Electronic Notes in Theoretical Informatics And Computer Science

ENTICS
https://entics.episciences.org

Volume 5
Proceedings of MFPS 2025

Compositional Inference for Bayesian Networks and Causality

Bart Jacobs[a,1] Márk Széles[a,2] Dario Stein[a,3]

[a] *Radboud University*
Nijmegen, The Netherlands

Abstract

Inference is a fundamental reasoning technique in probability theory. When applied to a large joint distribution, it involves updating with evidence (conditioning) in one or more components (variables) and computing the outcome in other components. When the joint distribution is represented by a Bayesian network, the network structure may be exploited to proceed in a compositional manner — with great benefits. However, the main challenge is that updating involves (re)normalisation, making it an operation that interacts badly with other operations.
String diagrams are becoming popular as a graphical technique for probabilistic (and quantum) reasoning. Conditioning has appeared in string diagrams, in terms of a disintegration, using bent wires and shaded (or dashed) normalisation boxes. It has become clear that such normalisation boxes do satisfy certain compositional rules. This paper takes a decisive step in this development by adding a removal rule to the formalism, for the deletion of shaded boxes. Via this removal rule one can get rid of shaded boxes and terminate an inference argument. This paper illustrates via many (graphical) examples how the resulting compositional inference technique can be used for Bayesian networks, causal reasoning and counterfactuals.

Keywords: inference, Bayesian network, causality, string diagrams, disintegration

1 Introduction

Disintegration is a technique in probability theory that allows one to factorise a joint probability distribution as a product of a marginal distribution and a conditional probability. In traditional probability notation, one writes $P(X) \cdot P(Y \mid X) = P(X,Y) = P(Y) \cdot P(X \mid Y)$. Computing such conditional probabilities is an essential ingredient of Bayesian reasoning.

Because of their basic role, disintegrations (also called conditionals) have been extensively studied in categorical probability theory [8,6,12]. The usual setting for the axiomatic study of disintegration is the theory of Markov categories. These categories model key aspects of probabilistic computation: they can be composed in sequence and in parallel, technically via a symmetric monoidal structure. Crucially, there is also structure for copying and discarding.

Recently, there has been progress in formulating categorical probability theory in terms of so-called CD-categories [25]. CD-categories are just like Markov-categories, but morphisms represent possibly partial

[1] Email: bart@cs.ru.nl
[2] Email: mark.szeles@ru.nl
[3] Email: dario.stein@ru.nl

Rec'd Apr 3,2025; Pub'd Dec 15,2025
10.46298/entics.17029
Proceedings Available Online at ENTICS MFPS 41 Proceedings

computation. For discrete probability theory, this means that a map $X \to Y$ is an X-indexed family of subdistributions — with probabilities adding up to *at most* one instead of to *precisely* one. This partial perspective is more natural for disintegration than the Markov-category approach for two reasons.

(i) Disintegration $P(X,Y) = P(X) \cdot P(Y \mid X)$ may not be well-defined, typically in presence of joint distributions $P(X,Y)$ that do not have full support, that is, when certain elements of the product $X \times Y$ are not in the (support of) the distribution. In the Markov-category approach, the conditional distribution $P(Y \mid X)$ is then allowed to take an arbitrary value. The partial approach involves a least disintegration without any arbitrary, junk information. This order-theoretic perspective will be elaborated elsewhere, but it does play a role here in the background.

(ii) Closely related to the previous point is that Bayesian updating is an inherently partial operation. If one is presented with evidence incompatible with the prior belief, then the update cannot be performed. The formalism of CD-categories can handle such partiality better than Markov categories.

One advantage of categorical probability theory is that it comes with the intuitive graphical calculus of string diagrams. In (an early version of) [20] a notation was introduced for disintegration using bent wires and shaded boxes. These shaded areas describe the part of the diagram that is normalised: the enclosed subdistributions are turned into proper distributions. Normalisation is a peculiar operation that is typically highly non-compositional. Hence these shaded boxes did not seem to match well with the formalism of string diagrams. However, gradually it became clear that there are useful compositional rules for these shaded (normalisation) boxes, see Definition 3.1 below. This was realised by several authors, see for instance [29,15,41,20] (and also implicitly in [12]).

The main contribution of the current paper is to add the final step, pushing the compositionality of these shaded boxes to a new level, so that they become actually useful for graphical reasoning with conditioning. The crucial new rule here is about the removal of shaded boxes, see Proposition 4.3 below. This new removal rule is combined with already existing rules (in Definition 3.1 and Lemma 3.3). This makes it possible to apply disintegration to a Bayesian network by first introducing bent wires and shaded boxes for conditioning, then performing a number of graph rewrite steps, finally ending up with a new network from which shaded boxes are removed. This technique will be illustrated in many examples below, copied from various places in the literature. While the new removal rule is not difficult to prove, it introduces a new feature, namely to return to the world of proper ('non-sub') distributions. This streamlines graphical reasoning about disintegration. For instance, different kinds of interventions treated in [29] can be unified.

