Qualitative Probabilistic Relational Models

Linda C. van der Gaag[1(✉)] and Philippe Leray[2]

[1] Department of Information and Computing Sciences, Utrecht University,
Princetonplein 5, 3584 CC Utrecht, The Netherlands
L.C.vanderGaag@uu.nl
[2] LS2N UMR CNRS 6004, DUKe research group,
University of Nantes, Nantes, France
philippe.leray@univ-nantes.fr

Abstract. Probabilistic relational models (PRMs) were introduced to extend the modelling and reasoning capacities of Bayesian networks from propositional to relational domains. PRMs are typically learned from relational data, by extracting from these data both a dependency structure and its numerical parameters. For this purpose, a large and rich data set is required, which proves prohibitive for many real-world applications. Since a PRM's structure can often be readily elicited from domain experts, we propose manual construction by an approach that combines qualitative concepts adapted from qualitative probabilistic networks (QPNs) with stepwise quantification. To this end, we introduce qualitative probabilistic relational models (QPRMs) and tailor an existing algorithm for qualitative probabilistic inference to these new models.

Keywords: Probabilistic relational models
Qualitative notions of probability · Qualitative probabilistic inference

1 Introduction

The formalism of *Bayesian networks* (BNs) is generally considered an intuitively appealing and powerful formalism for capturing knowledge from a problem domain along with its uncertainties [10,17]. A BN is composed of a directed graphical structure, encoding the random variables of the domain and the probabilistic (in)dependencies between them, and an associated set of conditional probability distributions, capturing the strengths of the dependencies. For computing probabilities from a Bayesian network, tailored algorithms are available that derive their efficiency from exploiting the independencies read from the graphical structure through the well-known d-separation criterion [17].

Bayesian networks are propositional in nature, as they describe a problem domain from the perspective of a single object class. Modelling a domain involving multiple interacting classes by such a representation, in fact, may result in unacceptable loss of information [15]. The formalism of *probabilistic relational*

models (PRMs) extends on the propositional modelling capacity of BNs by allowing the representation of relational information [6,9,14]. A PRM describes the object classes of a relational schema by a graphical dependency structure, with relational dependencies across classes, and supplements this structure again with conditional probability distributions. By instantiating object classes with sets of concrete objects, a *ground Bayesian network* is obtained in which inference is performed using any standard Bayesian-network algorithm.

A propositional Bayesian network for a real-world application is constructed automatically from data, using tailored machine-learning techniques [2,7,11,16], or manually, with the help of experts [13]. Automatically learning a Bayesian network typically requires a large amount of sufficiently rich data, which may prove prohibitive for various real-world applications. Since probabilistic relational models are more involved than Bayesian networks, this observation holds unabatedly for the construction of a PRM. In fact, although tailored algorithms for learning probabilistic relational models are available [5,6,15,25], many real-world relational contexts resist automated model construction. Now, experiences with building propositional Bayesian networks with domain experts show that, while assessing the typically large number of probabilities required is quite daunting for the experts involved [3], configuring the dependency structure is quite doable. These experiences have given rise to a stepwise approach to building Bayesian networks [19], that exploits qualitative notions of probability available from the framework of qualitative probabilistic networks. In this paper, we adapt and extend these qualitative notions for use with relational models, and thereby define the framework of *qualitative probabilistic relational models*.

A *qualitative probabilistic network* (QPN) is a qualitative abstraction of a propositional Bayesian network [22]. Having the same dependency structure as its numerical counterpart, a QPN summarises the probabilistic relationships between its variables by qualitative signs instead of by conditional probability tables; these signs in essence indicate the direction in which a distribution changes with observed values for the variables involved. For qualitative probabilistic inference with a QPN, an elegant message-passing algorithm is available that amounts to propagating signs throughout the network [4]. This algorithm again exploits the independencies read from the dependency structure through the d-separation criterion, and is known to have a polynomial runtime complexity.

The advocated stepwise approach to building a Bayesian network with domain experts now amounts to first building a qualitative probabilistic network and then stepwise replacing signs with numerical information. After configuring the dependency structure of a BN in the making, a domain expert assigns qualitative signs to the modelled dependencies to arrive at a QPN; specifying qualitative signs is known to require considerably less effort from experts than specifying numbers [4,8]. This qualitative network is then used to perform an initial study of the reasoning behaviour of the BN under construction. With quantification efforts yielding conditional probability distributions for (some of) the variables involved, signs are replaced with this numerical information. Before

proceeding with the quantification, the reasoning behaviour of the intermediate network, with both signs and numbers, is studied [19]. Possible inadequacies in the dependency structure can thus be detected, and amended, early on in the quantification process. This process of quantifying parts of the network and studying reasoning behaviour is continued until the network is fully quantified.

Building upon the experience that the dependency structure of a PRM can often be readily elicited from domain experts, we envision a similar approach to stepwise quantification of probabilistic relational models as for propositional Bayesian networks. In view of this goal, we introduce in this paper QPRMs, that is, *qualitative probabilistic relational models*. We will argue that the notion of qualitative sign from the framework of QPNs readily fits a relational framework. Since not all probabilistic notions employed in PRMs have qualitative counterparts in the framework of QPNs, we define additional qualitative probabilistic notions, such as *qualitative aggregation functions*, for use with QPRMs. We then tailor the basic algorithm for qualitative probabilistic inference with QPNs, to application with ground models constructed from a QPRM.

