta''_{a,c} &= \delta_{a,c} \oplus (\delta_{a,d} \otimes ?) \\ \delta''_{a,d} &= \delta_{a,d} \oplus (\delta_{a,c} \otimes \delta_{c,d}). \end{aligned} \tag{5}$$

We also must modify any synergy involving a on these variables.

$$\delta''_{\{a,b\},d} = \delta_{\{a,b\},d} \oplus (\delta_{\{a,b\},c} \otimes \delta_{c,d}) \oplus (\delta_{a,c} \otimes \delta_{\{b,c\},d}) \oplus (\delta_{b,c} \otimes \delta_{\{a,c\},d}). \tag{6}$$

Synergies on c in G'' are assigned "?" because more specific signs are not determinable from the pre-reversal qualitative relations.

[4] Derivation for this as well as all other update rules are presented elsewhere [12].

5 Query Processing

The purpose of inference in QPNs is to determine qualitative relations entailed by those specified in the original network. By applying sequences of the truth-preserving operations above, we can transform the original QPN to one where the relations of interest are direct. For example, repeated reductions and reversals transform the generic QPN of Figure 1 to that of Figure 2, where the qualitative relations of the action and information on value are explicit.

A QPN query specifies j, the *target node*, and a set of *conditioning variables*, K. In the example just mentioned, j is the value node and $K = \{act, info\}$. The *query-processing task* is to arrange the two QPN operations in a sequence such that the relations of K on j are direct after executing the transformation. The qualitative relation of K on j is said to be direct if the following conditions are satisfied:

1. $pred(j) \subseteq K$, and
2. $j \notin pred^*(k)$, for all $k \in K$,

where $pred(n)$ is the set of direct predecessors of node n, and $pred^*(n)$ the set of all predecessors, direct and indirect (that is, nodes with a directed path to n).

Under these conditions, the influences and synergies incident on j directly encode the QPN's constraints on the probability distribution for j given K. The first condition rules out the presence of extraneous conditioning variables. The second ensures that any $k \in K$ not a direct predecessor of j is conditionally independent of the target given the rest of K [12, Lemma 4.1]. In the transformed network, $\delta_{k,j}$ denotes the sign of the qualitative influence from k to j in context $K - \{k\}$, and $\delta_{\{k^1,k^2\},j}$ the qualitative synergy of k^1 and k^2 on j in context $K - \{k^1, k^2\}$.

To process the query $j = v$, $K = \{act, info\}$ for the generic QPN, we could reduce the nodes *observe*, *costs*, and *benefits* (in any order), then reverse the link from θ to *info*, and finally reduce θ. At each step, the object of the operation is eligible for reduction or reversal in the intermediate network.

For any QPN query, in fact, there exists a sequence of truth-preserving operations that produces a network in which the relation of K on j is direct. This proposition follows from Shachter's analogous result for numeric influence diagrams [9], since the prerequisites for reductions (at most one successor) and reversals (must not create a cycle) are identical. For that matter, we can obtain a query-processing algorithm for QPNs simply by adapting a known inference procedure for numeric influence diagrams, replacing the probability revision equations with the QPN update rules of Section 4.

However, the qualitative probabilistic inference problem deserves special analysis, as it differs from its numeric counterpart in two significant areas. First, unlike numeric influence diagram evaluation, QPN query evaluation is sensitive to the sequence of operators applied to uncover the direct relation. Although the solutions are consistent across different sequences, the strength of the results may vary. For instance, one sequence of operations might produce $\delta_{k,j} = +$ while another results in ambiguity, $\delta_{k,j} = ?$.

Second, complexity considerations for the two types of model are quite distinct. In contrast to manipulation of numeric influence diagrams, the efficiency of QPN inference is insensitive to the cardinality of the outcome sets for each variable. (In fact, qualitative models need not even specify these outcome sets.) To measure the complexity of a QPN query-processing algorithm, we merely count the number of sign additions and multiplications necessary to perform the required network transformation.

In the following sections, I explore these issues and their implications for QPN inference algorithms. Note that many of the results presented are independent of the particular probabilistic definitions for qualitative influences and synergies specified above; they are valid for any qualitative relations that combine according to the algebra of Section 4. (However, there are conditions for which these definitions are *necessary* for the kind of qualitative inference we desire [12].)

6 Information Loss

A numeric influence diagram represents the unique joint probability distribution over the variables in the network. The reduction and reversal operators preserve information, producing an influence diagram representing the unique projection of the original distribution onto the new variable set.

Because qualitative relations enforce inequalities on conditional probabilities rather than unique numeric values, a QPN represents a *set* of probability distributions consistent with the constraints. QPN transformations are sound in that all qualitative relations in the modified network are entailed by the original constraints. However, graphical inference is not *complete*; there may be distributions consistent with the modified network that fail to satisfy the original qualitative relations.

For example, suppose a and c both have positive links to d and are marginally independent. Reversing the link from c to d yields a QPN with a "?" link from a to c, by (5). The modified network no longer captures the unconditional independence of a and c. Worse, if we reverse the link back using the same update rule, the "?" remains. Though the two reversals should ideally be inverse operations, applying them in sequence considerably weakens the network's constraint on the joint probability distribution.

Unfortunately, we cannot remedy this by strengthening the individual QPN update rules (4–6); each produces the strongest constraint expressible within the qualitative probability algebra. The problem is that the set of qualitative signs $\{+, -, 0, ?\}$ is not closed under the network transformation operators. In the example above, an ambiguous relation ("?"—the vacuous constraint) is introduced because no other qualitative conclusion is valid.

The actual class of distributions consistent with the original qualitative relations is not characterized by any of the qualitative signs available. (Distributions allowed by "?" but not consistent with the original constraints are analogous to the spurious solutions produced by qualitative reasoning mechanisms in general [4].) Not surprisingly, a similar problem arises in other representations admitting partial information about probability distributions. For instance, Fertig and Breese found that the class of lower probability functions is not closed under influence diagram transformations [3].

The following result states that the potential for information loss in reversal is in fact realized whenever non-degenerate predecessors exist—in other words, whenever there is information to lose.[5]

Theorem 6.1 *Information is lost in reversing the link from c to d unless all predecessors of c or d have "?" influences to both nodes and "?" synergies with any other predecessors.*

The following result establishes that information can be lost in reduction as well.

Theorem 6.2 *Information is lost in reducing c from the network iff there are nodes a and d such that:*

- *At least one of a and d is non-binary, and*
- $\delta_{a,d} \in \{+, -\}$, *and*
- $\delta_{a,c} \otimes \delta_{c,d} = \ominus\, \delta_{a,d}$,

where $\ominus$ is sign negation.

When the direct link between a and d is of opposite sign from the path through c, $S^{?}(a,d)$ is the result even though not all distributions $F_{d|a}$ are consistent with the original constraints.

7 Minimizing Ambiguity

7.1 Order Dependence

Given the prospect of information loss, processing queries by graphical manipulation may not lead to the strongest possible conclusions about the qualitative relations among variables of interest. Moreover, different strategies (sequences of operations) for processing a given query may differ in information lost and consequently yield solutions of varying strength.

Consider the QPN of Figure 4. Two simple transformations for revealing the direct relation of $K = \{k^1, k^2\}$ on j are:

[5] Proofs of these results are provided in Appendix A.

1. reverse $y \to j$; reduce y; reverse $j \to k^1$, and
2. reverse $y \to k^1$; reduce y.

