KYBERNETIKA — VOLUME *52* (2016), NUMBER 5, PAGES 696–723

COMPOSITIONAL MODELS, BAYESIAN MODELS AND RECURSIVE FACTORIZATION MODELS

Francesco M. Malvestuto

Compositional models are used to construct probability distributions from lower-order probability distributions. On the other hand, Bayesian models are used to represent probability distributions that factorize according to acyclic digraphs. We introduce a class of models, called *recursive factorization models*, to represent probability distributions that recursively factorize according to sequences of sets of variables, and prove that they have the same representation power as both compositional models generated by sequential expressions and Bayesian models. Moreover, we present a linear (graphical) algorithm for deciding if a conditional independence is valid in a given recursive factorization model.

Keywords: Bayesian model, compositional model, conditional independence, Markov property, recursive model, sequential expression

Classification: 05C65, 05C85, 1699, 65C50, 60E99, 68T37

1. INTRODUCTION

Compositional models of sequential type [2, 4, 5, 6, 8, 9] were originally introduced to construct probability distributions from lower-order probability distributions as an operational alternative to *Bayesian models* [3] (also called "directed Markov models" [16]). Compositional models were also applied to belief functions [7, 11, 12], possibility functions [11] and Shenoy valuations [10]. A more general type of compositional model, namely, the model generated by a *compositional expression*, was introduced to compose two or more *metric distribution functions* [19, 20], and such models find applications also to multidimensional databases [18, 23].

In the framework of probability distributions, with a compositional expression we associate a *composition scheme* (see Section 5.1), which is a symbolic formula for the result of the composition. In some cases, the composition scheme has a closed form and a typical case is the composition scheme associated with a sequential expression (see "formal ratios" introduced by Kratochvíl [13, 14]).

In this paper we first prove that the following question has an affirmative question: Does the formalism of compositional models of sequential type have the same representation power as the formalism of Bayesian models?

DOI: 10.14736/kyb-2016-5-0696

Our result is stronger than the result proven by Jiroušek and Kratochvíl (see Section 5 in [9]) which states that, for every Bayesian model, there exists a sequential expression that generates the same probability distributions represented by the Bayesian model. In order to prove this sort of equivalence between compositional models generated by sequential expressions and Bayesian models, we introduce *recursive factorization models* and prove that they are equivalent to both compositional models generated by sequential expressions and Bayesian models. Finally, we provide a linear graphical algorithm for recognizing conditional independences holding in a recursive factorization model which can be applied to both compositional models generated by sequential expressions and Bayesian models.

The paper is organized as follows. Section 2 contains basic definitions on probability distributions and their composition. In Section 3 the definitions of a compositional expression and of its value under a valid interpretation are recalled and some further properties are provided. In Section 4 we revise the evaluation procedure given in [20] and introduce the notion of the composition scheme for a compositional expression. In Section 5 we recall the definitions of compositional models and of Bayesian models as well as their Markov properties. In Section 6 we introduce recursive factorization models and prove that they have the representation power as both sequential compositional models and Bayesian models. In Section 7 we provide a linear algorithm for recognizing conditional independences valid in a recursive factorization model. Finally, in Section 8 we suggest a possible direction for future research.

2. PRELIMINARIES

Throughout we only consider discrete variables which take their values from finite sets and whose values are mutually exclusive and exhaustive. We use the initial capital-case letters of the alphabet (e. g., A, B, C) to denote variables, and the other capital-case letters to denote sets of variables (e. g., X, Y, Z); moreover, sets of variables are written as strings of variables; thus, ABC stands for $\{A, B, C\}$.
Let X be a non-empty set of variables. A *configuration* of X is an assignment of values to the variables in X. We use the lower-case letter $\mathbf{x}$ to denote a configuration of X; for example, the configuration of ABC with $A = \mathbf{a}$, $B = \mathbf{b}$ and $C = \mathbf{c}$ is written $\mathbf{abc}$. By $\mathbf{X}$ we denote the set of all configurations of X. Let Y be a non-empty subset of X; given a configuration $\mathbf{x}$ of X, by $\mathbf{x}_Y$ we denote the configuration of Y obtained from $\mathbf{x}$ by ignoring the values of the variables in $X \setminus Y$.

2.1. Probability distributions

Let X be a non-empty set of variables. A *probability distribution on* X is a mapping $f(X)\colon \mathbf{X} \to [0, 1]$ such that

$$\sum_{\mathbf{x} \in \mathbf{X}} f(\mathbf{x}) = 1\,.$$

The *support* of $f(X)$, denoted by $\|f\|$, is the set of all configurations $\mathbf{x}$ of X with $f(\mathbf{x}) \neq 0$.

Let $f(X)$ and $g(X)$ be probability distributions; $f(X)$ is *dominated* by $g(X)$ if $\|f\| \subseteq \|g\|$.

By definition, the support of a probability distribution on X may be any non-empty subset of the configuration space $\mathbf{X}$ of X and can be viewed as a *relation on scheme* X in the sense of relational algebra [17]; therefore, we can apply the following two operators of relational algebra to supports of probability distributions:

(*projection*) Let r be a relation on scheme X, and let Y be a non-empty subset of X. The projection of r onto Y is the relation

$$\pi_Y(r) = \{\mathbf{x}_Y : \mathbf{x} \in r\}.$$

Note that if $X = Y$ then $\pi_Y(r) = r$.

(*natural join*) Let r and s be relations on schemes X and Y respectively. The (natural) join of r and s is the relation on scheme $V = X \cup Y$ defined as follows:

$$r \bowtie s = \{\mathbf{v} \in \mathbf{V} : \mathbf{v}_X \in r \text{ and } \mathbf{v}_Y \in s\}.$$

Note that if $X = Y$ then $r \bowtie s = r \cap s$. The join of relations is both associative and commutative [17].

Remark 2.1. Let r and s be relations on schemes X and Y respectively. A configuration $\mathbf{x}$ of X belonging to r contributes to $r \bowtie s$ only if $\mathbf{x}$ matches some configuration $\mathbf{y} \in s$ in the sense that $\mathbf{x}_{X\cap Y} = \mathbf{y}_{X\cap Y}$. As a consequence, one has that

(*i*) if $Y \subseteq X$ then $r \bowtie s \subseteq r$ where the equality holds if and only if $\pi_Y(r) \subseteq s$;

(*ii*) if $X \cap Y = \emptyset$ then $\pi_X(r \bowtie s) = r$; otherwise, $\pi_X(r \bowtie s) = r \bowtie \pi_{X\cap Y}(s)$ so that, by (*i*), $\pi_X(r \bowtie s) \subseteq r$ where the equality holds if and only if $\pi_{X\cap Y}(r) \subseteq \pi_{X\cap Y}(s)$.

Remark 2.2. Let r be a relation on scheme V, and let X and Y be subsets of V. For every $\mathbf{v} \in r$, we have $\mathbf{v}_X \in \pi_X(r)$ and $\mathbf{v}_Y \in \pi_Y(r)$ so that $\mathbf{v} \in \pi_X(r) \bowtie \pi_Y(r)$; it follows that $r \subseteq \pi_X(r) \bowtie \pi_Y(r)$. More in general, if r is a relation on scheme $V = X_1 \cup \ldots \cup X_n$, then

$$r \subseteq \pi_{X_1}(r) \bowtie \cdots \bowtie \pi_{X_n}(r).$$

2.2. Marginals

Let $f(X)$ be a probability distribution, and let Y be a non-empty subset of X. The *marginal* of $f(X)$ on Y, written $f^{\downarrow Y}$, is the probability distribution on Y defined as follows: for every configuration $\mathbf{y}$ of Y

$$f^{\downarrow Y}(\mathbf{y}) = \sum_{\mathbf{x}\in\mathbf{X}:\ \mathbf{x}_Y=\mathbf{y}} f(\mathbf{x}).$$

Lemma 2.3. (see Remark 4.6 in Malvestuto [19]) The support of the marginal of a probability distribution $f(X)$ on a non-empty subset Y of X is given by the projection of the support of $f(X)$ onto Y, that is,

$$\|f^{\downarrow Y}\| = \pi_Y(\|f\|)\,.$$

By Lemma 2.3, for every configuration $\mathbf{y}$ of Y one has

$$f^{\downarrow Y}(\mathbf{y}) = \begin{cases} \sum_{\mathbf{x}\in\|f\|:\, \mathbf{x}_Y=\mathbf{y}} f(\mathbf{x}) & \text{if } \mathbf{y} \in \pi_Y(\|f\|) \\ 0 & \text{otherwise.} \end{cases}$$

Finally, we make use of the notation $f^{\downarrow\emptyset}$ for $\sum_{\mathbf{x}\in\mathbf{X}} f(\mathbf{x})$ so that $f^{\downarrow\emptyset} = 1$.

2.3. Conditional independence

Given a probability distribution $f(V)$, let X, Y, Z be disjoint subsets of V and let $U = X \cup Y \cup Z$. The sets X and Y are *independent given Z under $f^{\downarrow U}$* if

$$f^{\downarrow U} \times f^{\downarrow Z} = f^{\downarrow X\cup Z} \times f^{\downarrow Y\cup Z}\,. \tag{1}$$

Note that if $X = \emptyset$ or $Y = \emptyset$, then the equality in eq. (1) trivially holds.

Assume that neither X nor Y is the empty set. If $Z = \emptyset$, eq. (1) requires that, for every configuration $\mathbf{u}$ of U, one has

$$f^{\downarrow U}(\mathbf{u}) = f^{\downarrow X}(\mathbf{u}_X) \times f^{\downarrow Y}(\mathbf{u}_Y)\,.$$

If $Z \neq \emptyset$, eq. (1) requires that, for every configuration $\mathbf{u}$ of U, one has

$$f^{\downarrow U}(\mathbf{u}) \times f^{\downarrow Z}(\mathbf{u}_Z) = f^{\downarrow X\cup Z}(\mathbf{u}_{X\cup Z}) \times f^{\downarrow Y\cup Z}(\mathbf{u}_{Y\cup Z})$$

or, equivalently,

$$f^{\downarrow U}(\mathbf{u}) = \begin{cases} \dfrac{f^{\downarrow X\cup Z}(\mathbf{u}_{X\cup Z}) \times f^{\downarrow Y\cup Z}(\mathbf{u}_{Y\cup Z})}{f^{\downarrow Z}(\mathbf{u}_Z)} & \text{if } \mathbf{u} \in \|f^{\downarrow U}\| \\ 0 & \text{otherwise.} \end{cases}$$

If X and Y are independent given Z under $f^{\downarrow U}$, we also say that the *conditional independence*

$$X \perp\!\!\!\perp Y \mid Z$$

holds under $f(V)$. We call this conditional independence *trivial* if $X = \emptyset$ or $Y = \emptyset$, and *nontrivial* otherwise. It can be proved that conditional independence satisfies the so-called *semigraphoid axioms* [15].