The paper is organised as follows. In Sect. 2 we review PRMs and introduce our notational conventions. Throughout the paper we will use the example introduced in Sect. 3. Section 4 then introduces QPRMs, focusing on qualitative signs and qualitative aggregation functions. In Sect. 5 we will present our algorithm for inference with a QPRM and in Sect. 6 we will demonstrate its application to our example. The paper closes in Sect. 7 with our future plans.

2   Preliminaries

Relational data involve different types of object and multiple relationships between them. For modelling and reasoning about such data, we build in this paper on the framework of probabilistic relational models [9]. In this section, we briefly review the notions involved and thereby introduce our notational conventions; we assume basic knowledge of propositional Bayesian networks throughout.

2.1   A Relational Schema and Its Instances

A *relational schema* is a pair $\mathcal{S} = (\mathcal{X}, \mathcal{R})$, where $\mathcal{X}$ is a finite non-empty set of object classes and $\mathcal{R}$ models the reference relationships between them. Each class $X \in \mathcal{X}$ has associated a finite, possibly empty, set $\mathcal{F}(X)$ of descriptive features. A *feature F* of a class X will be denoted as $X.F$ and its set of possible values will be indicated by $\Omega(X.F)$. We assume that $\Omega(X.F) = \{false, true\}$ for all $F \in \mathcal{F}(X)$, $X \in \mathcal{X}$. A known value *true* for the feature F of the class X will be indicated as $X.F = true$, or $X.f$ for short; $X.\bar{f}$ is used to indicate that F has the value *false* in X. A class $X \in \mathcal{X}$ can be *instantiated* with a finite, possibly empty, set of concrete *objects* $\mathcal{I}_{\mathcal{S}}(X) = \{x_1, \ldots, x_{n(X)}\}$, $n(X) \geq 0$.

The set $\mathcal{R}(X)$ of *reference slots* of a class X describes the relationships of objects in X with objects in other classes of the relational schema. A slot

R of a class X, indicated by $X.R$, has associated a class pair, composed of a domain class and a range class. The *domain class* of the slot $X.R$ is denoted by $Dom(X.R)$ and equals X by definition. The *range class* of $X.R$ is indicated by $Range(X.R)$ and equals some class $X' \in \mathcal{X}$ with $X' \neq X$, that is, we do not allow self-referencing slots. Each reference slot $X.R$ further has associated an inverse slot $X'.R^{-1}$ with $Dom(X'.R^{-1}) = Range(X.R) = X'$ and $Range(X'.R^{-1}) = Dom(X.R) = X$. Upon instantiation of the slot $X.R$, the reference is a set of pairs (x, x') of related objects with $x \in X$ and $x' \in X'$. The slot is a *one-to-one* reference slot if for each object $x \in X$ there exists at most one object $x' \in Range(x.r)$ for which $(x, x') \in x.r$; the slot is of type *one-to-many* if for each object $x \in X$ there may be multiple objects $x'_i \in X'$ for which $(x, x'_i) \in Range(x.r)$. *Many-to-one* and *many-to-many* reference slots are defined analogously.

While a reference slot describes a direct relationship between two classes, references can also be indirect through other classes. A *slot chain* of length k is a sequence $X.\rho = X.R_1 \ldots R_k$, $k \geq 1$, of (possibly inversed) reference slots R_i with $Range(R_i) = Dom(R_{i+1})$. The domain class of the slot chain $X.\rho$ equals the domain of the first reference slot $X.R_1$, that is, $Dom(X.\rho) = X$; the range of $X.\rho$ equals the range class of the chain's final slot, that is, $Range(X.\rho) = Range(X'.R_k)$ with $X' = Range(X.R_1 \ldots R_{k-1})$. A slot chain $X.\rho$ describes a relation between the objects from its domain and range classes and, upon instantiation, also is a set of pairs of related objects. A slot chain can again be of type one-to-one, one-to-many, many-to-one and many-to-many.

An *instance* $\mathcal{I}_{\mathcal{S}}$ of a relational schema $\mathcal{S}$ includes an assignment of sets of objects to the classes in $\mathcal{S}$, as described above. The instance may further include assignments to the features and reference slots of the classes. In the sequel, we assume that instances have *feature uncertainty* only, that is, while features may not have an assigned value, the values of all reference slots are known.

2.2   The PRM and Its Ground Bayesian Network

A relational schema $\mathcal{S}$ has associated a dependency structure $\mathcal{G}_{\mathcal{S}}$, which essentially is a directed acyclic graph composed of nodes, modelling the classes' features, and their interrelationships; this structure is a *meta-model*, accommodating possible instances of the relational schema [6].