We briefly speculate about possible practical advantages of disintegration in Bayesian networks.

(i) Disintegration turns a 'closed' network into an 'open' one, in the terminology of [29]. This means that updating does not happen pointwise, but yields a single function that performs the update for every point at once, see the remarks about the derived update function at the very end of Section 5.

(ii) Moreover, these functions (arising via disintegration) can be pre-computed for specific, often occurring scenarios, e.g. in a medical setting, so that the usual point-updates do not have to be performed real-time. This may improve the usability of inference in Bayesian networks.

This paper is structured as follows. In Section 2 we recall some basics about discrete probability distributions and subdistributions. Section 3 is about the rules of comparators and normalisation. In Section 4 we recall how well-behaved comparators and normalisation give rise to disintegrations. We also prove the derived rule for the disappearence of shaded boxes. The rest of the paper is dedicated to demonstrating the versatility of the extended graphical calculus via a zoo of examples. Section 5 shows examples involving Bayesian networks. Sections 6 and 7 contain probabilistic programming examples on conditional independence and on nested "reasoning about reasoning". Finally, Sections 8 and 9 treat causal and counterfactual reasoning.

Related work

To put this paper's contributions in context, we briefly summarise the main developments that precede our work. The string diagrammatic treatment of Bayesian networks originates in [11]. It was extended with updating in [22], for forward and backward inference. The graphical notation for disintegration using shaded boxes and bent wires was introduced in an earlier version (from 2021) of the book draft [20]. Shaded boxes were used to depict disintegrations in the Kleisli-category of the finitary subdistribution monad $\mathcal{D}$ on the category of sets and functions. Some compositional rules for the shaded box were identified, but a systematic formal treatment was missing.

The authors of the (as of yet unpublished) manuscript [29] took this further by decomposing the notation into cap morphisms and normalisation boxes. Their choice of semantic universe is the category $\mathsf{Mat}_{\mathbb{R}_{\geq 0}}$ of matrices with non-negative entries; it is very similar to our choice of $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$, see Section 2. The disappearence rule of Proposition 4.3 below does not occur in [29]. The authors of [29] define the notion of full support for states (distributions), but not for channels as in Definition 3.4 below.

Examples of causal inference and counterfactual reasoning appeared in string diagrammatic terms in [21]. There, the problem of the current Section 8 is solved using so-called "comb disintegrations". That approach is different from the one adopted in this paper, as it is not internal to the category in which the model lives, but instead relies on an embedding into a larger compact closed category. A category theoretic treatment of Pearl's do-calculus for causal inference appeared in [42]. The related topic of conditional independence and causal compatibility is investigated diagramatically in [14]. In [29, Section 8], a diagrammatic algorithm is presented to identify solutions of a certain class of counterfactual problems.

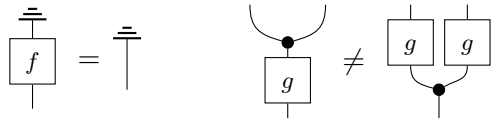
2 Distributions and subdistributions

In this paper we shall work with finite discrete (sub)distributions. We briefly fix notation and recall the essentials. A *subdistribution* on a set X is a finite formal sum of the form $\sum_i r_i|x_i\rangle$, with elements $x_i \in X$ occurring with associated probabilities $r_i \in [0,1]$ satisfying $\sum_i r_i \leq 1$. It is called a *distribution* when $\sum_i r_i = 1$. One can identify a (sub)distribution with a function $\omega\colon X \to [0,1]$ whose support $\mathit{supp}(\omega) \coloneqq \{x \in X \mid \omega(x) \neq 0\}$ is finite and whose sum $\sum_x \omega(x)$ is below or equal to 1. We write $\mathcal{D}_{\leq 1}(X)$ for the set of subdistributions on X, with subset $\mathcal{D}(X) \subseteq \mathcal{D}_{\leq 1}(X)$ of 'proper' distributions. Writing 1 for a one-element set, say $1 = \{0\}$, then $\mathcal{D}(1) \cong 1$ and $\mathcal{D}_{\leq 1}(1) \cong [0,1]$. Thus, functions $X \to \mathcal{D}_{\leq 1}(1)$ can be identified with (fuzzy) predicates. We shall write $\mathbf{1}\colon X \to \mathcal{D}_{\leq 1}(1)$ for the 'truth' function/predicate that maps every element $x \in X$ to the top element $1 \in [0,1]$.