2.4. Composition of probability distributions

A probability distribution $g(X)$ is *composable* [19] with a probability distribution $h(Y)$ if

(a) either $X \cap Y = \emptyset$, or

(b) $g^{\downarrow X\cap Y}$ is dominated by $h^{\downarrow X\cap Y}$.

By Lemma 2.3, condition (b) is equivalent to

(b') $\pi_{X\cap Y}(\|g\|) \subseteq \pi_{X\cap Y}(\|h\|)$.

Note that if there exists a probability distribution $f(X \cup Y)$ such that $f^{\downarrow X} = g(X)$ and $f^{\downarrow Y} = h(Y)$ then $g(X)$ is composable with $h(Y)$.

Assume that $g(X)$ is composable with $h(Y)$. Let $V = X \cup Y$ and $Z = X \cap Y$. The *composition* of $g(X)$ with $h(Y)$ [19] is the distribution $f(V)$ with support $\|f\| = \|g\| \bowtie \|h\|$ which, for every V-tuple $\mathbf{v} \in \|f\|$, takes on the value

$$f(\mathbf{v}) = \begin{cases} g(\mathbf{v}_X) \times h(\mathbf{v}_Y) & \text{if } Z = \emptyset \\ \dfrac{g(\mathbf{v}_X) \times h(\mathbf{v}_Y)}{h^{\downarrow Z}(\mathbf{v}_Z)} & \text{otherwise.} \end{cases}$$

If $g(X)$ is not composable with $h(Y)$, then the composition of $g(X)$ with $h(Y)$ is *undefined.*

The next two results [19] provide key properties of the composition of distributions.

Theorem 2.4. Let $g(X)$ and $h(Y)$ be probability distributions such that $g(X)$ is composable with $h(Y)$, and let $f(X \cup Y)$ be the composition of $g(X)$ with $h(Y)$.

(i) $f^{\downarrow X} = g(X)$;

(ii) $f(X \cup Y)$ equals the composition of $f^{\downarrow X}$ with $f^{\downarrow Y}$.

Theorem 2.5. Let $g(X)$ and $h(Y)$ be probability distributions such that $g(X)$ is composable with $h(Y)$. The (possibly trivial) conditional independence

$$X \setminus Y \perp\!\!\!\perp Y \setminus X \quad | \quad X \cap Y$$

holds under the composition of $g(X)$ with $h(Y)$.

After Jiroušek [4], we make use of the notation "$g(X) \triangleright h(Y)$" for the composition of $g(X)$ with $h(Y)$. Of course, in general the composition operator "$\triangleright$" is neither commutative nor associative.

3. COMPOSITIONAL EXPRESSIONS

A *compositional expression* [20] is a parenthesized expression formed out by (not necessarily distinct) non-empty sets of variables, and the symbol "$\triangleright$". Explicitly, the following provides a formal definition of a compositional expression:

(i) a set of variables is a compositional expression;

(ii) if θ and η are compositional expressions, then $(\theta)\triangleright(\eta)$ is a compositional expression;

(iii) there are no other compositional expressions than those defined by (i) and (ii).

The *base sequence* of a compositional expression θ is the sequence of all the sets of variables featured in θ arranged in order of appearance. Let $(X_1, \ldots, X_n)$, $n \geq 1$, be the base sequence of θ. We call X_i the ith term of θ, $1 \leq i \leq n$, and the set $X_1 \cup \ldots \cup X_n$ the *frame* of θ. Note that a set featured in θ may have more than one occurrence, that is, it may happen that for two distinct terms X_i and X_j of θ we have that $X_i = X_j$.

Henceforth, a compositional expression of either form $(X)\triangleright(\theta)$ or $(\theta)\triangleright(X)$ or $(X)\triangleright(Y)$ will be written simply as $X \triangleright (\theta)$ or $(\theta) \triangleright X$ or $X \triangleright Y$, respectively.

A compositional expression θ with base sequence $(X_1, \ldots, X_n)$ can be naturally viewed as a string of symbols taken from the set $S = \{(,), \triangleright, X_1, \ldots, X_n\}$. Let $\theta = a_1 \ldots a_l$, where $a_h \in S$ for all h; a *subexpression* of θ is a compositional expression of the form $a_h \ldots a_k$ for some h and k, $1 \leq h \leq k \leq l$. A subexpression θ' of θ is *atomic* if it is of the form $\theta' = X_i$ for some i. As was proved in [19], there exist exactly n atomic subexpressions of θ and $n-1$ non-atomic subexpressions of θ.

Example 3.1. Consider the compositional expression

$$\theta = ABC \triangleright ((AB \triangleright AC) \triangleright (BDE \triangleright CDF)) .$$

The base sequence and the frame of θ are (ABC, AB, AC, BDE, CDF) and $ABCDEF$ respectively. The atomic subexpressions of θ are the five terms of θ, and the non-atomic subexpressions of θ are the following four compositional expressions

$$ABC \triangleright ((AB \triangleright AC) \triangleright (BDE \triangleright CDF))$$

$$(AB \triangleright AC) \triangleright (BDE \triangleright CDF) \qquad AB \triangleright AC \qquad BDE \triangleright CDF .$$

3.1. Syntax tree

The syntactic structure of a compositional expression θ with base sequence $(X_1, \ldots, X_n)$ can be represented by an (ordered) binary tree $\mathcal{T}$, to be called the *syntax tree* for θ [19, 20], whose leaves correspond one-to-one to the n atomic subexpressions $X_1, \ldots, X_n$ of θ, and whose internal nodes correspond one-to-one to the $n-1$ non-atomic subexpressions of θ. Note that each internal node v of $\mathcal{T}$ has two children and, if $(\theta') \triangleright (\theta'')$ is the subexpression corresponding to v, then the child of v corresponding to θ' is called the *left child* of v and the child of v corresponding to θ'' is called the *right child* of v. We choose to direct the arcs of $\mathcal{T}$ away from the root of $\mathcal{T}$; thus, $v \to u$ means that u is a child of v or, equivalently, v is the parent of u. In what follows, for each node v of $\mathcal{T}$, by θ_v we denote the subexpression of θ corresponding to v. Accordingly, the root of $\mathcal{T}$ is the node corresponding to θ; moreover, if v is an internal node with left child u and right child w, then $\theta_v = (\theta_u) \triangleright (\theta_w)$. It should be noted that the node set of $\mathcal{T}$ can be linearly ordered as follows. Let u and w be two distinct nodes of $\mathcal{T}$, and let v be the deepest common ancestor of u and w. Then u *precedes* w if either $u = v$ (that is, u is an ancestor of w) or the left child of v is an ancestor of u (and the right child of v is an ancestor of w). Accordingly, the leaves of $\mathcal{T}$ are ordered in such a way that the ith leaf of $\mathcal{T}$ corresponds precisely to the atomic subexpression given by the ith term (X_i) of θ. Finally, we *label* each node v of $\mathcal{T}$ with the frame of θ_v, which will be denoted by L_v. Thus, if $(X_k, \ldots, X_m)$ is the base sequence of θ_v, then $X_k, \ldots, X_m$ are the labels of the leaves of the subtree $\mathcal{T}_v$ of $\mathcal{T}$ rooted at v and $L_v = X_k \cup \ldots \cup X_m$.

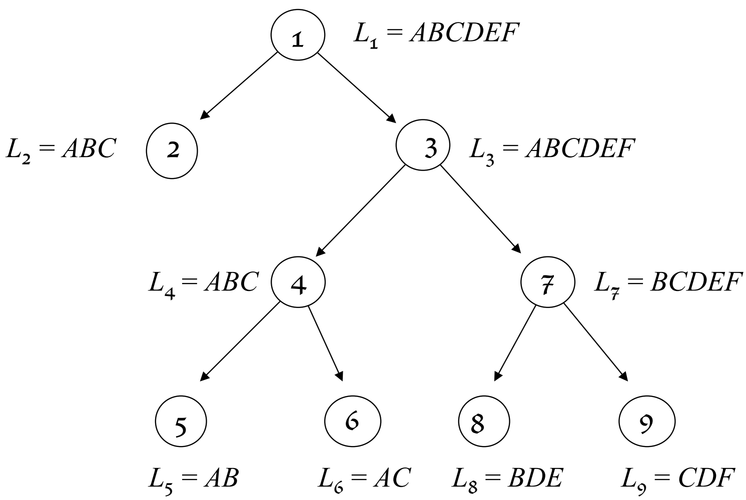

Fig. 1. The syntax tree for the compositional expression $ABC \triangleright ((AB \triangleright AC)) \triangleright (BDE \triangleright CDF)$.

Figure 1 shows the syntax tree $\mathcal{T}$ for the compositional expression of Example 3.1 and Table 1 reports the nodes of $\mathcal{T}$ with the corresponding subexpressions.

In what follows, we will need to go through a syntax tree $\mathcal{T}$. To achieve this, we will perform the *postorder traversal* [1] of $\mathcal{T}$ during which, for each internal node v, we visit first the nodes of the subtree of $\mathcal{T}$ rooted at the left child of v, next the nodes of the subtree of $\mathcal{T}$ rooted at the right child of v and then v. For example, during the postorder traversal of the syntax tree of Fig. 1 the nodes are visited in the following order:

$$2 \quad 5 \quad 6 \quad 4 \quad 8 \quad 9 \quad 7 \quad 3 \quad 1.$$

node v	*subexpression* θ_v
1	$ABC \triangleright ((AB \triangleright AC) \triangleright (BDE \triangleright CDF))$
2	ABC
3	$(AB \triangleright AC) \triangleright (BDE \triangleright CDF)$
4	$AB \triangleright AC$
5	AB
6	AC
7	$BDE \triangleright CDF$
8	BDE
9	CDF

Tab. 1. The subexpressions of the compositional expression $ABC \triangleright ((AB \triangleright AC)) \triangleright (BDE \triangleright CDF)$ corresponding to the nodes of its syntax tree.