The *dependency structure* $\mathcal{G}_{\mathcal{S}}$ associated with a schema $\mathcal{S}$ includes a node $X.F_i$ for every feature $F_i \in \mathcal{F}(X)$ of each class $X \in \mathcal{X}$. The arcs of $\mathcal{G}_{\mathcal{S}}$ capture the dependencies between the modelled features. The set of all parents of a node $X.F_i$ in $\mathcal{G}_{\mathcal{S}}$ is denoted as $\Pi_{\mathcal{G}_{\mathcal{S}}}(X.F_i)$. A parent of $X.F_i$ can be either a feature $X.F_j$ of the same class or a feature $X.\rho.F_j$ of the class reachable through the slot chain $X.\rho$; the associated arc is called a *type-I arc* in the former case and a *type-II arc* in the latter case. When the slot chain underlying a type-II arc is of type many-to-one or many-to-many, the value of the feature $x.F_i$ of some concrete object x upon instantiation may depend on the values of the features $x'.F_j$ of multiple objects $x' \in \mathcal{I}_{\mathcal{S}}(X')$ with $X' = Range(X.\rho)$. The exact dependency then is described by an *aggregation function* γ which takes a multiset of values

from $\Omega(X'.F_j)$ and outputs a single such value; the function is indicated over the corresponding arc in the dependency structure. Common aggregation functions include the *minimum*, *maximum* and *mode* functions for numerical features [9].

A probabilistic relational model (PRM) for a relational schema $\mathcal{S}$ now is composed of the dependency structure $\mathcal{G}_{\mathcal{S}}$ of the schema, supplemented with a conditional probability table $\Pr(X.F_i \mid \Pi_{\mathcal{G}_{\mathcal{S}}}(X.F_i))$ for each node $X.F_i$. The conditional probability table for the node $X.F_i$ thereby includes conditional probability distributions over $X.F_i$ given each possible value combination for its parents, as in a propositional Bayesian network. For an instance $\mathcal{I}_{\mathcal{S}}$ of the relational schema, a *ground Bayesian network* is constructed by replicating the PRM for every concrete object in $\mathcal{I}_{\mathcal{S}}$. The dependency structure of this ground network includes, for each node $X.F_i$ from $\mathcal{G}_{\mathcal{S}}$ and for every object $x \in \mathcal{I}_{\mathcal{S}}(X)$, a replicate node $x.F_i$, along with copies of the original incident arcs from $\mathcal{G}_{\mathcal{S}}$. For the aggregation function associated with a many-to-one type-II arc $X'.F_j \to X.F_i$ in $\mathcal{G}_{\mathcal{S}}$, an auxiliary node is included in the dependency structure of the ground network, with incoming arcs from all replicate nodes $x'_k.F_j$ with $x'_k \in \mathcal{I}_{\mathcal{S}}(X')$ and a single emanating arc to $x.F_i$ with $x \in \mathcal{I}_{\mathcal{S}}(X)$; for the aggregation function associated with a many-to-many type-II arc, a similar construct is included. The nodes modelling features are further assigned copies of the conditional probability tables from the PRM, and the auxiliary nodes are assigned a conditional probability table conform the semantics of their associated aggregation function.
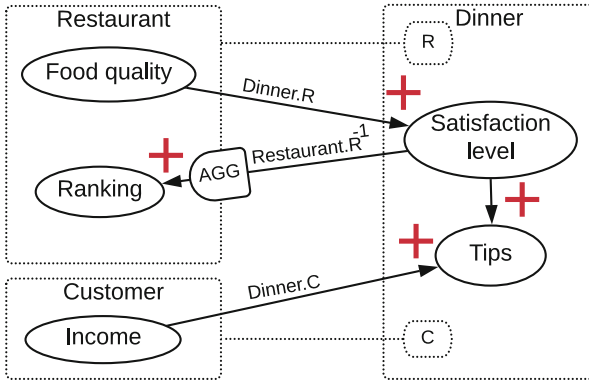

Fig. 1. The dependency structure of the example PRM, with five features (ellipses) for three classes (rounded boxes); the structure is supplemented with qualitative signs (the product synergies involved and their signs are not shown).

3   An Example

To illustrate a PRM and an associated ground Bayesian network, we introduce a small fictitious model for our example. The example pertains to customer satisfaction with restaurants. We assume that the quality of the food served at a restaurant is a determinant of a customer's level of satisfaction. This level can

be read from the tip left by the customer after dining, although the size of the tip is dependent of her income as well. For the various restaurants involved, an internet ranking is maintained, reflecting customer satisfaction.

The example corresponds to a relational schema $\mathcal{S}$ which includes the three object classes Restaurant, Dinner and Customer, with five features in all, and the two reference slots R and C, defining the restaurant and customer, respectively, related to a specific dinner. The dependency structure of the PRM, shown in Fig. 1, includes a single type-I arc, between the Satisfaction level and Tips nodes of the Dinner class. It further includes three type-II arcs: the arc labelled Dinner.C between the Income and Tips nodes describes the dependency between the Dinner.Tips and Dinner.C.Income features; the many-to-one type-II arc between the Restaurant.Ranking and Restaurant.R^{-1}.Satisfaction level features expresses the information that the ranking of a restaurant is dependent of the satisfaction levels of all dinners taken there, with their joint effect captured by the aggregation function AGG; the third type-II arc describes the dependency between the satisfaction level of a specific dinner and the food quality at that specific restaurant.