These operations $\mathcal{D}_{\leq 1}$ and $\mathcal{D}$ are both monads on the category **Sets** of sets and functions. We shall write the associated Kleisli categories as $\mathcal{K}\ell(\mathcal{D}) \hookrightarrow \mathcal{K}\ell(\mathcal{D}_{\leq 1})$. Both categories have arbitrary sets X as objects. A morphism f from X to Y in $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ is a function $f\colon X \to \mathcal{D}_{\leq 1}(Y)$. Composition with $g\colon Y \to \mathcal{D}_{\leq 1}(Z)$ is written as $g \circ f\colon X \to \mathcal{D}_{\leq 1}(Z)$ and is defined as $(g \circ f)(x) = \sum_z \big(\sum_y f(x)(y) \cdot g(y)(z)\big)|z\rangle$. The unit map $X \to \mathcal{D}_{\leq 1}(X)$ sends x to the singleton distribution $1|x\rangle$. Morphisms in $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ are called *subchannels*, whereas morphisms in $\mathcal{K}\ell(\mathcal{D})$ are called *channels*. The latter can be recognised inside $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ as those $f\colon X \to \mathcal{D}_{\leq 1}(Y)$ that satisfy $\mathbf{1} \circ f = \mathbf{1}$. This means that each $f(x)$ is a proper distribution and that f restricts to $X \to \mathcal{D}(Y)$. Distributions and subdistributions can be recognised within $\mathcal{K}\ell(\mathcal{D})$ and $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ as the morphisms with a one-element set 1 as domain.

For two subdistributions $\omega \in \mathcal{D}_{\leq 1}(X)$ and $\rho \in \mathcal{D}_{\leq 1}(Y)$ one can form their parallel / tensor product $\omega \otimes \rho \in \mathcal{D}_{\leq 1}(X \times Y)$, given by $\big(\omega \otimes \rho\big)(x,y) = \omega(x) \cdot \rho(y)$. This tensor restricts to $\otimes\colon \mathcal{D}(X) \times \mathcal{D}(Y) \to \mathcal{D}(X \times Y)$. Via pointwise definition it turns the Kleisli categories $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ and $\mathcal{K}\ell(\mathcal{D})$ into symmetric monoidal categories. The tensor unit is the one-element set 1, which is final in $\mathcal{K}\ell(\mathcal{D})$.

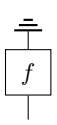
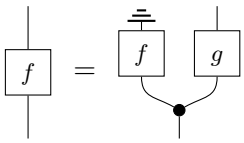
For each set X there is a copy map $\Delta\colon X \to \mathcal{D}(X \times X)$, namely $\Delta(x) = 1|x,x\rangle$. There are first and second projection maps $\mathcal{D}(X) \leftarrow X \times Y \to \mathcal{D}(Y)$, namely $\pi_1(x,y) = 1|x\rangle$ and $\pi_2(x,y) = 1|y\rangle$. These copiers and projections turn both $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ and $\mathcal{K}\ell(\mathcal{D})$ into so-called Copy-Delete (CD) categories, see [6]. The Kleisli category $\mathcal{K}\ell(\mathcal{D})$ is not only a CD-category but also a Markov category, see [12]: all its maps are channels; $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ is called a *partial* Markov category in [25].



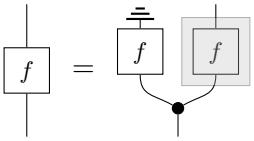
These CD and Markov categories have become the standard universes for categorical probability theory, see *e.g.* [12,6]. They are standardly used with the convenient language of string diagrams. We shall use the basics of this language without further explanation — except that we write copying and discarding as $\Delta = \curlyvee$ and $\mathbf{1} = \bar{\top}$. We refer to [6,12,35] for further details. We will explicitly describe how to do the less standard operations of normalisation, comparison and disintegration (conditioning) within the setting of string diagrams. The only thing we wish to emphasise is that a map / box f is called a *channel* if the equation $\mathbf{1} \circ f = \mathbf{1}$ on the left holds. Further, the copy equation on the right fails in general. The maps g for which it does hold are called *deterministic*. A box without incoming wires is called a *state*.