Both procedures yield the relation $S^-(k^1, j)$. The first, however, is ambiguous about the relation between k^2 and j, whereas the second yields the more precise influence S^+. Note that both are valid; the transformation rules are sound probabilistic inferences. However, the conclusions returned by the first procedure are weakened by the information loss incurred upon reversal of the link from y to j.

qt2.ps

Figure 4: The choice of links to reverse before reducing y determines the strength of the final result.

Given a QPN and a query, we are clearly interested in finding the transformation sequence that produces the strongest conclusions. If computational efficiency were not an issue, we could perform all possible transformations, returning the strongest qualitative relations found among all variables of interest. Such an approach straightforwardly implemented, however, would require time exponential in the size of the network. Since each individual query-processing transformation can be applied in polynomial time, an algorithm based on a single operator sequence would be preferred. Ideally, we would like such an algorithm to apply the sequence of operators minimizing ambiguity in the end result.

The problem, then, is to find a method to compute this ambiguity-minimizing sequence, if it exists, given a QPN and a query. Although it would be convenient to derive this transformation from structure alone, it is not possible to do so, as the transformation yielding the strongest results may depend on the signs of the qualitative relations in the network. ("Structure" distinguishes only between zero and nonzero influences.)

The network of Figure 5 demonstrates the sign-dependence of ambiguity minimization. Processing a query with $K = \{k^1, \ldots, k^4\}$ requires the reduction of y from the network along with some reversals. If $\delta = +$, the transformation sequence

reverse $j \to y$; reverse $y \to k^3$; reverse $y \to k^4$; reduce y

yields the network with $S^+(k^i, j)$, $i = 1, \ldots, 4$. If $\delta = ?$, the same sequence results in "?" influences from each k^i to j. The ambiguity is spurious for k^1 and k^2, however, as revealed by the transformation

reverse $y \to k^3$; reduce y; reverse $j \to k^3$; reverse $j \to k^4$.

qt33.ps

Figure 5: The optimal transformation sequence depends on whether $\delta = ?$.

The latter sequence, on the other hand, returns $S^?(k^3, j)$ when $\delta = +$. In fact, no sequence of reductions and reversals applied to this network yields the strongest results in both cases.

The sign-dependence of ambiguity minimization is troubling, because in general determining the sign of a particular relationship could require query processing for an arbitrary sub-network. Nevertheless, at present it is an open question whether an efficient algorithm exists for minimizing ambiguity taking the signs of qualitative relations into account. This is true even for the special case where all influences are "?" links. Smith considers the problem of maintaining information about conditional independence under graphical transformations, concluding that the problem of minimizing spurious ambiguity remains open [10].

The existence of an ambiguity-minimizing sequence is itself uncertain. There may be cases (I know of none but have no proof of their impossibility) where no single sequence of transformations produces results as strong as the union of all transformations. And even if an ambiguity-minimizing sequence exists in this sense, it remains possible that stronger results could be achieved using other transformation operators or perhaps an entirely different inference method.

7.2 Transformation Invariants

Despite these unanswered questions, there is still much we can say about the inferential properties of various query-processing strategies. The following propositions characterize the legality of certain patterns in sequences of operations and establish invariants among classes of sequences.

Proposition 7.1 *Reducing a node does not negate the eligibility of any other node for reduction nor any link for reversal, but may establish such eligibility. Reversing an arc from c to d negates the eligibility of both nodes' predecessors for reduction, possibly negates d's eligibility, and possibly renders c eligible. All links to c become ineligible for reversal, some links to d may become eligible, and some other links may become ineligible.*

Proposition 7.2 *If two reductions, or a reduction and a reversal, can legally be applied in either order, the resulting networks are identical. This is not true for two reversals unless the nodes involved are disjoint.*

A corollary of this result is that node *removals* (reversal of all outgoing links followed by reduction) are order-invariant if the nodes are not directly connected and share no successors. The next proposition characterizes a less restrictive form of invariance holding for reductions.

Proposition 7.3 *Any sequence consisting exclusively of reductions is equivalent to any other legal sequence of the same reductions.*

Proposition 7.4 *Reducing a node c is equivalent to reversing its outgoing link (if any), then reducing it.*

This last proposition is not surprising considering the correspondence between update rules for the two operations (compare (2) and (3) with (5) and (6)). It implies that we can simplify matters by only considering reduction of nodes with no successors. Since reducing these *barren nodes* [8] entails no loss of information (by Theorem 6.2) and does not restrict further operations (Proposition 7.1), these may as well be performed as early as possible. Propositions 7.5 and 7.6 generalize this conclusion.

Proposition 7.5 *To process a query (j, K), any node y without a directed path to j or some $k \in K$ can be summarily spliced from the network without updating the remaining links.*

Proposition 7.6 *Any node may be reduced as soon as it is eligible without increasing the ambiguity of the transformation.*

Treating the reductions as automatic, the transformation construction task becomes essentially a matter of selecting the best sequence of reversals. The remaining proposition determines this best sequence for a special case.

Proposition 7.7 *If a node $y \in pred(j)$ is to be reduced after reversing all but one outgoing link, the link not reversed should be $y \rightarrow j$.*

The value of these propositions is that they provide constraint on the set of minimally ambiguous transformations. One can use them to generate a single admissible transformation, or within a search procedure to prune the space of operator sequences that need be considered. For example, Proposition 7.6 dictates that the query-processing algorithm should greedily reduce nodes not in $K \cup \{j\}$, whereas Proposition 7.3 establishes that a search algorithm may neglect transformations differing only in reduction order from one to be performed.

7.3 Upper Bounds

Because the coverage of these propositions is incomplete, graphical inference still requires heuristic choice of operators and/or a combinatorial search. Fortunately, we can often limit the search by computing bounds on the optimal query-processing strategy. The following results characterize some situations where isolated features of the network can dictate the best results achievable by any sequence of transformation operations.

Definition 3 *A* free path *from x to y with respect to K is a sequence N of nodes indexed by $(1, \ldots, m)$ such that*

- $1 \leq z \leq m$,
- $i \neq i' \Rightarrow N(i) \neq N(i')$,
- $\delta_{N(i),N(i-1)} \neq 0$, *for all* $i \in \{2, \ldots, z\}$,
- $\delta_{N(i),N(i+1)} \neq 0$, *for all* $i \in \{z, \ldots, m-1\}$, *and*
- $N(1) = x$, $N(m) = y$, $N(i) \notin K$ *for* $1 < i < m$.

Node $N(z)$ is called the pivot *of N.*

In other words, x and y are connected by a series of distinct nonzero links with no pair pointing into the same node and no links traversing nodes in K. Note that the node indexed z could be x or y, in which case there is a directed path between the two nodes.

Proposition 7.8 *Consider a network G with a free path N from j to k, $k \in K$. Any transformation of G processing the query (j, K) will result in $S^{\delta}(k, j)$, where*

$$\delta = \left(\bigotimes_{2 \leq i \leq z} \delta_{N(i),N(i-1)} \otimes \bigotimes_{z \leq i \leq m-1} \delta_{N(i),N(i+1)} \right) \oplus \delta_G,$$

and δ_G depends on the rest of the network.

Proposition 7.8 establishes an *upper bound* for queries based on a straightforward computation. Let δ_N be the sign product of the influences along a free path N. By the proposition, the exact result of the query is $\delta_N \oplus \delta_G$ for some δ_G. By the properties of $\oplus$, this value can be no "stronger" than δ_N in the ordering of qualitative signs (zero provides the strongest constraint; "?" the weakest). Let $paths(x, y, K)$ be the set of all free paths from x to y with respect to K. Then an upper bound for the $\delta_{k,j}$ upon processing the query (j, K) can be found by computing

$$\bigoplus_{N \in paths(j,k,K)} \delta_N.$$

In particular, finding a free path N with $\delta_N = ?$ (indeed, finding a free path with any "?" link) indicates that "?" is an upper bound for the relation of interest. The same is entailed by the existence of two paths N_1 and N_2 such that $\delta_{N_1} = +$ and $\delta_{N_2} = -$. In these cases, the bound is exact, as ambiguity is the weakest qualitative relation. An upper bound of $+$, in contrast, entails only that the result is in the set $\{+, ?\}$.