Figure 2 shows an example ground Bayesian network constructed from the relational schema and dependency structure of Fig. 1, for an instance with two restaurants, four dinners and two customers. We note that instantiation of the many-to-one arc between the Restaurant.R^{-1}.Satisfaction level and Restaurant.Ranking features in the dependency structure has resulted, in the ground Bayesian network, in an auxiliary node with a single emanating arc to the ranking of the restaurant and incoming arcs from the satisfaction levels of all dinners taken there; we will return to the semantics of the auxiliary node in Sect. 4.2.
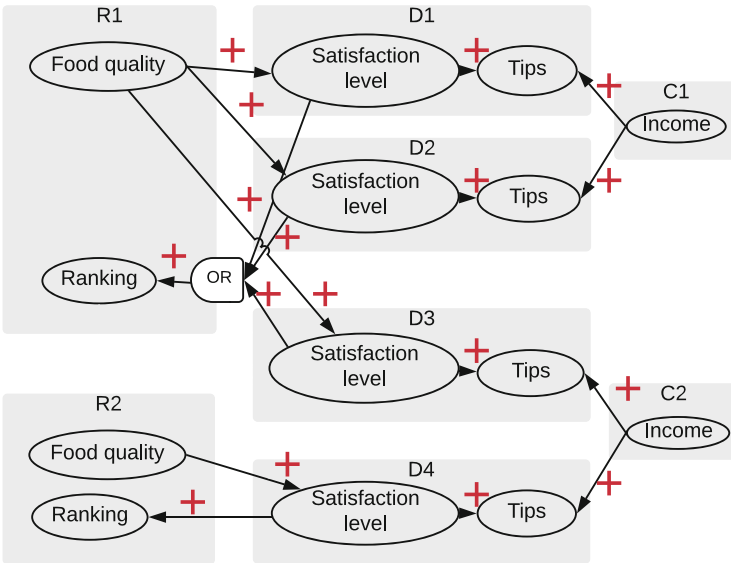

Fig. 2. The dependency structure of the ground Bayesian network constructed for an instance with two restaurants, two customers and four dinners; the structure is supplemented with qualitative signs (product synergies and their signs are not shown).

4   Qualitative Probabilistic Relational Models

Building upon the relational concepts reviewed above, we introduce in this section *qualitative probabilistic relational models* (QPRMs). Like a PRM, a qualitative relational model takes a relational schema and has an associated dependency structure. It differs from a PRM in its associated uncertainty specifications: while a PRM specifies a conditional probability table for each node in the dependency structure, a QPRM specifies a qualitative sign for each arc. These signs are taken from the propositional framework of *qualitative probabilistic networks* (QPNs) [22] and are now fitted into the relational framework.

4.1   Qualitative Signs for QPRMs

The qualitative signs of propositional QPNs assume an ordering on the value sets of all variables involved [22]. For using these signs in qualitative probabilistic relational models therefore, we have to define orderings on the value sets of the variables of such a model. We recall from Sect. 2.1 that we assumed all features of a relational model to be binary. Without loss of generality, we now assume the total ordering on the value set $\Omega(X.F)$ of a feature $X.F$ to be $\bar{f} \leq f$. The orderings per feature induce a partial ordering $\preceq$ over the set of value combinations for any (sub)set of features. We say that a value combination $\mathbf{f}$ for some feature set $\mathbf{F} \subseteq \bigcup_{X \in \mathcal{X}} \mathcal{F}(X)$ is *higher-ordered* than a value combination $\mathbf{f}'$ for $\mathbf{F}$, if $\mathbf{f}' \preceq \mathbf{f}$; if $\mathbf{f} \preceq \mathbf{f}'$, we say that $\mathbf{f}$ is lower-ordered than $\mathbf{f}'$.

A propositional QPN has associated with each arc in its dependency structure a *qualitative sign* $\delta \in \{+, -, 0, ?\}$. Although these signs are qualitative in nature, they have a well-defined probabilistic meaning in terms of the orderings on the value sets per variable [22]; stated informally, the sign $\delta = $ '+', for example, for an arc $A \rightarrow B$ in a QPN's dependency structure means that observing the higher value for the variable A causes the probability of the higher value of the variable B to increase, regardless of any other such influences on the probability distribution of B. In the dependency structure of a relational model, we now likewise associate qualitative signs with arcs. The sign $\delta = $ '+' for the (type-I or type-II) arc $X'.F_j \rightarrow X.F_i$, for example, is taken to indicate that

$$\Pr(X.f_i \mid X'.f_j, \mathbf{w}) \;\; \geq \;\; \Pr(X.f_i \mid X'.\bar{f}_j, \mathbf{w})$$

for all value combinations $\mathbf{w} \in \Omega(\mathbf{W})$ for the set $\mathbf{W} = \Pi_{\mathcal{G}_S}(X.F_i) \setminus \{X'.F_j\}$ of parents of $X.F_i$ other than $X'.F_j$. This sign thus expresses that observing the higher value of the feature $X'.F_j$ induces an increased probability for the higher value of the feature $X.F_i$, regardless of the values of the other parents of $X.F_i$; the sign thereby has essentially the same semantics as in a propositional QPN. The signs '−' and '0' have analogous meanings, replacing $\geq$ in the inequality above by $\leq$ and $=$, respectively. A '?'-sign for the arc $X'.F_j \rightarrow X.F_i$ indicates that there are value combinations $\mathbf{w}_1, \mathbf{w}_2 \in \Omega(\mathbf{W})$, $\mathbf{w}_1 \neq \mathbf{w}_2$, such that observing the higher value of $X'.F_j$ induces an increased probability of the higher value of $X.F_i$ in the context of $\mathbf{w}_1$ and causes a decrease of this probability in the context

of $\mathbf{w}_2$. The sign thus indicates that the direction of change in the probability distribution of $X.F_i$ depends on the context and, hence, is not unambiguous.