Finally, the *leftmost branch* of a syntax tree $\mathcal{T}$ is the subtree of $\mathcal{T}$ induced by the node set that is recursively defined as follows:

- the root of $\mathcal{T}$ is a node of the leftmost branch of $\mathcal{T}$;
- if v is an internal node of the leftmost branch of $\mathcal{T}$, then the left child of v is a node of the leftmost branch of $\mathcal{T}$.

Of course, the leftmost branch of $\mathcal{T}$ has exactly one leaf, which is the leaf of $\mathcal{T}$ labeled with the first term (X_1) of θ; moreover, each internal node of the leftmost branch of $\mathcal{T}$ has exactly one child. For example, the leftmost branch of the syntax tree shown in Fig. 1 has node set $\{1, 2\}$.

3.2. Interpretations

Let θ be a compositional expression with base sequence $(X_1, \ldots, X_n)$ and syntax tree $\mathcal{T}$. An *interpretation* of θ is a sequence $I = (f_1(X_1), \ldots, f_n(X_n))$ of probability distributions.

Given an interpretation $I = (f_1(X_1), \ldots, f_n(X_n))$ of θ, for each node v of $\mathcal{T}$ the *value* of the subexpression θ_v *under* I is recursively defined as follows:

- if v is a leaf of $\mathcal{T}$ and $\theta_v = X_i$ for some i, $1 \leq i \leq n$, then the value of θ_v under I is $f_i(X_i)$;
- if v is an internal node of $\mathcal{T}$ with left child u and right child w, then the value of θ_v under I is the composition of the value of θ_u with the value of θ_w under I.

Of course, if v is a leaf of $\mathcal{T}$, the value of θ_v under I is defined. Consider now an internal node v of $\mathcal{T}$ with left child u and right child w. Then, the value of θ_v under I is defined if

- the values of θ_u and θ_w under I are both defined, and

– either $L_u \cap L_w = \emptyset$ or the marginal on $L_u \cap L_w$ of the value of θ_u under I is dominated by the marginal on $L_u \cap L_w$ of the value of θ_w under I.

If this is the case, we denote the value of θ_v under I by $I[\theta_v]$.

An interpretation I of θ is *valid* if for the root v of $\mathcal{T}$ the value of θ_v under I is defined. If this is the case, $I[\theta_v]$ provides the value of θ under I, and we write simply $I[\theta]$ instead of $I[\theta_v]$.

3.3. A validity test

Let $I = (f_1(X_1), \ldots, f_n(X_n))$ be an interpretation of a compositional expression θ, and let $\mathcal{T}$ be the syntax tree for θ. A procedure for testing I for validity was given in [20] and is now recalled for the sake of completeness. That procedure takes as input the supports of the probability distributions in I and performs a postorder traversal of $\mathcal{T}$ during which, for every vertex v, the value of θ_v under I is checked to be defined, and if this is not the case then we stop the traversal of $\mathcal{T}$ and conclude that I is not a valid interpretation of θ. Of course, if v is a leaf of $\mathcal{T}$, then $\theta_v = X_i$ for some i and the value of θ_v under I is defined so that the support of $I[\theta_v]$ is $\|f_i\|$. Consider the case that v is an internal node of $\mathcal{T}$, and let u and w be the left child and right child of v respectively. Assume that the values of both θ_u and θ_w under I are defined. Then the value of θ_v under I is defined if and only if either $L_u \cap L_w = \emptyset$ or $I[\theta_u]^{\downarrow L_u \cap L_w}$ is dominated by $I[\theta_w]^{\downarrow L_u \cap L_w}$, which by Lemma 2.3 can be checked by testing the inclusion

$$\pi_{L_u \cap L_w}(\|I[\theta_u]\|) \subseteq \pi_{L_u \cap L_w}(\|I[\theta_w]\|)\,.$$

If this is the case, the support of $I[\theta_v]$ is given by the join of the supports of $I[\theta_u]$ and $I[\theta_w]$:

$$\|I[\theta_v]\| = \|I[\theta_u]\| \bowtie \|I[\theta_w]\|\,. \tag{2}$$

3.4. Properties of $I[\theta]$

Let I be a valid interpretation of a compositional expression θ with syntax tree $\mathcal{T}$. We begin by providing a join expression for the support of $I[\theta]$. By repeated application of eq. (2) we have that, for each node v of $\mathcal{T}$, if $(X_k, \ldots, X_m)$, $k \leq m$, is the base sequence of θ_v, then the support of $I[\theta_v]$ is given by

$$\|I[\theta_v]\| = \|f_k\| \bowtie \cdots \bowtie \|f_m\| \tag{3}$$

so that the support of $I[\theta]$ is given by the join expression

$$\|f_1\| \bowtie \cdots \bowtie \|f_n\|\,. \tag{4}$$

We now show that it may happen that, for some i, the support $\|f_i\|$ is redundant in the join expression (4) and can be omitted. For example, consider the compositional expression θ of Example 3.1, and let $I = (f_1(ABC), f_2(AB), f_3(AC), f_4(BDE), f_5(CDF))$ be a valid interpretation of θ. By (4), for the support of $I[\theta]$ we have the join expression

$$\|f_1\| \bowtie \|f_2\| \bowtie \|f_3\| \bowtie \|f_4\| \bowtie \|f_5\|\,. \tag{5}$$

We now show that the validity of I also entails that the supports of $f_2(AB)$ and $f_3(AC)$ are redundant in the join expression (5). Bearing in mind that the left child and the right child of the root 1 of the syntax tree (see Fig. 1) are 2 and 3 respectively, for the support of $I[\theta]$ we have

$$\|I[\theta]\| = \|I[\theta_2]\| \bowtie \|I[\theta_3]\|$$

and, by eq. (3), the supports of $I[\theta_2]$ and $I[\theta_3]$ are given by

$$\|I[\theta_2]\| = \|f_1\| \qquad \|I[\theta_3]\| = \|f_2\| \bowtie \|f_3\| \bowtie \|f_4\| \bowtie \|f_5\|\,.$$

Since the value of θ under I is defined, $I[\theta_2]$ is composable with $I[\theta_3]$, which implies that $I[\theta_2]$ is is dominated by $I[\theta_3]^{\downarrow ABC}$ so that by Lemma 2.3 we have

$$\|f_1\| \subseteq \pi_{ABC}(\|f_2\| \bowtie \|f_3\| \bowtie \|f_4\| \bowtie \|f_5\|)\,.$$

By part (ii) of Remark 2.1 we have

$$\pi_{ABC}(\|f_2\| \bowtie \|f_3\| \bowtie \|f_4\| \bowtie \|f_5\|) = \|f_2\| \bowtie \|f_3\| \bowtie \pi_{BC}(\|f_4\| \bowtie \|f_5\|)$$

and by part (i) of Remark 2.1 we have

$$\|f_2\| \bowtie \|f_3\| \bowtie \pi_{BC}(\|f_4\| \bowtie \|f_5\|) \subseteq \|f_2\| \bowtie \|f_3\|\,.$$

Therefore, we have

$$\|f_1\| \subseteq \|f_2\| \bowtie \|f_3\|$$

so that by part (i) of Remark 2.1 we have

$$\|f_1\| \bowtie \|f_2\| \bowtie \|f_3\| = \|f_1\|$$

which allows to reduce the join expression (5) to

$$\|f_1\| \bowtie \|f_4\| \bowtie \|f_5\|\,.$$

The next result is a consequence of Theorem 2.4 and states a useful property of the marginals of $I[\theta]$ on the labels of nodes of the leftmost branch of $\mathcal{T}$.

Lemma 3.2. Let θ be a compositional expression with frame V and syntax tree $\mathcal{T}$, and let I be a valid interpretation of θ. For each internal node v of the leftmost branch of $\mathcal{T}$, if u and w are the left child and right child of v in $\mathcal{T}$, then

(i) $I[\theta_v]^{\downarrow L_u} = I[\theta_u]$;

(ii) $I[\theta_v] = I[\theta_v]^{\downarrow L_u} \triangleright I[\theta_v]^{\downarrow L_w}$.

By Lemma 3.2, the following holds.

Theorem 3.3. Let θ be a compositional expression with frame V and syntax tree $\mathcal{T}$, and let I be a valid interpretation of θ. For each node v of the leftmost branch of $\mathcal{T}$, we have that (*i*) $I[\theta]^{\downarrow L_v} = I[\theta_v]$;

(*ii*) if v is an internal node and u and w are its left child and right child in $\mathcal{T}$, then

$$I[\theta]^{\downarrow L_v} = I[\theta]^{\downarrow L_u} \triangleright I[\theta]^{\downarrow L_w} .$$

3.5. Evaluation

Let θ be a compositional expression with syntax tree $\mathcal{T}$, and let I be a valid interpretation of θ. A brute-force approach to the evaluation of $I[\theta]$ consists in performing the postorder traversal of $\mathcal{T}$ during which, for each node v, the value of θ_v under I is computed when v is visited. Ultimately, we obtain $I[\theta]$.

A more efficient evaluation method [20] consists in constructing a (symbolic) algebraic expression for $I[\theta]$, which allows to compute $I[\theta]$ without passing through the probability distributions $I[\theta_v]$ for the non-root nodes of $\mathcal{T}$. For example, given a valid interpretation

$$I = (f_1(ABC), f_2(AB), f_3(AC), f_4(BDE), f_5(CDF))$$

of the compositional expression θ of Example 3.1, the algebraic expression for $I[\theta]$ is given by (see Example 4.1 below)

$$\frac{f_1(ABC) \times f_4(BDE) \times f_5(CDF)}{f_5^{\downarrow D} \times \sum_D \dfrac{f_4^{\downarrow BD} \times f_5^{\downarrow CD}}{f_5^{\downarrow D}}} \tag{6}$$

and the numeric value of $I[\theta](\mathbf{abcdef})$, for every configuration $\mathbf{abcdef} \in \|I[\theta]\|$, is calculated as follows:

$$I[\theta](\mathbf{abcdef}) = \frac{f_1(\mathbf{abc}) \times f_4(\mathbf{bde}) \times f_5(\mathbf{cdf})}{f_5^{\downarrow D}(\mathbf{d}) \times \sum_{\mathbf{d}'} \dfrac{f_4^{\downarrow BD}(\mathbf{bd}') \times f_5^{\downarrow CD}(\mathbf{cd}')}{f_5^{\downarrow D}(\mathbf{d}')}} .$$

In the next section, we recall the procedure given in [20] for constructing the algebraic expression for $I[\theta]$ and add some refinements.