The following propositions describe two general situations where ambiguity is inevitable, even if the original network contains no "?" links.

Proposition 7.9 *Suppose the network contains free paths N_1 from j to k' and N_2 from k to k', $k, k' \in K$, with pivots indexed by z_1 and z_2, such that*

1. (disjoint prefix) $N_1(i_1) \neq N_2(i_2)$ *for all* $i_1 \leq z_1$ *and* $i_2 \leq z_2$, *and*

2. (convergence) *both N_1 and N_2 end with links* into k' *(that is, $N_1(z_1) \neq k' \neq N_2(z_2)$).*

Then any transformation of this network processing the query (j, K) will result in $S^?(k, j)$.

Proposition 7.10 *Suppose the network contains two free paths from j to k, $k \in K$, denoted N_1 and N_2 with corresponding pivots indexed by z_1 and z_2, such that $N_1(z_1) \notin N_2$ and $N_2(z_2) \notin N_1$. Then any transformation of this network processing the query (j, K) will result in $S^?(k, j)$.*

7.4 More Open Questions

The bounds given by the propositions above are not generally sharp, and do not cover all situations. Further transformation principles are necessary to select which reversal to perform in situations where several are available. Experience with various algorithms suggests that upstream reversals are generally preferred (since information loss tends to get propagated upstream anyway); however, we currently lack precise results to this effect.

It remains to be seen whether optimal algorithms can be identified for special cases of QPNs and queries. For example, queries with singleton K (that is, the relation of k on j given nothing else) may be more amenable to analysis than the general problem. One property commonly exploited in numeric probabilistic networks, single-connectivity, does not appear to offer much advantage. Note that the networks of Figures 4 and 5 are both singly connected.

8 Algorithmic Complexity

As mentioned above, any query can be processed in time polynomial in the size of the network. This result follows from the analysis of Rege and Agogino's "topological" solution of numerical influence diagrams [6], with an extra quadratic factor for updating synergies. In fact, any non-redundant transformation sequence (one without multiple reversals of the same link) can be applied in polynomial time.

This conclusion highlights the disparity in computational properties between qualitative and numeric influence diagrams. For the latter, complexity is highly sensitive to the size of each variable's value set (see, for example, the analysis of Ezawa [2]), a factor irrelevant in the qualitative realm. For some intractable numeric diagrams (those solvable by reductions alone—including the 3SAT construction used by Cooper [1] to demonstrate the NP-hardness of numeric query processing), the efficient QPN transformation is guaranteed to be minimally ambiguous.

The straightforward comparison is misleading, however, because graphical inference in QPNs is incomplete. Although determining qualitative relations is often easier than determining precise numeric relations, this is not necessarily true in the general case. A more definitive statement awaits further results on minimally ambiguous transformations, the complexity of transformation search, and the abstract qualitative probabilistic inference problem.

9 Summary

QPNs provide a representation for the qualitative relationships among variables without requiring excessively precise assessments. Like numeric influence diagrams, QPNs support inference via graphical manipulations. Because qualitative models incompletely specify the joint distribution among variables, information is not always preserved by these manipulations. In designing a query-processing algorithm, minimizing qualitative ambiguity is of paramount concern.

I have established several properties of QPN transformations that enhance the strength of derived relations or limit the search necessary to ensure minimal ambiguity. Unfortunately, there is no procedure for deriving a minimally ambiguous transformation sequence based solely on the topology of a QPN.

As indicated above, several open questions remain regarding the nature and even existence of optimal query-processing algorithms. In addition to these, several issues in qualitative probabilistic reasoning are worthy of further investigation. For example, inference methods based on macro- or entirely new transformation operators may offer advantages over sequences of reductions and reversals. Inference approaches not based on graphical transformations should also be explored. Finally, extensions of the qualitative representation itself ([11, Section 7.5]) will necessitate continued work on inferential aspects of qualitative probabilistic networks.

Acknowledgments

The suggestions of Jack Breese, Eric Hanson, Elisha Sacks, Ross Shachter, and Kate Unrath improved the quality of this paper.

A Proofs

The notation for probability distributions follows [12].

Theorem 6.1 *Information is lost in reversing the link from c to d unless all predecessors of c or d have "?" influences to both nodes and "?" synergies with any other predecessors.*

Proof First we must establish that no information is lost when the stated condition is met. If all predecessors have "?" influences and synergies on both c and d, the effect of two reversals is to leave the network unchanged. Thus, the QPN after two reversals admits the same set of joint distributions. Since reversal is a sound inference operator (all distributions consistent with the model prior to reversal are also consistent post-reversal), each individual reversal must also have preserved the set of consistent distributions.

To demonstrate information loss when the conditions are not met, I construct spurious distributions corresponding to each distinct case. In doing so, I simplify matters by taking c and d to be binary. This generalizes directly to multi-valued variables by constructing distributions over the larger domains with the probability mass allocated according to some binary criterion.

case 1: $\delta_{a,c} \neq ?$. Suppose that $\delta_{a,c} = \delta_{c,d} = +$ (the other sub-cases are analogous). Thus, in the original network, $a_1 > a_2$ implies

$$\Pr(C|a_1 x) \geq \Pr(C|a_2 x), \tag{7}$$

for all contexts x (instantiations of other predecessors of c). The reversal introduces d as a predecessor of c.

$$\Pr(C|a_i x) = \Pr(C|a_i D x)\Pr(D|a_i x) + \Pr(C|a_i \bar{D} x)\Pr(\bar{D}|a_i x). \tag{8}$$

Let $\alpha_i = \Pr(D|a_i x)$. Consider a distribution with

$$\begin{aligned}
\Pr(C|a_1 D x) &= \alpha_2, \\
\Pr(C|a_1 \bar{D} x) &< \alpha_2, \\
\Pr(C|a_2 D x) &> \max(\alpha_1, \frac{\alpha_2(1-\alpha_1)}{1-\alpha_2}), \text{ and} \\
\Pr(C|a_2 \bar{D} x) &= \frac{\alpha_2(1-\alpha_1)}{1-\alpha_2}.
\end{aligned}$$

Such a distribution satisfies the both of the post-reversal constraints ($\delta''_{d,c} = +$ and $\delta''_{a,c} = ?$), as $Pr(C|aDx) > \Pr(C|a\bar{D}x)$ for either value of a. However, applying these inequalities to the expansion of $\Pr(C|a_i x)$ (8), we find that

$$\begin{aligned}\Pr(C|a_1 x) &< \alpha_2\alpha_1 + \alpha_2(1-\alpha_1), \text{ while}\\ \Pr(C|a_2 x) &> \alpha_1\alpha_2 + \frac{\alpha_2(1-\alpha_1)}{1-\alpha_2}(1-\alpha_2).\end{aligned}$$

After simplifying, the two inequalities directly contradict the original constraint based on $\delta_{a,c} = +$ (7). Thus, the QPN after reversal admits a strictly greater class of distributions than the original.