In addition to the signs per arc, a propositional QPN associates two qualitative signs with each pair of parents per node. As in a numerical Bayesian network, an observed value for a multi-parent node A of a QPN induces a mutual dependency, called a *product synergy*, between each parent pair of A [4,23]. The sign of this dependency is dependent of the specific value observed for A. The two qualitative signs associated with a parent pair of the node A now are the signs of the dependencies induced by the two possible values of A. In the dependency structure of a relational qualitative model, we similarly associate two signs with each parent pair $X'.F_j$, $X''.F_k$ of a node $X.F_i$. The parent pair is said to exhibit a *product synergy* of sign $\delta = $ '$-$' for the value f_i of $X.F_i$ if

$$\begin{aligned} \Pr(X.f_i \,|\, X'.f_j, X''.f_k, \mathbf{w}) \cdot \Pr(X.f_i \,|\, X'.\bar{f}_j, X''.\bar{f}_k, \mathbf{w}) - \\ \Pr(X.f_i \,|\, X'.f_j, X''.\bar{f}_k, \mathbf{w}) \cdot \Pr(X.f_i \,|\, X'.\bar{f}_j, X''.f_k, \mathbf{w}) \;\; \leq \;\; 0 \end{aligned}$$

for all value combinations $\mathbf{w}$ of $\mathbf{W} = \Pi_{\mathcal{G}_S}(X.F_i) \setminus \{X'.F_j, \; X''.F_k\}$. Product synergies of signs '$+$', '0' and '?' for f_i again are defined analogously; a negative product synergy for the value $\bar{f}_i$ of $X.F_i$ is defined by replacing f_i by $\bar{f}_i$ in the formula above. Since product synergies in essence are induced between pairs of parents of a node with an observed value, their signs are defined for parent pairs [4,23]; an extended concept encompassing all parents of a node has been proposed, however [1].

Qualitative signs exhibit various useful properties upon combination within a propositional QPN [22]. The *symmetry* property states that, if a node A exerts a qualitative influence on a node B over the arc $A \rightarrow B$, then B exerts an influence of the same sign over this arc on A. The property of *transitivity* states that, if a node A exerts a qualitative influence of sign δ_i on a node B which in turn has an influence of sign δ_j on node C, then the net influence of A on C over the trail of associated arcs is of sign $\delta_i \otimes \delta_j$, with the $\otimes$-operator as in Table 1. The *composition* property states that influences along multiple parallel trails convening at a node A combine into a net influence on A of a sign given by the $\oplus$-operator from Table 1. The three properties of symmetry, transitivity and composition underlie the basic algorithm for qualitative probabilistic inference with a propositional QPN; we will return to this observation in Sect. 5, where we adapt this algorithm to the relational context of QPRMs.

4.2   Qualitative Aggregation Functions

Thus far, we fitted the notion of qualitative sign from propositional QPNs into the framework of QPRMs. A relational dependency structure with such signs can now in essence be instantiated to a ground QPN. This ground network is constructed in a similar way as a ground Bayesian network is constructed from a numerical PRM, that is, by replicating the qualitative relational model for every concrete object of the instance at hand. Each replicate arc thereby inherits the qualitative sign of the original arc from the dependency structure and each

Table 1. The sign-product operator $\otimes$ and the sign-plus operator $\oplus$ for combining qualitative signs.

$\otimes$	$+$	$-$	0	$?$		$\oplus$	$+$	$-$	0	$?$
$+$	$+$	$-$	0	$?$		$+$	$+$	$?$	$+$	$?$
$-$	$-$	$+$	0	$?$		$-$	$?$	$-$	$-$	$?$
0	0	0	0	0		0	$+$	$-$	0	$?$
$?$	$?$	$?$	0	$?$		$?$	$?$	$?$	$?$	$?$

parent pair per node inherits the two signs of the induced product synergies. Upon constructing a ground Bayesian network from a numerical PRM however, instantiation of a many-to-one or many-to-many type-II arc is more involved than simple replication. We recall from Sect. 2.2 that such an arc has associated an aggregation function that takes a multiset of values from the value set of some feature and outputs a single such value. Upon instantiation, an auxiliary node is created with a conditional probability table encoding the semantics of this aggregation function. Since propositional QPNs include just type-I arcs, the notion of aggregation function does not have a qualitative counterpart as yet. We now define qualitative aggregation functions and detail the instantiation of a type-II arc upon construction of a ground qualitative model from a QPRM.