4. THE COMPOSITION SCHEME

Let θ be a compositional expression with base sequence $(X_1, \ldots, X_n)$ and syntax tree $\mathcal{T}$, and let $I = (f_1(X_1), \ldots, f_n(X_n))$ be a valid interpretation of θ. The algebraic expression for $I[\theta]$ is obtained by performing the postorder traversal of $\mathcal{T}$ during which, for each node v of $\mathcal{T}$, we construct a product expression $\mathtt{P}(v)$ for $I[\theta_v]$. Ultimately, the algebraic expression for $I[\theta]$ will be derived from $\mathtt{P}(v)$, where v is the root of $\mathcal{T}$.

It should be noted that, if $J = (g_1(X_1), \ldots, g_n(X_n))$ is another valid interpretation of θ, then the algebraic expression for $J[\theta]$ can be obtained from the algebraic expression for $I[\theta]$ simply by replacing each occurrence of f_i with g_i. In other words, for every (valid interpretation) I of θ, the formal structure of the algebraic expression for $I[\theta]$ is a property of θ only, which we call the *composition scheme* for θ. For example, the composition scheme for the compositional expression of Example 3.1 is given by the formal ratio of the algebraic expression (6).

4.1. The procedure

During the postorder traversal of the syntax tree $\mathcal{T}$ for θ, for each node v of $\mathcal{T}$ we construct a product expression $\mathtt{P}(v)$ for $I[\theta_v]$, which is always in reduced form thanks to three reduction rules: *delete*, *cancel* and *factor out* [20], and looks like as a product of *factors* each of which is of one of the following four types:

$$f_i(X_i) \qquad f_i^{\downarrow Y} \qquad \frac{1}{f_i(X_i)} \qquad \frac{1}{f_i^{\downarrow Y}} \, .$$

Here, $f_i(X_i)$ stands either for a probability distribution from I (in which case $i \le n$), or for an "extra distribution" (in which case $i > n$) which is introduced in order to facilitate the application of above-mentioned reduction rules. In detail, $\mathtt{P}(v)$ is obtained as follows.

Case 1: v is a leaf of $\mathcal{T}$. In this case there exists a unique value of i, $1 \le i \le n$, such that v corresponds to the ith term of θ and, then, $\mathtt{P}(v)$ is set to $f_i(X_i)$.

Case 2: v is an internal node of $\mathcal{T}$. Let u and w be the left child and the right child of v, respectively. Then $\mathtt{P}(v)$ is obtained by simplifying the product

$$\mathtt{P}(u) \times \mathtt{P}(w) \times \frac{1}{\displaystyle\sum_{A \in L_w \setminus Lu} \mathtt{P}(w)} \, .$$

Explicitly, the construction of $\mathtt{P}(v)$ takes the following three steps.

Step 1. We first find the deepest node x of the leftmost branch of $\mathcal{T}_w$ for which $L_u \cap L_w \subseteq L_x$. At this point, if $L_u \cap L_w = L_x$, then we set $\mathtt{S} = \mathtt{P}(x)$ and go to Step 2; otherwise, we reduce the sum $\sum_{A \in L_x \setminus Lu} \mathtt{P}(x)$ using the above-mentioned reduction rules (see [20] for details). After doing that, it may happen that one or more factors in the reduction of the sum $\sum_{A \in L_x \setminus Lu} \mathtt{P}(x)$ is neither of the type $f_i(X_i)$ nor of the type $f_i^{\downarrow Y}$, where $f_i(X_i)$ is either a distribution from I or an extra distribution that has been previously introduced; in this case, it is of the type $\sum_Z(\cdot)$ and, then, we introduce one more extra distribution $f_j(X_j)$, for some $j > n$, which represents the argument of the sum $\sum_Z$, and express that factor as $f_j^{\downarrow X_j \setminus Z}$. Finally, we set $\mathtt{S}$ to the result of the reduction of the sum $\sum_{A \in L_x \setminus Lu} \mathtt{P}(x)$.

Step 2. We reduce the product $\mathtt{P}(w) \times \frac{1}{\mathtt{S}}$ by canceling factors common to $\mathtt{P}(w)$ and $\mathtt{S}$. Let $\mathtt{R}$ denote the result of the reduction of $\mathtt{P}(w) \times \frac{1}{\mathtt{S}}$.

Step 3. We set $\mathtt{P}(v) = \mathtt{P}(u) \times \mathtt{R}$.

Finally, the composition scheme for θ is obtained from $\mathtt{P}(v)$, where v is the root of $\mathcal{T}$, by expressing the extra distributions present in $\mathtt{P}(v)$ (if any) in terms of the probability distributions in I. If no extra distribution is present in $\mathtt{P}(v)$, then $\mathtt{P}(v)$ itself provides the composition scheme for θ.

Example 4.1. Consider again the compositional expression

$$\theta = ABC \triangleright ((AB \triangleright AC) \triangleright (BDE \triangleright CDF))$$

of Example 3.1, and let $I = (f_1(ABC), f_2(AB), f_3(AC), f_4(BDE), f_5(CDF))$ be any valid interpretation of θ. In order to obtain the composition scheme for θ, we first perform the postorder traversal of the syntax tree $\mathcal{T}$ for θ (shown in Fig. 1), during which the product expression $\mathtt{P}(v)$ is constructed for each node v of $\mathcal{T}$. The result is reported in Table 3. Note that $\mathtt{P}(1)$ contains the marginal on BC of the extra distribution $f_6(BCD) = \dfrac{f_4^{\downarrow BD} \times f_5^{\downarrow CD}}{f_5^{\downarrow D}}$ introduced when $\mathtt{P}(3)$ was constructed (see **Step 1**).

node v	$\mathtt{P}(v)$
2	$f_1(ABC)$
5	$f_2(AB)$
6	$f_3(AC)$
4	$\dfrac{f_2(AB) \times f_3(AC)}{f_3^{\downarrow A}}$
8	$f_4(BDE)$
9	$f_5(CDF)$
7	$\dfrac{f_4(BDE) \times f_5(CDF)}{f_5^{\downarrow D}}$
3	$\dfrac{f_2(AB) \times f_3(AC) \times f_4(BDE) \times f_5(CDF)}{f_3^{\downarrow A} \times f_5^{\downarrow D} \times f_6^{\downarrow BC}}$
1	$\dfrac{f_1(ABC) \times f_4(BDE) \times f_5(CDF)}{f_5^{\downarrow D} \times f_6^{\downarrow BC}}$

Tab. 2. The product expressions constructed during the postorder traversal of the syntax tree.

After visiting the root 1 of $\mathcal{T}$, we obtain the composition scheme (6) from $\mathtt{P}(1)$ by expressing $f_6^{\downarrow BC}$ as $\sum_D \dfrac{f_4^{\downarrow BD} \times f_5^{\downarrow CD}}{f_5^{\downarrow D}}$.

4.2. Reduction of a compositional expression

Let θ be a compositional expression with base sequence $(X_1, \ldots, X_n)$, and let $I = (f_1(X_1), \ldots, f_n(X_n))$ be any (valid) interpretation of θ. We say that a term X_i of θ, for some i, is *redundant* if the composition scheme for θ does not contain any occurrence of f_i, that is, it contains neither $f_i(X_i)$ nor any marginal of $f_i(X_i)$. Note that the first term X_1 of θ is never redundant. Thus, for the compositional expression θ of Example 4.1, since the composition scheme (6) contains no occurrences of f_2 and no occurrences of f_3, the second term ($X_2 = AB$) and the third term ($X_3 = AC$) of θ are redundant. The following is more instructive example.

Example 4.2. Consider the compositional expression

$$\theta = ABC \triangleright (BD \triangleright (AC \triangleright CD)) \qquad I = (f_1(ABC), f_2(BD), f_3(AC), f_4(CD))$$

and let $I = (f_1(ABC), f_2(BD), f_3(AC), f_4(CD))$ be any valid interpretation of θ. For the composition scheme of θ we find

$$\frac{f_1(ABC) \times f_2(BD) \times f_4(CD)}{\sum_C \frac{f_3^{\downarrow C} \times f_4(CD)}{f_4^{\downarrow C}} \times \sum_D \frac{f_2(BD) \times f_4(CD)}{\sum_C \frac{f_3^{\downarrow C} \times f_4(CD)}{f_4^{\downarrow C}}}}$$

and, since each f_i has at least one occurrence, no term of θ is redundant.

The *reduction* of a compositional expression θ is the compositional expression obtained from θ by deleting all redundant terms. A compositional expression θ is *reduced* if θ contains no redundant terms and, in this case, the reduction of θ is itself. For example, the reduction of the compositional expression of Example 4.1 is $ABC \triangleright (BDE \triangleright CDF)$.

4.3. Closed-form composition schemes

We say that the composition scheme for a compositional expression θ has a *closed form* if it is a product expression, which happens if and only if the product expression associated with the root of the syntax tree for θ contains no extra distributions (see the procedure of Section 4.1). We now give two classes of compositional expressions having closed-form composition schemes. To this end, for a given sequence $(X_1, \ldots, X_n)$, $n > 1$, of non-empty sets we make use of the following notation:

$$\partial X_i = \begin{cases} \emptyset & \text{if } i = 1 \\ \left(\cup_{1 \le j \le i-1} X_j\right) \cap X_i & \text{if } i > 1\,. \end{cases}$$

A compositional expression θ with base sequence $(X_1, \ldots, X_n)$ is

- a *sequential expression* [19] if $\theta = ((\ldots(X_1 \triangleright X_2) \triangleright \ldots) \triangleright X_{n-1}) \triangleright X_n$;
- a *canonical expression* [19] if the sequence $(X_1, \ldots, X_n)$ enjoys the *running intersection property* which requires that for each $i > 1$ there exists $j_i < i$ such that $\partial X_i \subseteq X_{j_i}$.