case 2: $\delta_{a,d} \neq ?$. As a representative sub-case, suppose $\delta_{a,d} = \delta_{c,d} = +$. These original constraints dictate that $\Pr(D|a_1 Cx) \geq \Pr(D|a_2 Cx)$ for $a_1 > a_2$. Using Bayes's theorem,

$$\Pr(D|a_i Cx) = \frac{\Pr(C|a_i Dx)\Pr(D|a_i x)}{\Pr(C|a_i Dx)\Pr(D|a_i x) + \Pr(C|a_i \bar{D}x)\Pr(\bar{D}|a_i x)}.$$

Suppose that

$$\frac{\Pr(C|a_1 Dx)}{\Pr(C|a_1 \bar{D}x)} = \frac{\Pr(C|a_2 Dx)}{\Pr(C|a_2 \bar{D}x)} = \alpha.$$

This is consistent with the constraints after reversal, as long as $\alpha \geq 1$ (to satisfy $\delta''_{d,c} = +$). Then

$$\Pr(D|a_i Cx) = \frac{\alpha \Pr(D|a_i x)}{\alpha\Pr(D|a_i x) + 1 - \Pr(D|a_i x)}. \tag{9}$$

The "?" link from a to d allows the possibility that $\Pr(D|a_1 x) < \Pr(D|a_2 x)$, which, together with (9), would imply that

$$\Pr(D|a_1 Cx) < \Pr(D|a_2 Cx),$$

violating the original constraint.

case 3: $\delta_{\{a,b\},c} \neq ?$. Assume that both a and b have "?" influences on both c and d; the other possibilities are covered by the cases above. Further, let us take $\delta_{\{a,b\},c} = \delta_{c,d} = +$ (similar arguments will work for the other combinations). The positive synergy implies

$$\Pr(C|a_1 b_1 x) - \Pr(C|a_2 b_1 x) \geq \Pr(C|a_1 b_2 x) - \Pr(C|a_2 b_2 x), \tag{10}$$

for $a_1 > a_2$ and $b_1 > b_2$. After reversal, d becomes a predecessor of c and there is no synergy constraint among these variables. Consider a joint distribution for $\{a,b,c,d\}$ where $\Pr(a_1, b_2, C, D) = 2/17$ and $\Pr(a,b,c,d) = 1/17$ for all other values of the variables. Such a distribution is consistent with the post-reversal constraints ($\delta''_{d,c} = +$) but fails to satisfy (10).

case 4: $\delta_{\{a,b\},d} \neq ?$. We need only cover cases where a and b have "?" influences on both c and d, and "?" synergy on c. Suppose as a representative sub-case that $\delta_{\{a,b\},c} = \delta_{c,d} = +$. Thus, we must satisfy

$$\Pr(D|a_1 b_1 cx) - \Pr(D|a_2 b_1 cx) \geq \Pr(D|a_1 b_2 cx) - \Pr(D|a_2 b_2 cx).$$

However, the distribution of case 3 also violates this constraint, though it is permitted by the post-reversal network.

case 5: $\delta_{\{a,c\},d} \neq ?$. For $\delta_{\{a,c\},d} = \delta_{c,d} = +$, the distribution

$$\begin{aligned} \Pr(a_1, \bar{C}, D) &= 2/9, \\ \Pr(a, c, d) &= 1/9, \text{ all other values} \end{aligned}$$

is a spurious consequence of reversal just as in the previous two cases.

Since these cases exhaust the condition of the theorem, reversal in all but degenerate situations necessarily admits spurious probability distributions. □

Theorem 6.2 *Information is lost in reducing c from the network iff there are nodes a and d such that:*

- *At least one of a and d is non-binary, and*
- $\delta_{a,d} \in \{+, -\}$, *and*
- $\delta_{a,c} \otimes \delta_{c,d} = \ominus\, \delta_{a,d}$,

where $\ominus$ is sign negation.

PROOF

(if): I demonstrate information loss for a case with a non-binary, $\delta_{a,c} = \delta_{a,d} = +$, and $\delta_{c,d} = -$. The other combinations satisfying the conditions can be argued analogously. After reducing c, the QPN provides no constraint on the probabilistic relation between a and d, as $\delta'_{a,d} = ?$.

When c and d are binary, the probability distribution for d given a can be written as

$$\Pr(D|a_i x) = \Pr(D|a_i C x)\Pr(C|a_i x) + \Pr(D|a_i \bar{C} x)\Pr(\bar{C}|a_i x),$$

or, with symbols substituted for corresponding terms:

$$\Pr(D|a_i x) = \alpha_i \beta_i + \gamma_i (1 - \beta_i). \tag{11}$$

Consider an ascending sequence of values for a: $a_1 < a_2 < a_3 < a_4$. I show that the distribution

$$\Pr(D|a_i x) = 2/3, 1/3, 2/3, 1/3 \text{ for } i = 1, 2, 3, 4 \tag{12}$$

is inconsistent with the original constraints, though it is allowed by the reduced network.

Since $\alpha_i \leq \gamma_i$ ($\delta_{c,d} = -$), it follows from (11) that

$$\alpha_i \leq \Pr(D|a_i x) \leq \gamma_i. \tag{13}$$

Therefore $\gamma_1 \geq 2/3$. And since both α_i and γ_i are nondecreasing in i (by $\delta_{a,d} = +$), γ_2 is also at least $2/3$. Given that α_i is nonnegative, we can use (11) to bound β_i from below:

$$\beta_i \geq 1 - \frac{\Pr(D|a_i x)}{\gamma_i}.$$

Thus, $\beta_2 \geq 1 - (1/3)/(2/3) = 1/2$. The inequality holds for β_3 as well, as $\delta_{a,c} = +$. We use this lower bound in turn to derive a bound on α. Rearranging (11), we get

$$\alpha_i = \gamma_i - \frac{\gamma_i - \Pr(D|a_i x)}{\beta_i}.$$

Noting that $2/3 \leq \gamma_3 \leq 1$ and $\beta_3 \geq 1/2$, we can use this to deduce that $\alpha_3 \geq 1/2$. As α_i is nondecreasing, α_4 is also bounded from below by $1/2$. But this contradicts the first inequality of (13), as $\Pr(D|a_4 x) = 1/3$. Therefore the distribution of (12) violates the original constraints, and information has been lost in reducing c.

(only if): I show that each of the conditions is *necessary* for information loss by establishing that violating any one of them guarantees that all distributions consistent post-reduction are also consistent pre-reduction.

condition 1: one of a and d non-binary. As a representative case, take $\delta_{a,c} = \delta_{a,d} = +$ and $\delta_{c,d} = -$. For simplicity assume c is binary, though it would do just as well to consider it multi-valued with all intermediate values having zero probability. After reducing c, all probability distributions for a and d are consistent with the network. Since both variables are binary, the post-reduction distribution is fully specified by $\Pr(D|Ax)$ and $\Pr(D|\bar{A}x)$. These probabilities are related to the original constraints as follows:

$$\begin{aligned} \Pr(D|Ax) &= \Pr(D|ACx)\Pr(C|Ax) + \Pr(D|A\bar{C}x)\Pr(\bar{C}|Ax) && (14) \\ \Pr(D|\bar{A}x) &= \Pr(D|\bar{A}Cx)\Pr(C|\bar{A}x) + \Pr(D|\bar{A}\bar{C}x)\Pr(\bar{C}|\bar{A}x). && (15) \end{aligned}$$

The question is whether all values for the LHS probabilities can be obtained from some assignment to the RHS terms that is consistent with the specified qualitative influences.