While a ground Bayesian network encodes the semantics of an aggregation function from a PRM in the conditional probability table of an auxiliary node, a ground QPN only offers qualitative signs and their combination for this purpose. Based upon this observation, we take a *qualitative aggregation function* to be a function that takes for its input a multiset of qualitative signs and outputs a single such sign. We say that the function is '+'-*preserving* if it returns a '+' whenever its input includes at least one '+'; it is called '−'-*preserving* if it returns a '−' whenever its input includes at least one '−'. If a qualitative aggregation function is neither '+'- nor '−'-preserving, it is called *non-preserving.* An example non-preserving function is the parity function which returns a '+' if the number of '+'s in its input is even and a '−' otherwise. In Sect. 5, we will see that non-preserving aggregation functions like this parity function are undesirable in QPRM applications as these will lead to ambiguous results upon inference. To accommodate possible dependencies induced between its input features upon inference, a qualitative aggregation function has further associated two qualitative signs to describe the product synergies involved.

We now address instantiation of a type-II arc of a QPRM. We consider to this end the many-to-one type-II arc $X'.F_j \rightarrow X.F_i$, with an associated qualitative aggregation function γ and sign δ. For the ground QPN given an instance $\mathcal{I}_\mathcal{S}$, again an auxiliary node A is created, with multiple incoming arcs $x'_k.F_j \rightarrow A$ from all objects $x'_k \in \mathcal{I}_\mathcal{S}(X')$, $k = 1, \ldots, n(X')$, and a single emanating arc $A \rightarrow x.F_i$ for $x \in \mathcal{I}_\mathcal{S}(X)$. The sign δ of the original arc from the QPRM is assigned to this latter arc. For embedding the semantics of the aggregation function γ, we build on the idea underlying the composition property of qualitative influences and introduce a new operator $\oplus_\gamma$ for combining signs of multiple parallel trails.

Where the $\oplus$-operator gives a combined sign for multiple trails convening at any feature node in general, the $\oplus_\gamma$-operator does so for trails convening at the auxiliary node A specifically. All arcs $x'_k.F_j \to A$ are further assigned the '+'-sign to mimic the partial order $\preceq$ on the value combinations of the set $\{x'_k.F_j \mid k = 1, \ldots, n(X')\}$. The thus specified signs now guarantee that, for example, with a '+'-preserving aggregation function, a higher-ordered value combination **f** for the feature $x'_k.F_j$ from all objects x'_k, results in an influence of sign '+' $\otimes \delta$ on $x.F_i$. To conclude, for each parent pair of the auxiliary node A, the signs of the two possible product synergies are copied from the aggregation function γ.

procedure PropagateSign(*trail*,*from*,*to*,*messagesign*):

sign[*to*] $\leftarrow$ sign[*to*] $\oplus$ *messagesign*;
trail $\leftarrow$ *trail* $\cup$ {*to*};
for each active neighbour $x'.F_j$ of *to*
do *linksign* $\leftarrow$ sign of (induced) influence between *to* and $x'.F_j$;
    messagesign $\leftarrow$ sign[*to*] $\otimes$ *linksign*;
    if $x'.F_j \notin trail$ **and** sign[$x'.F_j$] $\neq$ sign[$x'.F_j$] $\oplus$ *messagesign*
    then PropagateSign(*trail*,*to*,$x'.F_j$,*messagesign*).

Fig. 3. A sketch of the sign-propagation algorithm for qualitative probabilistic inference with ground QPNs.

5 Qualitative Inference with a Ground QPN

Probabilistic inference with an instance of a numerical PRM essentially amounts to inference in the ground Bayesian network constructed from the instance. Since this ground network is a BN, essentially any standard inference algorithm can be used for this purpose, although various tailored algorithms are available as well [12,18,24]. In this section, we detail an algorithm for qualitative probabilistic inference with an instance of a QPRM, which builds more or less directly on the basic sign-propagation algorithm available for propositional QPNs.

    The basic idea of the sign-propagation algorithm for inference with QPNs is to trace the effect of observing the value of a specific node on the other nodes in the dependency structure, by passing messages between neighbours. The algorithm thereby establishes qualitative signs for the nodes, which indicate the direction of change in the node's probability distribution, occasioned by the new observation [4]. A sketch of the basic algorithm is given in Fig. 3, stated in terms of the feature nodes of a ground QPN. Initially, all node signs are set to '0'. To enter a new observation, the appropriate sign is sent to the observed node, that is, either a '+' for the value *true* or a '−' for *false*. In a ground QPN, this observed node is taken as the *perspective* from which probabilistic effects will be traced along trails in the dependency structure. Each feature node receiving a message updates its own sign with this message, by application of the $\oplus$-operator; dependent of the origin of the message, an auxiliary node applies the

$\oplus_\gamma$-operator for its aggregation function γ to this end. If its sign changes, the node sends a sign-message to each active neighbour that is not yet on the trail being traversed from the perspective. An *active* neighbour of a node $x.F_i$ is a non-instantiated node $x'.F_j$ that satisfies one of the following conditions:

(i) $x'.F_j$ is directly connected to $x.F_i$ either by an emanating arc or an incoming arc; or,

(ii) $x'.F_j$ is connected to $x.F_i$ indirectly through a head-to-head trail $x'.F_j \rightarrow x''.F_k \leftarrow x.F_i$ such that either $x''.F_k$ or one of its descendants is observed.