It is routine to check that the composition scheme for the sequential expression with base sequence $(X_1, \ldots, X_n)$ has the following closed form [13, 14]:

$$\prod_{\substack{i=1 \\ \partial X_i \neq X_i}}^{n} \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}}. \tag{7}$$

(Recall that $f_i^{\downarrow \emptyset} = 1$.) Moreover, from a result proven in [19] we have that the composition scheme for a canonical expression with base sequence $(X_1, \ldots, X_n)$ is again given by (7).

Note that by (7) a sequential expression or a canonical expression with base sequence $(X_1, \ldots, X_n)$ is reduced if and only if, for each $i > 1$, one has $\partial X_i \neq X_i$ (in which case we also have that $X_i = X_j$ if and only if $i = j$). Of course, if θ is a reduced, sequential or canonical expression with base sequence $(X_1, \ldots, X_n)$, then the composition scheme (7) reads

$$\prod_{i=1}^{n} \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}}. \tag{8}$$

We can take the composition scheme (8) to define a class of reduced compositional expressions which we may call *regular expressions.* Thus, both reduced sequential expressions and reduced canonical expressions are examples of regular expressions. The following is an example of a regular expression which is neither sequential nor canonical

$$(AB \triangleright AC) \triangleright (BCD \triangleright CE).$$

5. COMPOSITIONAL AND BAYESIAN MODELS

5.1. Compositional models

Let θ be a compositional expression with frame V. We say that a probability distribution $f(V)$ is *conformal to* (or is *represented by*) the (*compositional*) *model generated by* θ if there exists a valid interpretation I of θ such that $I[\theta] = f(V)$. We say that a conditional independence is *valid* in the model generated by a compositional expression θ if it holds under every probability distribution conformal to the model generated by θ. As a consequence of Theorem 2.5 and part (i) of Theorem 3.3 we have the following.

Theorem 5.1. Let θ be a compositional expression with syntax tree $\mathcal{T}$. For each internal node v of the leftmost branch of $\mathcal{T}$, if u and w are the left child and the right child of v in $\mathcal{T}$, then the conditional independence

$$L_u \setminus L_w \perp\!\!\!\perp L_w \setminus L_u \quad | \quad L_u \cap L_w$$

is valid in the model generated by θ.

Starting from the conditional independences mentioned in Theorem 5.1, we can derive many other conditional independences valid in the model generated by θ using semigraphoid axioms.

In what follows, we limit our considerations to models generated by reduced compositional expressions. Moreover, compositional models generated by regular expressions and by reduced sequential expressions will be referred to as *regular compositional models* and *sequential compositional models*, respectively.

5.2. Bayesian models

Let $\mathcal{D}$ be an acyclic digraph (a *dag*, for short) whose vertices represent variables. Let V be the vertex set of $\mathcal{D}$. For each vertex A of $\mathcal{D}$, by $pa(A)$ we denote the (possibly empty) set of parents of A in $\mathcal{D}$. A probability distribution $f(V)$ is conformal to the *Bayesian model* generated by the dag $\mathcal{D}$ if

$$f(V) = \prod_{A \in V} \frac{f^{\downarrow \{A\} \cup pa(A)}}{f^{\downarrow pa(A)}}. \tag{9}$$

From a dual perspective, the Bayesian model generated by $\mathcal{D}$ can be viewed as a representation of conditional independences.

Theorem 5.2. (Lauritzen et al. [16], Pearl [22], Verma and Pearl [25]) Let X, Y and Z be three mutually disjoint sets of vertices of a dag $\mathcal{D}$. The following three statements are pairwise equivalent:

(i) the conditional independence $X \perp\!\!\!\perp Y \mid Z$ is valid in the Bayesian model generated by $\mathcal{D}$;

(ii) X and Y are d-separated by Z in $\mathcal{D}$;

(iii) X and Y are separated by Z in the moral graph of the subgraph of $\mathcal{D}$ induced by the smallest ancestral set containing $X \cup Y \cup Z$.

Recall that

– X and Y are *d-separated* [22] by Z in a dag "if and only if there is no trail between a vertex in X and a vertex in Y along which (1) every node with converging arrows either is in Z or has a descendant in Z and (2) every node that delivers an arrow along the trail is outside Z" [3];

– an *ancestral set* [16] in a dag is a set U of vertices such that $pa(A) \subseteq U$ for all $A \in U$;

– the *moral graph* [16] of a dag $\mathcal{D}$ is the minimal (with respect to the number of edges) graph obtained from the undirected graph underlying $\mathcal{D}$ by adding edges in such a way that, for every vertex A of $\mathcal{D}$, the set $\{A\} \cup pa(A)$ is a clique;

– X and Y are *separated* by Z in an undirected graph if, for every $A \in X$ and $B \in Y$, every path (if any) joining A and B passes through Z.

Lauritzen et al. [16] exploited the equivalence between parts (*i*) and (*iii*) of Theorem 5.2 to devise an algorithm (henceforth referred to as the LDLL algorithm) which decides the validity of $X \perp\!\!\!\perp Y \mid Z$ in $O(|V|^2)$ time, where V is the vertex set of $\mathcal{D}$.

Geiger et al. [3] exploited the equivalence between parts (*i*) and (*ii*) of Theorem 5.2 to devise an algorithm (henceforth referred to as the GVP algorithm) which finds the maximal set U for which $X \perp\!\!\!\perp U \mid Z$ is valid in $O(|E|)$ time, where E is the set of directed edges of $\mathcal{D}$. Therefore, a conditional independence $X \perp\!\!\!\perp Y \mid Z$ is valid in the Bayesian model on $\mathcal{D}$ if and only if $Y \subseteq U$.

It should be noted that, since $|E| \leq |V|^2$, the GVP algorithm is a bit more efficient than the LDLL algorithm; moreover, as was observed in [3], the LDLL algorithm can be used to solve the GVP problem in $O(|V|^3)$ time.

6. RECURSIVE FACTORIZATION MODELS

In this section we introduce a class of factorization models that in some sense are equivalent to both sequential compositional models and Bayesian models.

Let $(X_1, \ldots, X_n)$ be a set sequence such that $\partial X_i \neq X_i$ for all i, and let $V = X_1 \cup \ldots \cup X_n$. We say that a probability distribution $f(V)$ *recursively factorizes according to* $(X_1, \ldots, X_n)$ or, equivalently, $f(V)$ is conformal to the *recursive* (*factorization*) *model* generated by $(X_1, \ldots, X_n)$ if

- for every configuration $\mathbf{v}$ of V, $f(\mathbf{v}) \neq 0$ if and only if $f^{\downarrow X_i}(\mathbf{v}_{X_i}) \neq 0$ for each i, which by Lemma 2.3 is equivalent to saying that

$$\|f\| = \pi_{X_1}(\|f\|) \bowtie \cdots \bowtie \pi_{X_n}(\|f\|)\,;$$

- for every configuration $\mathbf{v} \in \|f\|$, the value of $f(\mathbf{v})$ is given by

$$f(\mathbf{v}) = \prod_{i=1}^{n} \frac{f^{\downarrow X_i}(\mathbf{v}_{X_i})}{f^{\downarrow \partial X_i}(\mathbf{v}_{X_i})}\,.$$

We also say that a probability distribution $f(V)$ conformal to the recursive model generated by $(X_1, \ldots, X_n)$ has *factorization scheme*

$$\prod_{i=1}^{n} \frac{f^{\downarrow X_i}}{f^{\downarrow \partial X_i}}\,. \tag{10}$$

Example 6.1. Consider the recursive model generated by the set sequence

$$(ABC, ADE, CEFG)\,.$$

A probability distribution $f(ABCDEFG)$ conformal to the model has the factorization scheme

$$\frac{f^{\downarrow ABC} \times f^{\downarrow ADE} \times f^{\downarrow CEFG}}{f^{\downarrow A} \times f^{\downarrow CE}}\,.$$

6.1. Recursive models vs. sequential compositional models

We prove that recursive models have the same representation power as sequential compositional models. To achieve this, we need the following lemma.

Lemma 6.2. Every probability distribution conformal to a compositional model generated by a regular expression is conformal to the recursive model generated by its base sequence.

Proof. Let θ be a regular expression with frame V and base sequence $(X_1, \ldots, X_n)$. Consider any probability distribution $f(V)$ conformal to the compositional model generated by θ. Then there exists a valid interpretation $I = (f_1(X_1), \ldots, f_n(X_n))$ of θ such that $I[\theta] = f(V)$. We need to prove that

(*i*) $\|f\| = \pi_{X_1}(\|f\|) \bowtie \cdots \bowtie \pi_{X_n}(\|f\|)$, and

(*ii*) $f(V)$ has the factorization scheme (10).

Proof of (*i*). Let $V_i = X_1 \cup \ldots \cup X_{i-1} \cup X_{i+1} \cup \ldots \cup X_n$, $1 \leq i \leq n$. By (4), for each i we have that

$$\begin{aligned}\pi_{X_i}(\|f\|) &= \pi_{X_i}(\|f_1\| \bowtie \cdots \bowtie \|f_n\|)\\ &= \begin{cases} \|f_i\| & \text{if } X_i \cap V_i = \emptyset \\ \|f_i\| \bowtie \pi_{X_i \cap V_i}(\|f_1\| \bowtie \cdots \bowtie \|f_{i-1}\| \bowtie \|f_{i+1}\| \bowtie \cdots \bowtie \|f_n\|) & \text{otherwise}\end{cases}\end{aligned}$$

so that we always have that $\pi_{X_i}(\|f\|) \subseteq \|f_i\|$ and, hence,

$$\pi_{X_1}(\|f\|) \bowtie \cdots \bowtie \pi_{X_n}(\|f\|) \subseteq \|f_1\| \bowtie \cdots \bowtie \|f_n\| = \|f\|\,.$$

On the other hand, by Remark 2.2 one has

$$\|f\| \subseteq \pi_{X_1}(\|f\|) \bowtie \cdots \bowtie \pi_{X_n}(\|f\|)\,.$$

So, we have that

$$\|f\| = \pi_{X_1}(\|f\|) \bowtie \cdots \bowtie \pi_{X_n}(\|f\|)$$

which proves (*i*).