Consider the assignments

$$\begin{aligned}\Pr(D|\bar{A}Cx) &= 0\\ \Pr(D|A\bar{C}x) = \Pr(C|Ax) &= 1.\end{aligned}$$

These extreme values guarantee that any probabilities assigned to the other terms will satisfy the inequalities entailed by $\delta_{a,c}$, $\delta_{a,d}$, and $\delta_{c,d}$. In addition, let $\Pr(C|\bar{A}x) = 0$. Then the equations (14) and (15) reduce to:

$$\begin{aligned}\Pr(D|Ax) &= \Pr(D|ACx)\\ \Pr(D|\bar{A}x) &= \Pr(D|\bar{A}\bar{C}x).\end{aligned}$$

Since the RHS terms are unconstrained by the other assignments, the qualitative influences, or each other, any distribution for d given a and x alone is consistent with the original constraints.

condition 2: $\delta_{a,d} \in \{+,-\}$. If $\delta_{a,d} = 0$, the distributions consistent with the network after reducing c are those satisfying the constraint imposed by $\delta_{a,c} \otimes \delta_{c,d}$. Suppose that $\delta_{a,c} = +$. Then the qualitative influence of a on d after reduction is simply $\delta_{c,d}$. One cumulative distribution satisfying the constraint $\delta_{a,c} = +$ is[6]

$$F_c(c_0|ax) = \begin{cases} 1 & \text{if } c_0 \geq a \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

Since a and d are conditionally independent given c ($\delta_{a,d} = 0$), the distribution for d given a after reduction is:

$$F_d(d_0|ax) = \int F_d(d_0|c_0x)dF_c(c_0|ax).$$

Substituting in (16), we get

$$F_d(d_0|ax) = F_d(d_0|cx)|_{c=a},$$

where the distribution for d given c is evaluated at $c = a$. Thus, the distributions allowed after reduction for a and d are equivalent to those allowed by $\delta_{c,d}$, which is the qualitative influence obtained by reduction.

The other violation of condition 2 occurs when $\delta_{a,d} = ?$. In this case the distribution post-reduction is unconstrained. Indeed, no constraint is warranted, as any distribution is achievable. Regardless of $\delta_{c,d}$, the possibility that c and d are independent is consistent, in which case the relation between a and d is constrained only by $\delta_{a,d}$, which is to say not at all.

[6] For (16) to be well-defined, variables a and c must have identical domains. A more complicated argument is required when the domains differ.

condition 3: $\delta_{a,c} \otimes \delta_{c,d} = \ominus\ \delta_{a,d}$. Suppose condition 3 were violated while condition 2 were not. Then $\delta_{a,d}$ agrees with $\delta_{a,c} \otimes \delta_{c,d}$ and therefore persists as the qualitative relation between a and d after reducing c. As above, since conditional independence of c and d is always consistent, the distribution post reduction can be no more constrained than its pre-reduction counterpart. Thus, no information is lost.

□

Proposition 7.1 *Reducing a node does not negate the eligibility of any other node for reduction nor any link for reversal, but may establish such eligibility. Reversing an arc from c to d negates the eligibility of both nodes' predecessors for reduction, possibly negates d's eligibility, and possibly renders c eligible. All links to c become ineligible for reversal, some links to d may become eligible, and some other links may become ineligible.*

Proof

reduction/reduction: Recall that a node is eligible for reduction iff it has at most one successor. After reducing node c, c's predecessors lose c as a successor but inherit c's successor if there is one. Since every node's successor list either stays the same or decreases by one, previously eligible nodes remain so. If c has no successors, any predecessor linked only to c becomes eligible for reduction.

reduction/reversal: A link is eligible for reversal if doing so would not create a cycle. In other words, one can reverse the link from c to d if there is no indirect directed path connecting the two. Since reducing a node creates no new paths, the operation cannot spoil reversibility for any links. (Of course, links incident on the reduced node are eliminated by the operation.) If the only path denying the reversibility of $c \to d$ happens to be $c \to e \to d$, however, reducing e would render $c \to d$ eligible for reversal.

reversal/reduction: Upon reversing the link from c to d, each node gains the other's predecessors. Each of these predecessors thus has at least two successors and hence is ineligible for reduction. Since d gains c as a successor, it becomes ineligible if it previously had exactly one. Similarly, c loses d as a successor, so it becomes eligible if previously it had exactly two. Eligibility for other nodes is unaffected.

reversal/reversal: After the reversal, any predecessor of c has an indirect path to c via d, thus ruling out reversal of its link to c. Pre-reversal links to d are rendered eligible by the reversal just in case the only indirect path connecting its edges was through c. Links connecting other nodes may become ineligible, for example if one of the new predecessor links completes a path through c. No other links may become eligible for reversal, as any indirect paths canceled by the reversal are reinstated by inheritance of predecessors.

$\square$

Proposition 7.2 *If two reductions, or a reduction and a reversal, can legally be applied in either order, the resulting networks are identical. This is not true for two reversals unless the nodes involved are disjoint.*

PROOF

two reductions: Suppose we are interested in the qualitative relation between a and d after reducing c^1 and c^2. Since the QPN is acyclic, assume without loss of generality that $d \notin \mathit{pred}^*(a)$ and $c^2 \notin \mathit{pred}^*(c^1)$. With these constraints, the only configurations of interest have the nodes to be reduced between a and d—otherwise the reductions will not affect the relation of interest. Thus, the relevant network fragment has the general form of Figure 6. Some of the links shown may be optional (that is, $\delta = 0$). In fact, at least one of $\delta_{1,2}$ and $\delta_{1,d}$ must be zero in order to satisfy the requirement that c^1 be legally reducible.

reduce2.ps

Figure 6: General QPN fragment for analyzing the order-invariance of reducing c^1 and c^2.

Suppose we reduce c^1, then c^2. Let $\delta^i_{y,z}$ denote the qualitative influence of y on z after reducing c^i. Applying the update rule for reduction (2),

$$\begin{aligned}\delta^1_{a,d} &= \delta_{a,d} \oplus (\delta_{a,1} \otimes \delta_{1,d})\\ \delta^1_{a,2} &= \delta_{a,2} \oplus (\delta_{a,1} \otimes \delta_{1,2})\\ \delta^1_{2,d} &= \delta_{2,d}.\end{aligned}$$

Reducing c^2 yields

$$\delta^{1,2}_{a,d} = \delta^1_{a,d} \oplus (\delta^1_{a,2} \otimes \delta^1_{2,d}).$$

After substituting the expressions for δ^1 above,

$$\begin{aligned}\delta^{1,2}_{a,d} &= \delta_{a,d} \oplus (\delta_{a,1} \otimes \delta_{1,d}) \oplus ([\delta_{a,2} \oplus (\delta_{a,1} \otimes \delta_{1,2})] \otimes \delta_{2,d})\\ &= \delta_{a,d} \oplus (\delta_{a,1} \otimes \delta_{1,d}) \oplus (\delta_{a,2} \otimes \delta_{2,d}) \oplus (\delta_{a,1} \otimes \delta_{1,2} \otimes \delta_{2,d}).\end{aligned} \tag{17}$$

If instead c^2 were reduced first, the intermediate results would be

$$\begin{aligned}\delta^2_{a,d} &= \delta_{a,d} \oplus (\delta_{a,2} \otimes \delta_{2,d})\\ \delta^2_{a,1} &= \delta_{a,1}\\ \delta^2_{1,d} &= \delta_{1,d} \oplus (\delta_{1,2} \otimes \delta_{2,d}).\end{aligned}$$

Subsequently reducing c^1 yields

$$\begin{aligned}\delta^{2,1}_{a,d} &= \delta^2_{a,d} \oplus (\delta^2_{a,1} \otimes \delta^2_{1,d}) \\ &= \delta_{a,d} \oplus (\delta_{a,2} \otimes \delta_{2,d}) \oplus (\delta_{a,1} \otimes \delta_{1,d}) \oplus (\delta_{a,1} \otimes \delta_{1,2} \otimes \delta_{2,d}),\end{aligned}$$

which is equivalent to the expression for $\delta^{1,2}_{a,d}$ above (17).