In the first case, the sign of the message sent by $x.F_i$ to $x'.F_j$ is computed through application of the $\otimes$-operator to the (possibly new) sign of $x.F_i$ and the sign of the arc over which the message is sent. In the second case, the message will be sent over the dependency that is induced by observation of the value of (a descendant of) $x''.F_k$; we recall that the resulting product synergy has an associated sign that is available from the observed node or from the appropriate auxiliary node. The sign of the message sent over the induced dependency to the node $x'.F_j$ now is established through application of the $\otimes$-operator to the sign of $x.F_i$ and the sign of the product synergy involved. This process of message passing is repeated throughout the network, visiting nodes along trails starting at the perspective, until nodes no longer change sign. The algorithm is known to run in polynomial time as node signs can change at most twice, from the initial '0' to either '+' or '−', and then to '?' [4].

The original sign-propagation algorithm for propositional QPNs was designed for propagating the observed value of a single node, possibly in a context of previously entered evidence; multiple observations were dealt with essentially by entering them one at a time and combining results by means of the $\oplus$-operator. Later it was shown that multiple observations could be propagated simultaneously throughout a QPN by determining, for each observed node separately, to which nodes its effects should be traced; for further details, we refer to [21].

6   The Example Revisited

To illustrate qualitative probabilistic inference in a ground QPN, we consider again the dependency structure of our example QPRM, from Fig. 1. We further consider the ground network from Fig. 2, which resulted from the instance described in Sect. 3; we note that the OR function modelled in the ground network is a '+'-preserving qualitative aggregation function.

We suppose that we know that the quality of the food served at restaurant R1 is good, which is expressed by the value *true*. After entering this observation into the ground network, the sign-propagation algorithm traces its effects throughout the dependency structure. It thereby establishes the sign '+' for the nodes R1.Food quality, R1.Ranking, Di.Satisfaction level and Di.Tips, $i = 1, 2, 3$; all other nodes retain their initial node sign '0'. Although a high food quality will serve to increase the probability of large tips being left at the three dinners at this restaurant, it does not affect the probability distributions of the customers' incomes nor that of the food quality at the other restaurant.

We now suppose that customer C1 left large tips (expressed by the value *true*) at her first dinner at restaurant R1, and that customer C2 was less satisfied with his dinner at this restaurant and left just small tips (*false*). We further suppose that the product synergies given either value of the Tips node are equal to '0', which expresses that regardless of the size of the tips left, a customer's income and satisfaction with a specific dinner are independent. The two findings are entered into the ground network and are propagated throughout the dependency structure to the appropriate nodes by the sign-propagation algorithm; the resulting node signs are given in Table 2. The propagation results demonstrate, for example for the node R1.Food quality, that combining parallel qualitative influences by the $\oplus$-operator can yield ambiguous signs. Such an ambiguity, in fact, results whenever influences with opposite signs are combined; we say that the *trade-off* that is reflected by the conflicting influences cannot be resolved at the level of detail offered by the qualitative signs. In contrast, the $\otimes$-operator cannot introduce ambiguities by itself; it will cause ambiguous signs to spread throughout the network once they have arisen, however, as can be seen by the propagation result for, for example, the node D1.Satisfaction level.

Table 2. The node signs returned by the sign-propagation algorithm for our example ground QPN, given D1.Tips $= true$ and D3.Tips $= false$.

Node name	Node sign	Node name	Node sign
R1.Food quality	?	C1.Income	$+$
R1.Ranking	?		
R2.Food quality	0	C2.Income	$-$
R2.Ranking	0		
D1.Satisfaction level	?	D3.Satisfaction level	?
D1.Tips	$+$	D3.Tips	$-$
D2.Satisfaction level	?	D4.Satisfaction level	0
D2.Tips	?	D4.Tips	$-$

7   Conclusions and Future Research

Real-world application of PRMs is often prohibited by the need of a large amount of sufficiently rich data for their automated construction. For practical construction with the help of domain experts, we envision an approach to stepwise quantification of probabilistic relational models similar to the approach proposed before for propositional Bayesian networks. As this approach builds upon qualitative probabilistic notions, we introduced in this paper the framework of qualitative probabilistic relational models. For this purpose, we adapted and extended available qualitative notions of probability to the relational framework and have adapted an existing algorithm for qualitative probabilistic inference to ground qualitative networks. Our qualitative probabilistic relational models

are expected to allow ready construction by domain experts; in addition, they provide for efficiently studying reasoning behaviour, as the associated inference algorithm in essence is polynomial in the number of variables involved.

Straightforward use of qualitative probabilistic networks, be they propositional or relational, in real-world applications is associated with some disadvantages originating from their lack of representation detail. It is well known, for example, that qualitative probabilistic inference shows a tendency to lead to weak, and even uninformative, results. Researchers have attributed this tendency to the granularity of the qualitative signs employed and have proposed solutions such as including a notion of strength of qualitative signs [20]. It is expected that these and other extensions of the framework of propositional QPNs can be incorporated in our framework of qualitative probabilistic relational models. Another approach to forestalling uninformative results upon inference may be to further tailor the propagation algorithm to the prospective use of QPRMs. As the next step in our research, we intend to now first study the practicability of our new relational framework in a real-world application, in animal ecology.