Proof of (*ii*). By the composition scheme (8) of $f(V)$, we have

$$f(V) = f_1(X_1) \times \prod_{i=2}^{n} \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}} \tag{11}$$

so that

$$f^{\downarrow X_1 \cup \ldots \cup X_i} = \begin{cases} f_1(X_1) & \text{if } i = 1 \\ f^{\downarrow X_1 \cup \ldots \cup X_{i-1}} \times \dfrac{f_i(X_i)}{f_i^{\downarrow \partial X_i}} & \text{if } i > 1. \end{cases} \tag{12}$$

It follows that, for $i = 2, \ldots, n$

$$f^{\downarrow X_i} = \sum_{A \notin X_i} f^{\downarrow X_1 \cup \ldots \cup X_i} = \sum_{A \notin X_i} \left(f^{\downarrow X_1 \cup \ldots \cup X_{i-1}} \times \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}} \right)$$
$$= \left(\sum_{A \notin \partial X_i} f^{\downarrow X_1 \cup \ldots \cup X_{i-1}} \right) \times \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}} = f^{\downarrow \partial X_i} \times \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}}$$

so that

$$\frac{f^{\downarrow X_i}}{f^{\downarrow \partial X_i}} = \frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}} \qquad (i = 2, \ldots, n)\,.$$

By replacing $f_1(X_1)$ and $\frac{f_i(X_i)}{f_i^{\downarrow \partial X_i}}$ in eq. (11) with $f^{\downarrow X_1}$ and $\frac{f^{\downarrow X_i}}{f^{\downarrow \partial X_i}}$ respectively, we obtain the factorization scheme (10), which proves (ii). □

It should be noted that, if a compositional expression θ is not regular, then it is not true that every probability distribution $f(V)$ conformal to the compositional model generated by θ is equal to the value of θ under the interpretation $I_f = (f^{\downarrow X_1}, \ldots, f^{\downarrow X_n})$, where $(X_1, \ldots, X_n)$ is the base sequence of θ.

Theorem 6.3. Recursive models have the same representation power as sequential compositional models.

P r o o f. Let $(X_1, \ldots, X_n)$ be a set sequence such that $\partial X_i \neq X_i$ for all $i > 1$. We shall prove that a probability distribution is conformal to the recursive model generated by $(X_1, \ldots, X_n)$ if and only if it is conformal to the compositional model generated by the sequential expression with base sequence $(X_1, \ldots, X_n)$.

(*If*) Since reduced sequential expressions are regular expressions, the statement follows from Lemma 6.2.

(*Only if*) Let $f(V)$ be any probability distribution conformal to the recursive model generated by $(X_1, \ldots, X_n)$, and let θ be the sequential expression with base sequence $(X_1, \ldots, X_n)$. Consider the interpretation

$$I_f = (f^{\downarrow X_1}, \ldots, f^{\downarrow X_n})$$

of θ. Let $\mathcal{T}$ be the syntax tree for θ. We now prove by induction that, for each node v of the leftmost branch of $\mathcal{T}$, the value of θ_v under I is defined and $I_f[\theta_v] = f^{\downarrow L_v}$.

BASIS: v is the leaf of the leftmost branch of $\mathcal{T}$. Then the value of θ_v under I is trivially defined and $I_f[\theta_v] = f_1(X_1)$.

INDUCTION: v is an internal node of the leftmost branch of $\mathcal{T}$. Let u and w be the left child and the right child of v in $\mathcal{T}$, respectively, and let $L_u = X_1 \cup \ldots \cup X_i$ for some $i < n$. By hypothesis, the value of θ_u under I_f is defined and $I_f[\theta_u] = f^{\downarrow L_u}$. Then, the value of θ_w under I_f is trivially defined and $I_f[\theta_w] = f^{\downarrow X_{i+1}}$. Since both $I_f[\theta_u]$ and $I_f[\theta_w]$ are marginals of $f(V)$, $I_f[\theta_u]$ is composable with $I_f[\theta_w]$ and $I_f[\theta_v] = f^{\downarrow L_u} \triangleright f^{\downarrow X_{i+1}}$. By eq. (12) we have $I_f[\theta_v] = f^{\downarrow L_u \cup X_{i+1}}$.

Finally, for the root of $\mathcal{T}$, we have that the value of θ under I_f is defined and $I_f[\theta] = f(V)$, which proves that $f(V)$ is conformal to the compositional model generated by the sequential expression with base sequence $(X_1, \ldots, X_n)$. □

6.2. Recursive models vs. Bayesian models

We prove that recursive models have the same representation power as Bayesian models. To achieve this, we need the following notions.

Given a set sequence $(X_1, \ldots, X_n)$, let $V = X_1 \cup \ldots \cup X_n$ and, for each i, $1 \leq i \leq n$, let

$$k_i = |X_i \setminus \partial X_i| .$$

Consider an ordering $\pi = (A_1, \ldots, A_k)$ of the variables in V, where $k = k_1 + k_2 + \ldots + k_n$, obtained by choosing in order

the variables in $X_1 \setminus \partial X_1 \,(= X_1)$ (in any order),

the variables in $X_2 \setminus \partial X_2$ (in any order),

...

the variables in $X_n \setminus \partial X_n$ (in any order).

Explicitly, π is of the form

$$\pi = (A_1, \ldots, A_{k_1}, A_{k_1+1}, \ldots, A_{k_1+k_2}, \ldots, A_{k-k_n} \ldots, A_k).$$

Given π, let $\mathcal{D}$ be the dag with vertex set V in which, for each h and j with $1 \leq h < j \leq k$, $A_h \to A_j$ is a directed edge if and only if either $A_h, A_j \in X_1$ or there exists $i > 1$ such that $A_h \in X_i$ and $A_j \in X_i \setminus \partial X_i$. We call $\mathcal{D}$ a *dag associated with* $(X_1, \ldots, X_n)$. For example, Figure 2 shows a dag associated with the set sequence $(ABC, ADE, CEFG)$.

Remark 6.4. The number of directed edges of any dag associated with $(X_1, \ldots, X_n)$ is

$$\frac{1}{2} \sum_{i=1}^{n} \left(k_i + 2|\partial X_i| - 1\right) k_i$$

which is $O(|X_1|^2 + \cdots + |X_n|^2)$.

Theorem 6.5. Recursive models have the same representation power as Bayesian models.

Proof. We shall prove that

(*i*) Given a dag $\mathcal{D}$ with vertex set V, there exists a set sequence $(X_1, \ldots, X_n)$, $n = |V|$, such that a probability distribution $f(V)$ is conformal to the Bayesian model generated by $\mathcal{D}$ if and only if $f(V)$ is conformal to the recursive model generated by $(X_1, \ldots, X_n)$.

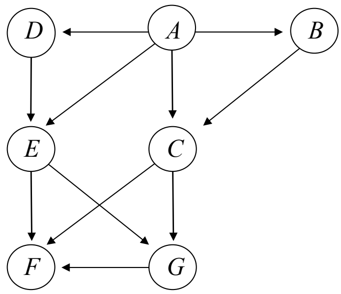

Fig. 2. A dag associated with $(ABC, ADE, CEFG)$.

(ii) Given a set sequence $(X_1, \ldots, X_n)$, there exists a dag $\mathcal{D}$ with vertex set $V = X_1 \cup \ldots \cup X_n$ such that a probability distribution $f(V)$ is conformal to the recursive model generated by $(X_1, \ldots, X_n)$ if and only if $f(V)$ is conformal to the Bayesian model generated by $\mathcal{D}$.

Proof of (i). Let $(A_1, \ldots, A_n)$ be a topological ordering [1] of vertices of $\mathcal{D}$, that is, $pa(A_1) = \emptyset$ and, for each i and j, if $A_i \in pa(A_j)$ then $i < j$. Let $X_i = \{A_i\} \cup pa(A_i)$, $1 \le i \le n$. At this point, it is easily seen that $\partial X_i = pa(A_i)$, $1 \le i \le n$, so that eq. (9) can be re-written as

$$f(V) = \prod_{1 \le i \le n} \frac{f^{\downarrow \{A_i\} \cup pa(A_i)}}{f^{\downarrow pa(A_i)}} ,$$

which entails that $f(V)$ has the factorization scheme (10). So, a probability distribution $f(V)$ is conformal to the Bayesian model generated by $\mathcal{D}$ if and only if $f(V)$ is conformal to the recursive model generated by $(X_1, \ldots, X_k)$.

Proof of (ii). Let $\mathcal{D}$ be a dag associated with $(X_1, \ldots, X_n)$. Then, we have that

$$f^{\downarrow X_1} = \prod_{h=1}^{k_1} \frac{f^{\downarrow \{A_h\} \cup pa(A_h)}}{f^{\downarrow pa(A_h)}}$$

and, for each $i > 1$,

$$\frac{f^{\downarrow X_i}}{f^{\downarrow \partial X_i}} = \prod_{h=k_1+\ldots+k_{i-1}+1}^{k_1+\ldots+k_i} \frac{f^{\downarrow \{A_h\} \cup pa(A_h)}}{f^{\downarrow pa(A_h)}}$$

so that the factorization scheme (10) can be re-written as

$$\prod_{1 \le h \le k} \frac{f^{\downarrow \{A_h\} \cup pa(A_h)}}{f^{\downarrow pa(A_h)}} .$$

So, a probability distribution $f(V)$ is conformal to the recursive model generated by $(X_1, \ldots, X_n)$ if and only if $f(V)$ is conformal to the Bayesian model generated by $\mathcal{D}$. □

Corollary 6.6. Sequential compositional models have the same representation power as Bayesian models.

P r o o f. By Theorems 6.3 and 6.5. □

7. MARKOV PROPERTIES OF RECURSIVE MODELS

We want to answer the following question:

> Given a (reduced) set sequence $(X_1, \ldots, X_n)$ and three mutually disjoint subsets X, Y and Z of $X_1 \cup \ldots \cup X_n$, is the conditional independence $X \perp\!\!\!\perp Y \mid Z$ valid in the recursive model generated by the set sequence $(X_1, \ldots, X_n)$?