To complete the argument we need to establish that the invariance holds for synergies as well. Consider another node b with the same outgoing links as a in Figure 6. By an exercise similar to that above (though more complicated[7]), we can derive the general expression for the qualitative synergy of a and b on d after reducing c^1 followed by c^2:

$$\begin{aligned}\delta^{1,2}_{\{a,b\},d} = {} & \delta_{\{a,b\},d} \oplus (\delta_{\{a,b\},1} \otimes \delta_{1,d}) \oplus (\delta_{b,1} \otimes \delta_{\{a,1\},d}) \oplus (\delta_{a,1} \otimes \delta_{\{b,1\},d}) \\ & \oplus (\delta_{\{a,b\},2} \otimes \delta_{2,d}) \oplus (\delta_{\{a,b\},1} \otimes \delta_{1,2} \otimes \delta_{2,d}) \oplus (\delta_{b,1} \otimes \delta_{\{a,1\},2} \otimes \delta_{2,d}) \\ & \oplus (\delta_{a,1} \otimes \delta_{\{b,1\},2} \otimes \delta_{2,d}) \\ & \oplus (\delta_{b,2} \otimes \delta_{\{a,2\},d}) \oplus (\delta_{b,2} \otimes \delta_{a,1} \otimes \delta_{\{2,1\},d}) \oplus (\delta_{b,1} \otimes \delta_{1,2} \otimes \delta_{\{a,2\},d}) \\ & \oplus (\delta_{b,1} \otimes \delta_{1,2} \otimes \delta_{a,1} \otimes \delta_{\{2,1\},d}) \qquad (18) \\ & \oplus (\delta_{a,2} \otimes \delta_{\{b,2\},d}) \oplus (\delta_{a,2} \otimes \delta_{b,1} \otimes \delta_{\{2,1\},d}) \oplus (\delta_{a,1} \otimes \delta_{1,2} \otimes \delta_{\{b,2\},d}) \\ & \oplus (\delta_{a,1} \otimes \delta_{1,2} \otimes \delta_{b,1} \otimes \delta_{\{2,1\},d}). \qquad (19)\end{aligned}$$

Note that lines (18) and (19) are zero and thus drop out, since at least one of $\delta_{1,2}$ and $\delta_{1,d}$ (and therefore $\delta_{\{2,1\},d}$) must be zero.

It turns out that performing the calculation with c^2 reduced first results in the same synergy expression. Thus, $\delta^{1,2}_{\{a,b\},d} = \delta^{2,1}_{\{a,b\},d}$ and reduction is pairwise order-invariant.

reduction and reversal: Consider the operations to reduce y and to reverse $c \to d$. In order for the operations to be applicable in either order, it must be the case that:

- $y \neq c$ and $y \neq d$ (else the link would not exist post-reduction) and
- $y \notin \mathit{pred}(c)$ and $y \notin \mathit{pred}(d)$ (see Proposition 7.1).

Given these constraints, the updates for the two operations affect a disjoint set of qualitative relations. Hence, the results are identical when performed in either order.

two reversals: Reversals are not order-invariant in general. In fact, even the topology of the resulting network depends on the order reversals are performed. For example, suppose the only non-zero links in a network are $c^1 \to d$ and $c^2 \to d$. The orientation of the link

[7]The calculations required to verify these propositions were substantially aided by a program for manipulating QPNs with qualitative relations specified by symbolic expressions.

connecting the c^is after the two reversals depends on which is performed first. (The sign on the link is "?" in either case.)

If the two arcs reversed do not share a node, the two operations affect a disjoint set of relations. In this case, the results do not depend on the order of the reversals.

□

Proposition 7.3 *Any sequence consisting exclusively of reductions is equivalent to any other legal sequence of the same reductions.*

PROOF Let σ be a sequence of nodes such that it is legal to reduce any node in the sequence after all of its predecessors have been reduced. For convenience, index the nodes by their order in σ, so we can write $\sigma = (1, \ldots, m)$. Consider σ', another legal sequence of node reductions containing the same nodes as σ, though possibly in a different order. Without loss of generality, suppose that the first node of σ (node 1) is not first in σ' (if it is, we need only compare the tails of the sequences anyway). Thus, $\sigma' = (s, i, 1, t)$, where s and t are subsequences of nodes.

Now consider $\sigma'' = (s, 1, i, t)$, the sequence obtained from σ' by swapping nodes i and 1 in the order. First we must consider the the legality of σ'' as a sequence of reductions.

1. The subsequence s is obviously legal, as it constitutes the first part of the legal sequence σ'.

2. The legality of σ implies that it is legal to reduce node 1 *first*, that is, in the starting network. Therefore it must be legal to reduce node 1 after performing the sequence s of reductions, as reduction alone cannot eliminate a node's eligibility for reduction (Proposition 7.1).

3. Similarly, if it is legal to reduce node i after s (as in σ'), it must also be legal to reduce it after $(s, 1)$.

Thus, we have established that it is legal to reduce nodes $(s, 1, i)$ in that order. Proposition 7.2 dictates that, for any given network in which both orders are legal, reductions $(i, 1)$ and $(1, i)$ have the same result. Therefore, since the prefixes are identical, reduction sequences $(s, i, 1)$ and $(s, 1, i)$ leave the network in the same state. Appending t then obviously preserves the identity, and we have established that σ' and σ'' are equivalent.

By repeatedly swapping node 1 with its neighbor to the left, we can eventually produce a sequence in which it appears first. Because each step preserves both legality and equivalence, the resulting sequence yields results identical to σ'. Having done this, we can then proceed to move node 2 to the left in the same fashion, until it is second in the list. Repeating this for all nodes in the series, we achieve in the end the sequence σ, thus demonstrating that it is equivalent to σ'. Since σ and σ' were unrestricted, we have established the equivalence of all legal reduction sequences containing the same set of nodes.

□

Proposition 7.4 *Reducing a node c is equivalent to reversing its outgoing link (if any), then reducing it.*

PROOF Let d be the sole successor of c and let a and b be generic predecessors of c and/or d. (If there is no d the equivalence is trivial.) Compare the states of the network obtained by reducing c and reversing $c \to d$. The correspondence of (2) and (3) with (5) and (6) establish that $\delta'_{a,d} = \delta''_{a,d}$ and $\delta'_{\{a,b\},d} = \delta''_{\{a,b\},d}$. In contrast to the reduction, however, the reversal may also result in other updates to influences and synergies on c. But since these updates are rendered moot by subsequent reduction of the now-barren c, the ending states are identical. $\square$

Proposition 7.5 *To process a query (j, K), any node y without a directed path to j or some $k \in K$ can be summarily spliced from the network without updating the remaining links.*

PROOF By Theorem 6.2, no information is lost in reducing barren nodes. Furthermore, the network after reduction does not depend on the links, if any, into the barren node. We can establish the proposition by demonstrating that all nodes satisfying the stated conditions can be removed by a sequence consisting solely of barren node reductions.