References

1. Bolt, J.H.: Bayesian networks: the parental synergy. In: Jaeger, M., Nielsen, T.D. (eds.) Proceedings of the Fourth European Workshop on Probabilistic Graphical Models, Hirtshals, pp. 33–40 (2008)
2. Daly, R., Shen, Q., Aitken, S.: Learning Bayesian networks: approaches and issues. Knowl. Eng. Rev. **26**, 99–157 (2011)
3. Druzdel, M.J., van der Gaag, L.C., Henrion, M., Jensen, F.V.: Building probabilistic networks: where do the numbers come from? Guest editors' introduction. IEEE Trans. Knowl. Data Eng. **12**, 481–486 (2000)
4. Druzdzel, M.J., Henrion, M.: Efficient reasoning in qualitative probabilistic networks. In: Fikes, R., Lehnert, W. (eds.) Proceedings of the 11th National Conference on Artificial Intelligence, pp. 548–553. AAAI Press, Menlo Park (1993)
5. Ettouzi, N., Leray, Ph., Ben Messaoud, M.: An exact approach to learning probabilistic relational model. In: Antonucci, A., Corani, G., de Campos, C. (eds.) Proceedings of the 8th Conference on Probabilistic Graphical Models, vol. 52, pp. 171–182. PMLR (2016)
6. Friedman, N., Getoor, L., Koller, D., Pfeffer, A.: Learning probabilistic relational models. In: Dean, T. (ed.) Proceedings of the 16th International Joint Conference on Artificial Intelligence, pp. 1300–1307. Morgan Kaufmann, San Francisco (1999)
7. Friedman, N., Nachman, I., Peér, D.: Learning Bayesian network structure from massive datasets: the sparse candidate algorithm. In: Laskey, K., Prade, H. (eds.) Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence, pp. 206–215. Morgan Kaufmann, San Francisco (1999)
8. van der Gaag, L.C., Renooij, S., Schijf, H.J.M., Elbers, A.R., Loeffen, W.L.: Experiences with eliciting probabilities from multiple experts. In: Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., Yager, R.R. (eds.) IPMU 2012. CCIS, vol. 299, pp. 151–160. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31718-7_16
9. Getoor, L.: Learning Statistical Models from Relational Data. Ph.D. thesis, Stanford (2001)

10. Jensen, F.V., Nielsen, T.D.: Bayesian Networks and Decision Graphs. Springer, New York (2007). https://doi.org/10.1007/978-1-4757-3502-4
11. Jordan, M.I.: Learning in Graphical Models. MIT Press, Cambridge (1999)
12. Kaelin, F., Precup, D.: An approach to inference in probabilistic relational models using block sampling. J. Mach. Learn. Res. **13**, 315–330 (2010)
13. Kjaerulff, U.B., Madsen, A.L.: Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis. Springer, New York (2010). https://doi.org/10.1007/978-1-4614-5104-4
14. Koller, D., Pfeffer, A.: Probabilistic frame-based systems. In: Mostov, J., Rich, C. (eds.) Proceedings of the 15th National Conference on Artificial Intelligence, pp. 580–587. AAAI Press, Menlo Park (1998)
15. Maier, M.E., Marazopoulou, K., Jensen, D.D.: Reasoning about independence in probabilistic models of relational data (2013). arXiv: 1302.4381
16. Neapolitan, R.E.: Learning Bayesian Networks. Pearson Prentice Hall, Upper Saddle River (2004)
17. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco (1988)
18. Pfeffer, A.J.: Probabilistic reasoning for complex systems. Ph.D. thesis, Stanford University, Stanford (2000)
19. Renooij, S., van der Gaag, L.C.: From qualitative to quantitative probabilistic networks. In: Darwiche, A., Friedman, N. (eds.) Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence, pp. 422–429. Morgan Kaufmann, San Francisco (2002)
20. Renooij, S., van der Gaag, L.C.: Enhanced qualitative probabilistic networks for resolving trade-offs. Artif. Intell. **172**, 1470–1494 (2008)
21. Renooij, S., van der Gaag, L.C., Parsons, S.: Propagation of multiple observations in QPNs revisited. In: Proceedings of the 15th European Conference on Artificial Intelligence, pp. 665–669. IOS Press, Amsterdam (2002)
22. Wellman, M.P.: Fundamental concepts of qualitative probabilistic networks. Artif. Intell. **44**, 257–303 (1990)
23. Wellman, M.P., Henrion, M.: Qualitative intercausal relations, or explaining explaining away. In: Allen, J.F., Fikes, R., Sandewall, E. (eds.) KR91: Principles of Knowledge Representation and Reasoning, pp. 535–546. Morgan Kaufmann, San Francisco (1991)
24. Wuillemin, P.-H., Torti, L.: Structured probabilistic inference. Int. J. Approx. Reason. **53**, 946–968 (2012)
25. Xiao-Lin, L., Xiang-Dong, H.: A hybrid particle swarm optimization method for structure learning of probabilistic relational models. Inf. Sci. **283**, 258–266 (2014)