By Theorem 6.5, an algorithm for recognizing a valid conditional independence can be obtained applying either the LDLL algorithm or the GVP algorithm to the acyclic dag constructed in the part (*ii*) of the proof of Theorem 6.5. Thus, using the GVP algorithm we obtain a quadratic recognition algorithm by Remark 6.4.

We now present a graphical procedure which runs in linear time. The input of our procedure is the *graph of the sequence* $(X_1, \ldots, X_n)$ which is the (undirected) bipartite graph with $n + |V|$ nodes defined as follows:

- The nodes on one side represent the terms $X_1, \ldots, X_n$ of the sequence and are called *term-nodes*, and the nodes on the other side represent all the variables in V and are called *variable-nodes*;
- each variable-node is labeled with the corresponding variable in V;
- each term-node is both labeled with the corresponding term, say X_i, and tagged with ∂X_i;
- an (unordered) couple (v, t), where v is a variable-node and t is a term-node, is an edge if and only if the label of v belongs to the label of t.

Let $\mathcal{G}$ denote the graph of the sequence $(X_1, \ldots, X_n)$. By an end-point of $\mathcal{G}$ we mean a node with exactly one incident edge.

Selective Reduction Algorithm

Input: The graph $\mathcal{G}$ of the sequence $(X_1, \ldots, X_n)$, and a set U of variables.

Output: A subgraph $\mathcal{H}$ of $\mathcal{G}$.

Step 1. Set $\mathcal{H} := \mathcal{G}$

Step 2. Repeat the following two operations until they cannot longer modify $\mathcal{H}$.

(2.1) If a variable-node v is an end-point of $\mathcal{H}$ and the label of v does not belong to U, then

- delete the label of v from the label of its adjacent term-node, and
- delete v from $\mathcal{H}$.

(2.2) If a term-node t has its label equal to its tag, then delete t from $\mathcal{H}$.

The graph $\mathcal{H}$ will be referred to as the *reduction* of $\mathcal{G}$ *with sacred* U.

Example 7.1. Consider the set sequence $(ABC, ADE, CEFG)$ generating the recursive model of Example 6.1. The graph $\mathcal{G}$ of $(ABC, ADE, CEFG)$ and its reduction $\mathcal{H}$ with sacred CE are shown in Figure 3.

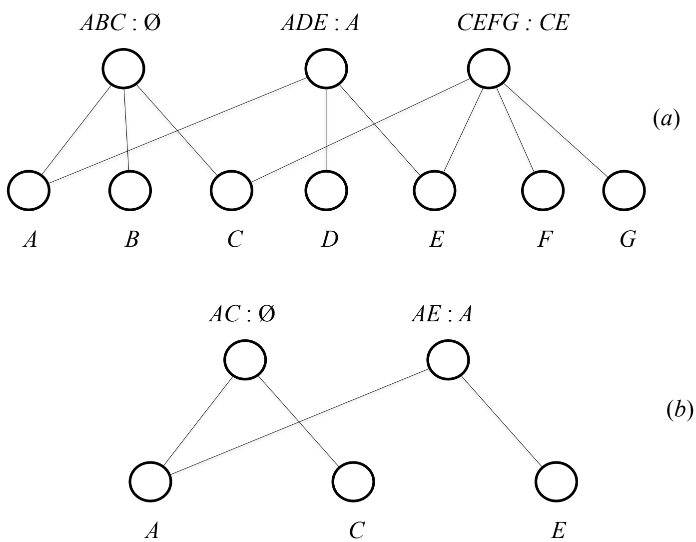

Fig. 3. (a) The graph $\mathcal{G}$ of the sequence $(ABC, ADE, CEFG)$, and (b) the graph $\mathcal{H}$ resulting from the reduction of $\mathcal{G}$ with sacred CE.

Remark 7.2. Let $\mathcal{H}$ be the reduction of $\mathcal{G}$ with sacred X. A variable-node of $\mathcal{H}$ is an end-point if its label belongs to X. Moreover, for each term-node t of $\mathcal{H}$, if t is labeled with L and tagged with ∂X_i for some $i > 1$, then ∂X_i is a proper (possibly empty) subset of L and each variable in $L \setminus \partial X_i$ either belongs to X or is the label of a variable-node u that is joined to an end-point (labeled with a variable belonging to X) of $\mathcal{H}$ by a path that does not pass through t.

In the following lemma, by the "adjacency graph" of a set system $\{Y_1, \ldots, Y_m\}$ we mean the undirected graph with vertex set $Y_1 \cup \ldots \cup Y_m$ and edges of the type (A, B) where $A \neq B$ and $\{A, B\} \subseteq Y_i$ for some i.

Lemma 7.3. Let $\mathcal{G}$ be the graph of the sequence $(X_1, \ldots, X_n)$, let U be a subset of $V = X_1 \cup \ldots \cup X_n$, and let $\mathcal{H}$ be the reduction of $\mathcal{G}$ with sacred U. Let $\mathcal{D}$ be a dag associated with $(X_1, \ldots, X_n)$, and let $An_{\mathcal{D}}(U)$ be the smallest ancestral set containing U in $\mathcal{D}$.

(i) The set of labels of variable-nodes of $\mathcal{H}$ is equal to $An_{\mathcal{D}}(U)$, and

(ii) the moral graph of the subdigraph of $\mathcal{D}$ induced by $An_{\mathcal{D}}(U)$ is the adjacency graph of the set of labels of term-nodes of $\mathcal{H}$.

Proof. Let W be the set of labels of variable-nodes of $\mathcal{H}$.

Proof of *(i)*. If $W = U$ then the statement is trivially true. Assume that $W \setminus U \neq \emptyset$. We need to prove that in $\mathcal{D}$ every variable in $W \setminus U$ is the ancestor of some variable in U. Let $A \in W \setminus U$ and let v be the variable-node of $\mathcal{H}$ labeled with A. By Remark 7.2, v is adjacent to (at least) two term-nodes of $\mathcal{H}$ and, hence, A belongs to the tag of some term-node t of $\mathcal{H}$. Assume that the label of t, denoted by Y, is the residual part of the ith term X_i for some i. Then, the tag of t is given by ∂X_i and $Y \setminus \partial X_i \neq \emptyset$. Let $B \in Y \setminus \partial X_i$. By construction of $\mathcal{D}$, A is a parent of B since $A \in \partial X_i$ and $B \in X_i \setminus \partial X_i$. If $B \in U$ then we are done. Otherwise, let u be the variable-node of $\mathcal{H}$ labeled with B, and let w be a variable-node that is an end-point of $\mathcal{H}$ and is joined to u by a path that does not pass through t (see Figure 4). (A variable-node such as w always exists by Remark 7.2.) Let C be the label of w. Then, in $\mathcal{D}$ the vertex B is an ancestor of C and, since A is a parent of B, A is an ancestor of C. By Remark 7.2, C belongs to U, which proves that A is an ancestor of some variable in U. So, every variable in $W \setminus U$ is the ancestor of some variable in U.

Proof of *(ii)*. Let $\mathcal{D}'$ be the subdigraph of $\mathcal{D}$ induced by $An_{\mathcal{D}}(U)(= W)$, and let (A, B) be an edge of the moral graph of $\mathcal{D}'$. By construction of $\mathcal{D}$, the set $\{A, B\}$ is contained in the label of some term-node of $\mathcal{H}$. On the other hand, if Y is the label some term-node of $\mathcal{H}$ and Y is the residual part of the ith term X_i for some i then, for every two distinct variables $A, B \in Y$, we can have one of the following four cases:

(a) $A, B \in \partial X_i$. By construction of $\mathcal{D}$, for every $C \in Y \setminus \partial X_i$, $A \to C$ and $B \to C$ are directed edges of $\mathcal{D}'$ so that (A, B) is an edge of the moral graph of $\mathcal{D}'$.

(b) $A \in \partial X_i$ and $B \in Y \setminus \partial X_i$. By construction of $\mathcal{D}$, $A \to B$ is a directed edge of $\mathcal{D}'$ so that (A, B) is an edge of the moral graph of $\mathcal{D}'$.

(c) $B \in \partial X_i$ and $A \in Y \setminus \partial X_i$. By construction of $\mathcal{D}$, $B \to A$ is a directed edge of $\mathcal{D}'$ so that (A, B) is an edge of the moral graph of $\mathcal{D}'$.

(d) $A, B \in Y \setminus \partial X_i$. By construction of $\mathcal{D}$, either $A \to B$ or $B \to A$ is a directed edge of $\mathcal{D}'$ so that (A, B) is an edge of the moral graph of $\mathcal{D}'$.

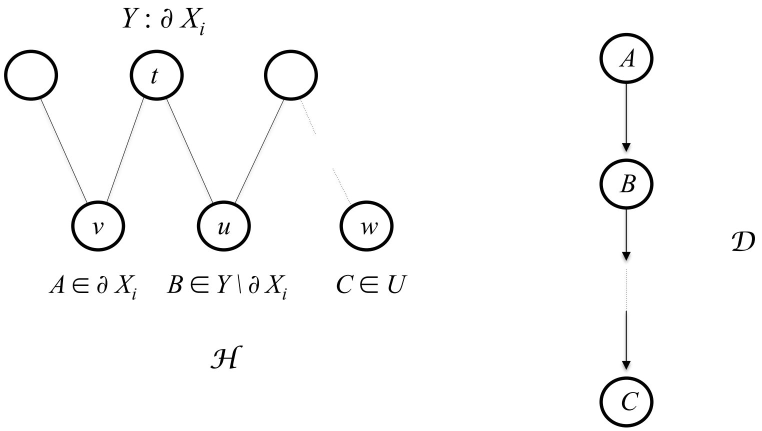

Fig. 4. Illustration of the proof of (i) in Theorem 7.3.

So, in each of the four cases, (A, B) is an edge of the moral graph of $\mathcal{D}'$.