Note that any barren node not in $\{j\} \cup K$ trivially satisfies the conditions for y in the proposition. Reducing all of these nodes has the same effect as summarily splicing them from the network. After these reductions, some other nodes may have become barren. Suppose we continue to reduce all barren nodes outside of $\{j\} \cup K$ until no more exist. At that point, all nodes meeting the conditions for y have been removed. To see this, observe that any y must either be barren or have a barren successor (since the network is acyclic). As the process ends only when there are no barren nodes not in $\{j\} \cup K$, y and all other nodes satisfying the proposition must have been removed. $\square$

Proposition 7.6 *Any node may be reduced as soon as it is eligible without increasing the ambiguity of the transformation.*

PROOF Let c be the node eligible for reduction, and d its predecessor. First, note that the only loss of information attributable to reduction is in case there is some a linked to both c and d such that the path from a to d via c and the direct link are of opposite signs (Theorem 6.2). In particular, no dependence information is lost. This, combined with the fact that reductions cannot nullify the eligibility of subsequent operations, implies that reducing c immediately is an optimal policy with respect to the information encoded in the topology of the network. Thus, we need consider only the potential introduction of spurious ambiguity due to premature reduction.

Next, let us examine the propagation of sign information in transformation operations. Upon reduction of c, the new qualitative influence is given by

$$\delta' = \delta_{a,d} \oplus (\delta_{a,c} \otimes \delta_{c,d}). \tag{20}$$

This information can be propagated to other relations via operations on adjacent nodes. Specifically, there are four ways δ' can have an impact on the signs of other relations.

1. Reduction of a, d, or a node that is a direct successor of a and a predecessor of d.

2. Reversal of $a \rightarrow d$.

3. Reversal of a link $x \rightarrow d$.

4. Reversal of a link $d \rightarrow x$.

In the case of reduction, δ' appears as one of the terms in a recursive invocation of (20), with the node names rebound. By the reasoning of Proposition 7.8, information lost in δ' can never be recovered by iteration of this formula. In a reversal of $a \rightarrow d$, all links into a are updated by the same formula, and the same argument applies. Links into d are converted to "?" regardless of δ', so the information loss had no adverse effect. The same is true of influences affected by $x \rightarrow d$ reversals: the $a \rightarrow d$ link is updated by simple chaining and $a \rightarrow x$ is indiscriminately made ambiguous. Upon reversing $d \rightarrow x$, $a \rightarrow d$ is either "?" or unchanged, while $a \rightarrow x$ is updated as if d were reduced.

The updates above follow a general pattern. The actual sign of δ' matters only for *chaining* relations: links arranged in a head-to-tail adjacency. Therefore, we have established that the information lost in reduction does not propagate in any but the inevitable directions. Given that there is no loss of dependence information, therefore, nothing can be gained by postponing reduction. $\square$

Proposition 7.7 *If a node $y \in pred(j)$ is to be reduced after reversing all but one outgoing link, the link not reversed should be $y \rightarrow j$.*

PROOF Suppose y has outgoing links to each of x and j, and let b be an arbitrary node with a free path to y. By Proposition 7.5, we can assume without loss of generality that x has a directed path to j or some $k' \in K$. If the path is to j, then $y \rightarrow j$ is not reversible and the conclusion of the proposition follows by necessity. (If j has a directed path to x, then $c \rightarrow x$ is not reversible and the choice presumed by the Proposition does not apply.) Thus, x has a directed path to k'.

Note that both b and j have free paths to k' via y. At this point, Proposition 7.9 does not apply, as the two paths violate its disjoint prefix condition (y is the pivot of the path from j). Upon reversal of $y \rightarrow j$, however, the pivot is eliminated and this condition is satisfied. The implication of the reversal, then, is that any further query processing must result in a "?" link between b and j. This ambiguity is not inevitable for a reversal of $x \rightarrow j$; hence, this operation is preferred. $\square$

Proposition 7.8 *Consider a network G with a free path N from j to k, $k \in K$. Any transformation of G processing the query (j, K) will result in $S^{\delta}(k, j)$, where*

$$\delta = \left(\bigotimes_{2 \le i \le z} \delta_{N(i),N(i-1)} \otimes \bigotimes_{z \le i \le m-1} \delta_{N(i),N(i+1)} \right) \oplus \delta_G, \tag{21}$$

and δ_G depends on the rest of the network.

Proof We establish the result by examining the effect of reductions and reversals on the composition of the free path. The proof proceeds by induction on the number of operations required for the transformation. Take as the base case the set of transformations implemented by zero steps. These correspond to the degenerate free path where j is connected directly to k and the query is answerable by inspection of the original network. In this situation, the multiplicative chain (21) reduces to a single term $\delta_{j,k}$, and δ_G is zero. Now hypothesize that all transformations up to a specified size yield results of the form of (21). We show that any additional operations preserve that form, as all expressions arising off the main path can be additively factored out into the δ_G term.

Let us adopt the notation $N(i \pm 1)$ to denote the successor of $N(i)$ along path N, whose index depends on whether i is less than or greater than the pivot, z. Reduction of a node $w \notin N$ can affect a link from $N(i)$ to $N(i \pm 1)$ iff $N(i)$ is a predecessor of w and $N(i \pm 1)$ is its sole successor. In this case, the sign on the link becomes

$$\delta'_{N(i),N(i\pm1)} = \delta_{N(i),N(i\pm1)} \oplus (\delta_{N(i),w} \otimes \delta_{w,N(i\pm1)}). \tag{22}$$

By the inductive hypothesis, the result of the rest of the transformation has the form of (21), which we can write as

$$(\delta'_{N(i),N(i\pm1)} \otimes \delta_R) \oplus \delta_G, \tag{23}$$

taking δ_R to denote the product of links from the rest of the chain. We can plug in the right-hand side of (22) and multiply through, yielding

$$(\delta_{N(i),N(i\pm1)} \otimes \delta_R) \oplus [(\delta_{N(i),w} \otimes \delta_{w,N(i\pm1)} \otimes \delta_R) \oplus \delta_G],$$

which satisfies the required form, with the expression in brackets substituted for the previous δ_G.

Next consider the reduction of a node $N(h)$, $h > z$. We can establish that the new sequence $N' = (N(1), \ldots, N(h-1), N(h+1), \ldots, N(m))$ satisfies the conditions for a free path. First, note that all links are unchanged except for[8]

$$\delta'_{N'(h-1),N'(h)} = \delta_{N(h-1),N(h+1)} \oplus (\delta_{N(h-1),N(h)} \otimes \delta_{N(h),N(h+1)}). \tag{24}$$

[8]Because the indices are renumbered, node $N'(i)$ corresponds to $N(i+1)$ in the original sequence, for $i \ge h$.

Therefore, the free path is still fully connected by nonzero links, and the segment from $N(z)$ to $N(m)$ remains directed. Again by the inductive hypothesis, the query result takes the form of (23), with $i = h-1$. Substituting the expression from (24) yields

$$\left(\left[\delta_{N(h-1),N(h+1)} \oplus (\delta_{N(h-1),N(h)} \otimes \delta_{N(h),N(h+1)})\right] \otimes \delta_R\right) \oplus \delta_G.$$

Multiplying through results in

$$(\delta_{N(h-1),N(h)} \otimes \delta_{N(h),N(h+1)} \otimes \delta_R) \oplus [(\delta_{N(h-1),N(h+1)} \otimes \delta_R) \oplus \delta_G],$$

again satisfying the required form.

The same conclusion holds by symmetry for reduction of $N(h)$, $h < z$. The pivot $N(z)$ cannot be reduced because it has more than one successor.