To sum up, (A, B) is an edge of the moral graph of $\mathcal{D}'$ if and only if the set $\{A, B\}$ is contained in the label of some term-node of $\mathcal{H}$ or, equivalently, if and only if (A, B) is an edge of the adjacency graph of the set of labels of term-nodes of $\mathcal{H}$. □

Theorem 7.4. Let $\mathcal{H}$ be the reduction of the graph of $(X_1, \ldots, X_n)$ with sacred $X \cup Y \cup Z$. A conditional independence $X \perp\!\!\!\perp Y \mid Z$ is valid in the recursive model generated by $(X_1, \ldots, X_n)$ if and only if X and Y are separated by Z in $\mathcal{H}$.

Proof. By the equivalence between parts (i) and (iii) of Theorem 5.2 and by Lemma 7.3. □

What remains to do is to check that X and Y are separated by Z in the reduction $\mathcal{H}$ of the graph of $(X_1, \ldots, X_n)$ with sacred $X \cup Y \cup Z$. To achieve this, we ignore labels and tags of term-nodes of $\mathcal{H}$, add a new node s to $\mathcal{H}$ and, for each variable-node v labeled with a variable belonging to X, we add the edge (s, v). Next, we color s and all the variable-nodes as follows: the node s "white", the variable-nodes labeled by variables belonging to Z "black", and the remaining variable-nodes "grey". At this point, we start a *breadth-first search* traversal [1] of $\mathcal{H}$ at s during which we avoid visiting black variable-nodes and change to "white" the color of each variable-node when (and if) it is visited. Eventually, we conclude that $X \perp\!\!\!\perp Y \mid Z$ holds in the recursive model generated by $(X_1, \ldots, X_n)$ if and only if each variable in Y is "grey". The following is an illustrative example.

Example 7.5. We want to decide whether the conditional independence

$$AH \perp\!\!\!\perp CD \mid BG$$

is valid in the recursive model generated by the sequence

$$(AB, BC, DEF, EG, GHI, HL)\,.$$

Given the graph $\mathcal{G}$ of $(AB, BC, DEF, EG, GHI, HL)$, we first construct the reduction $\mathcal{H}$ of $\mathcal{G}$ with sacred $ABCDGH$. Next, we add a node s to $\mathcal{H}$, make s adjacent to the two variable-nodes labeled with A and H, and color the seven variable-nodes: B and G "black", and the remaining variable-nodes "grey". At this point, we start the breadth-first search traversal of $\mathcal{H}$ at s. Ultimately, the variable-nodes of $\mathcal{H}$ are colored as shown in Figure 5. Since the variable-nodes C and D are both "grey", we conclude that the conditional independence $AH \perp\!\!\!\perp CD \mid BG$ is valid in the recursive model generated by $(AB, BC, DEF, EG, GHI, HL)$.

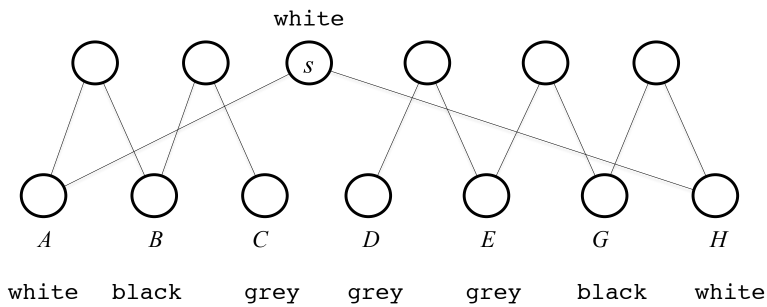

Fig. 5. The output of the validity test with input $(AB, BC, DEF, EG, GHI, HL)$, $X = AH$ and $Z = BG$.

From a computational point of view, our algorithm is linear in the size of the graph of $(X_1, \ldots, X_n)$ since its selective reduction can be performed in linear time [24] and the breadth-first search traversal of $\mathcal{H}$ can be performed in linear time too.

8. FUTURE RESEARCH

We have shown that certain compositional models (explicitly, models generated by reduced sequential expressions) have the same representation power as Bayesian models on acyclic digraphs. In order to make the formalism of compositional models as powerful as other graphical models (e. g, as chain-graph models [15]) we might introduce one more type of compositional (sub)expression such as

$$\theta = X_1 \triangleright (\cdots \triangleright (X_{n-1} \triangleright X_n) \ldots)^* \qquad \text{(for some } n > 1)$$

with the following meaning inspired by the Iterative Proportional Fitting Procedure. Given an interpretation $I = (f_1(X_1), \ldots, f_n(X_n))$ of θ, the value of θ under I is the limit $f^{(\infty)}(V)$, where $V = X_1 \cup \ldots \cup X_n$, of the sequence of probability distributions

$$f^{(0)}(V), f^{(1)}(V), f^{(2)}(V), \ldots$$

where

$$f^{(0)}(V) = f_1(X_1) \triangleright (\cdots \triangleright (f_{n-1}(X_{n-1}) \triangleright f_n(X_n)) \ldots)$$

and

$$f^{(r)}(V) = f_1(X_1) \triangleright (\cdots \triangleright (f_{n-1}(X_{n-1}) \triangleright (f_n(X_n) \triangleright f^{(r-1)}(V))) \ldots)$$

for $r > 0$. Since it is well-known that the Iterative Proportional Fitting Procedure converges if and only if the probability distributions $f_1(X_1), \ldots, f_n(X_n)$ are marginals of some probability distribution on V, we need to add such a consistency constraint to make I a valid interpretation of θ.

Finally, in a forthcoming paper [21] we shall introduce a generalized version of the composition operator which dispenses with the composability requirement.

ACKNOWLEDGEMENT

The author is grateful to an anonymous reviewer whose comments were helpful to improve the organization and readability of the paper.

(Received March 15, 2016)

REFERENCES

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman: Data Structures and Algorithms. Addison-Wesley Pub. Co, Reading 1987.

[2] V. Bína and R. Jiroušek: On computations with causal compositional models. Kybernetika *51* (2015), 525–539. DOI:10.14736/kyb-2015-3-0525

[3] D. Geiger, T. Verma, and J. Pearl: Identifying independence in Bayesian networks. Networks *20* (1990), 507–534. DOI:10.1002/net.3230200504

[4] R. Jiroušek: Composition of probability measures on finite spaces. In: Proc. XIII International Conference on Uncertainty in Artificial Intelligence (D. Geiger and P. P. Shenoy, eds.), Morgan Kaufmann, San Francisco 1997, pp. 274–281. DOI:10.1023/a:1014591402750

[5] R. Jiroušek: Decomposition of multidimensional distributions represented by perfect sequences. Ann. Math. Artif. Intelligence *5* (2002), 215–226. DOI:10.1023/a:1014591402750

[6] R. Jiroušek: Foundations of compositional model theory. Int. J. General Systems *40* (2011), 623–678.

[7] R. Jiroušek: Local computations in Dempster–Shafer theory of evidence. Int. J. Approx. Reasoning *53* (2012), 1155–1167. DOI:10.1016/j.ijar.2012.06.012

[8] R. Jiroušek: On causal compositional models: simple examples. In: Proc. XIV International Conference on Information Processing and Management of Uncertainty in Knowledge-Bases Systems (IPMU 2014) (A. Laurent et al., eds.), Part I, CCIS 442, pp. 517–526. DOI:10.1007/978-3-319-08795-5_53

[9] R. Jiroušek and V. Kratochvíl: Foundations of compositional models: structural properties. Int. J. General Systems *44* (2015), 2–25. DOI:10.1080/03081079.2014.934370

[10] R. Jiroušek and P. P. Shenoy: Compositional models in valuation-based systems. Int. J. Approx. Reasoning *55* (2014), 277–293. DOI:10.1016/j.ijar.2013.02.002

[11] R. Jiroušek and J. Vejnarová: General framework for multidimensional models. Int. J. Approx. Reasoning *18* (2003), 107–127. DOI:10.1002/int.10077

[12] R. Jiroušek, J. Vejnarová, and M. Daniels: Composition models of belief functions. In: Proc. V Symp. on Imprecise Probabilities and Their Applications (G. De Cooman, J. Vejnarová and M. Zaffalon, eds.), Action M Agency, Prague 2007, pp. 243–252.

[13] V. Kratochvíl: Characteristic properties of equivalent structures in compositional models. Int. J. of Approx. Reasoning *52* (2011), 599–612. DOI:10.1016/j.ijar.2010.12.005

[14] V. Kratochvíl: Probabilistic compositional models: solutions of an equivalence problem. Int. J. Approx. Reasoning *54* (2013), 590–601. DOI:10.1016/j.ijar.2013.01.002

[15] S. L. Lauritzen: Graphical Models. Oxford University Press, Oxford 1996.

[16] S. L. Lauritzen, A. P. Dawid, B. N. Larsen, and H.-G. Leimer: Independence properties of directed Markov fields. Networks *20* (1990), 491–505. DOI:10.1002/net.3230200503

[17] D. Maier: The Theory of Relational Databases. Computer Science Press, 1983.

[18] F. M. Malvestuto: A join-like operator to combine data cubes, and answer queries from multiple data cubes. ACM Trans. Database Systems *39*, (2014), 3, 1–31. DOI:10.1145/2638545

[19] F. M. Malvestuto: Equivalence of compositional expressions and independence relations in compositional models. Kybernetika *50* (2014), 322–362. DOI:10.14736/kyb-2014-3-0322

[20] F. M. Malvestuto: Marginalization in models generated by compositional expressions. Kybernetika *51* (2015), 541–570. DOI:10.14736/kyb-2015-4-0541

[21] F. M. Malvestuto: A general composition operator for probabilistic compositional models. Unpublished manuscript (2016).

[22] J. Pearl: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann Pub., San Mateo 1988.

[23] E. Pourabbas and A. Shoshani: Efficient estimation of joint queries from multiple OLAP databases. ACM Trans. Database Systems *32* (2007), 1, Article No. 2. DOI:10.1145/1206049.1206051

[24] R. E. Tarjan and M. Yannakakis: Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. SIAM J. Comput. *13* (1984), 566–579. DOI:10.1137/0213035

[25] T. Verma and J. Pearl: Causal networks: semantics and expressiveness. In: Proc. IV Workshop on Uncertainty in Artificial Intelligence, St. Paul 1988, pp. 352–359.

Francesco M. Malvestuto, Department of Computer Science, Sapienza University of Rome, Via Salaria 113, 00198 Rome. Italy.

e-mail: malvestuto@di.uniroma1.it