By a similar argument, reversing a link not on the path preserves both the existence of the path and the required form, since the update formula for either type of predecessor influence (5) includes the original sign as an additive term. Suppose, then, that the link from $N(h)$ to $N(h+1)$ is reversed, for $h > z$. We can construct N' exactly as for the case of reduction. The reversal introduces a new link from $N(h-1)$ to $N(h+1)$, with sign computed as for the corresponding reduction (24). By the argument above, the form of the result is preserved for the new path N'. A symmetric argument is valid for reversing the link from $N(h)$ to $N(h-1)$, $h < z$.

The argument needs to be modified for reversing a link from the pivot itself. In this operation no links on the path are affected except the one being reversed, and this one maintains the same sign. After the reversal, however, the designation of pivot z must be transferred to the other end of the reversed link in order to maintain the condition for a free path. Since this path retains the same nodes and the same δs, the inductive step follows directly.

Because each operation preserves the target form (21) given that it holds for the rest of the transformation, the entire transformation must maintain this result. □

Proposition 7.9 *Suppose the network contains free paths N_1 from j to k' and N_2 from k to k', $k, k' \in K$, with pivots indexed by z_1 and z_2, such that*

1. (disjoint prefix) $N_1(i_1) \neq N_2(i_2)$ *for all $i_1 \leq z_1$ and $i_2 \leq z_2$, and*

2. (convergence) *both N_1 and N_2 end with links* into k' *(that is, $N_1(z_1) \neq k' \neq N_2(z_2)$).*

Then any transformation of this network processing the query (j, K) will result in $S^?(k, j)$.

Proof The proof is by induction on the lengths of N_1 and N_2. In the base case, $|N_1| = |N_2| = 1$: each path consists of a single direct link into k'. To process the query, the path from j to k' needs to be removed. Since k' cannot be reduced, this entails a reversal of the

$j \to k'$ link. If k is a predecessor of k' at the time of the reversal, the operation introduces a link of sign "?" from k to j. This holds regardless of the signs of the links into k' (as long as they are nonzero, a definitional property of free paths), by the reversal update rule (5). The only way k would not be a predecessor of k' at this time would be if the $k \to k'$ link were reversed first. In this case, a "?" link from j to k is necessarily introduced. In either event, there exists a free path of sign $\delta_N = ?$ from j to k, and therefore Proposition 7.8 implies that the result of the query must be $S^?(k, j)$.

For the inductive hypothesis, assume that the result is true for all paths with $|N_1| \leq n_1$ and $|N_2| \leq n_2$. Given a chain with $|N_1| = n_1 + 1$ or $|N_2| = n_2 + 1$, we show that any operation must have one of the following effects:

1. to preserve the existence of paths N_1 and N_2, of the same length, satisfying the conditions of the proposition,

2. to preserve the existence of such paths with shorter length, or

3. to introduce a free path from k to j with at least one "?" link.

In the first case, the situation is identical from the standpoint of our proposition and the process can be repeated. In the second, we can invoke the inductive hypothesis that the conclusion holds for paths of the shorter length. In the final case, we can invoke Proposition 7.8 directly, as in the base case, to establish the final result. Since the first case cannot be maintained through the entire sequence of transformations—the query cannot be fully processed in that state—eventually one of the other two must prevail, thus establishing the inductive argument.

First consider a node reduction. As in the argument for Proposition 7.8 above, this operation preserves the existence of free paths whether or not the node was previously on one of the paths. Similarly, it cannot alter the disjointness of the path prefixes or the convergence of their final links into k'. If the node to be reduced is on one of the paths, this path is thereby shortened, thus fulfilling the condition for case two above.

Reversing a link not on one of the paths cannot negate its existence. As argued above, reversing a link on a free path also preserves its existence, possibly at a length one node shorter. However, we must address special attention to the case where the link reversed is the final one in the chain, that into k'. Reversing this link does not obviously maintain the convergence condition of our proposition. Suppose we reverse the last link of N_1 (the analysis for N_2 is symmetric). There are two relevant subcases, depending on whether the immediate predecessor of k' on N_1 is the pivot node. Let y_1 be the predecessor of k' on N_1 and y_2 its counterpart on N_2. If $y_1 \neq N_1(z_1)$, then the reversal introduces a new link from y_1's predecessor into k' and a new path N_1' is formed with one node fewer than N_1. A new path N_2' is also formed if $y_1 = y_2$. Otherwise, if $y_1 = N_1(z_1)$, the reversal introduces a link of sign "?" from y_2 to y_1. (By the disjointness condition, the two nodes cannot be identical if one is a pivot.) With this link, y_2 has a directed path to k. Since the segment of N_2 from j to y_2 is a free path, so is the extended path from j to k via y_2. As noted above, the existence of a free path with one "?" link is sufficient to establish that the result of query processing is $S^?(k, j)$.

Because these cases exhaust the possibilities, conditions are maintained or progressed, and the inductive argument is concluded. □

Proposition 7.10 *Suppose the network contains two free paths from j to k, $k \in K$, denoted N_1 and N_2 with corresponding pivots indexed by z_1 and z_2, such that $N_1(z_1) \notin N_2$ and $N_2(z_2) \notin N_1$. Then any transformation of this network processing the query (j, K) will result in $S^?(k, j)$.*

Proof By the arguments above, the existence of the free paths is preserved by all operations, and the only operation that can affect the conditions on the pivot node are reversals of links emanating from that node. Consider a reversal of the link from $N_1(z_1)$ to $y_1 = N_1(z_1+1)$ (the other cases are symmetric). If $y_1 \notin N_2$, then the conditions of the proposition are preserved with a change in pivot to y_1. Suppose y_1 is on the path N_2. Since y_1 is also on N_1, we know it cannot be the pivot of N_2 and therefore it must have one predecessor, x_2, on that path. However, the reversal of $N_1(z_1) \rightarrow y_1$ creates (or updates) a link from x_2 to $N_1(z_1)$ with a "?" sign. Since x_2 has a directed path along N_2 to either j or k, and $N_1(z_1)$ has directed paths to both, there must be a free path from j to k with pivot x_2 that includes this new link. By Proposition 7.8, this is sufficient to entail ambiguity of the query. □

References

[1] Gregory F. Cooper. The computational complexity of probabilistic inference using belief networks. *Artificial Intelligence*, 42:393–405, 1990.

[2] Kazuo John Ezawa. *Efficient Evaluation of Influence Diagrams*. PhD thesis, Stanford University, June 1986.

[3] K. W. Fertig and J. S. Breese. Interval influence diagrams. In *Proceedings of the Workshop on Uncertainty in Artificial Intelligence*, pages 102–111, Windsor, ON, 1989.

[4] Benjamin J. Kuipers. The limits of qualitative simulation. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 128–136, August 1985.

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

[6] Ashutosh Rege and Alice M. Agogino. Topological framework for representing and solving probabilistic inference problems in expert systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 18:402–414, 1988.

[7] Sheldon M. Ross. *Introduction to Stochastic Dynamic Programming*. Academic Press, 1983.

[8] Ross D. Shachter. Evaluating influence diagrams. *Operations Research*, 34:871–882, 1986.

[9] Ross D. Shachter. Probabilistic inference and influence diagrams. *Operations Research*, 36:589–604, 1988.

[10] J. Q. Smith. Influence diagrams for statistical modelling. *The Annals of Statistics*, 17:654–672, 1989.

[11] Michael P. Wellman. *Formulation of Tradeoffs in Planning Under Uncertainty*. Pitman and Morgan Kaufmann, 1990.

[12] Michael P. Wellman. Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence*, 1990.

[13] Michael P. Wellman, Mark H. Eckman, Craig Fleming, et al. Automated critiquing of medical decision trees. *Medical Decision Making*, 9:272–284, 1989.