3 Normalisation and comparison

Any subdistribution $\omega \in \mathcal{D}_{\leq 1}(X)$ that is not the (everywhere) zero subdistribution $\mathbf{0}$ can be normalised to a proper distribution $\frac{1}{\|\omega\|} \cdot \omega$ in $\mathcal{D}(X)$, where $\|\omega\| = \sum_x \omega(x)$ is the *weight* of ω. In this section we will first introduce normalisation and then comparators in string diagrams, following the approach of [29] and [25]. For an arbitrary map $f\colon X \to Y$, written diagrammatically as a box, we define its domain as $dom(f) \coloneqq \mathbf{1} \circ f$. We say that a map $g\colon X \to Y$ is a *normalisation* of f if f can be written as the tuple of its domain with g, as described on the side. The map f is called *normalised* if it is a normalisation of itself.

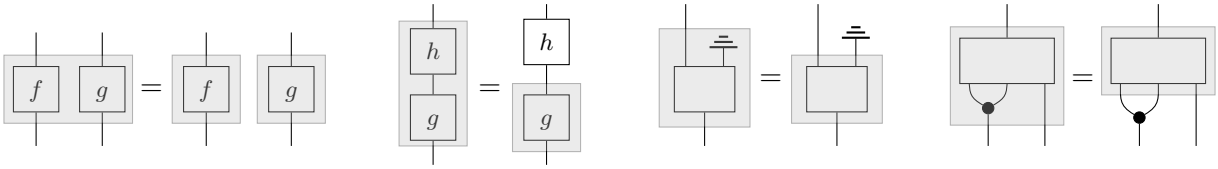
$dom(f) \coloneqq$  

Definition 3.1 ([29, Def. 8]) A normalisation structure on a CD-category assigns to every map $f\colon X \to Y$ a normalised map $nrm(f)\colon X \to Y$, written as a shaded box on the left below, satisfying the equation on the right, making $nrm(f)$ a normalisation of f.

$nrm(f) \coloneqq$   (1)

The following requirements should hold for this normalisation.

(i) Normalisation of parallel composition can be done separately, and a channel h can be pulled out of sequential composition; discarders $\bar{\top}$ and copiers $\curlyvee$ can be pulled out of normalisation boxes:

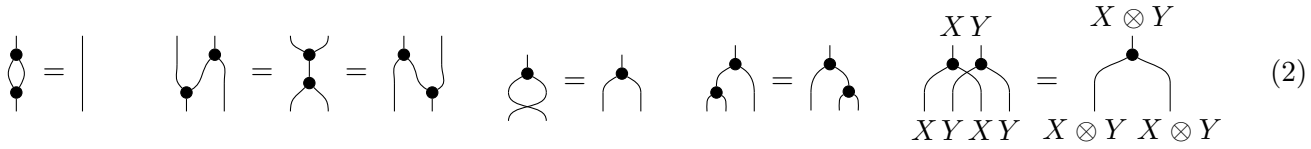


(ii) If f is normalised, that is, if $f = \big((\mathbf{1} \circ f) \otimes f\big) \circ \Delta$, then $nrm(f) = f$. This implies $nrm\big(nrm(g)\big) = nrm(g)$, so that two nested shaded boxes around g can be reduced to a single shaded box.

The second equation in point (i) can actually be derived, with some effort, see [29, Lem. 9], but we include it as a requirement, for simplicity. One can show that $nrm(f)$ is the least normalisation f in the sense that if a map g is a normalisation of f, then g also normalises $nrm(f)$ [29, Prop. 100].

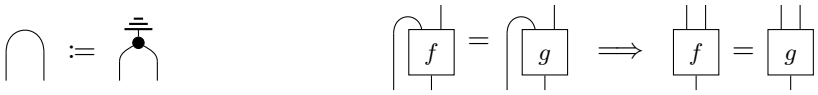
Definition 3.2 [25, Def. 3.23] A comparator structure on a CD-category assigns to each object X a map of the form $\nabla : X \otimes X \to X$, written as $\curlyvee$, satisfying commutativity, associativity, tensor-compatibility,

and the Frobenius equations:



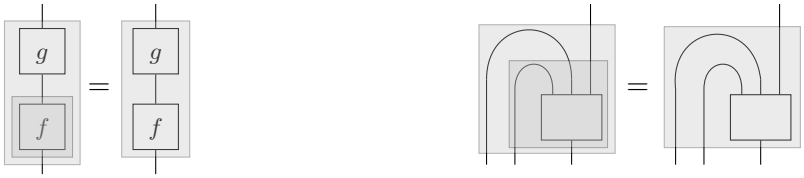
(2)

We then define a *cap* as $\mathbf{1} \circ \nabla \colon X \otimes X \to I$, see on the left below. We further formulate the requirement that these comparators are *cancellative* in terms of the property below on the right.



The second equation below comes from [12, Lem. 11.11]. It can be derived from the first equation.

Lemma 3.3 *In a CD-category with normalisation and cancellative comparators, nested boxes can be reduced to a single box, as on the left below:*



We can now define what 'full support' means for a distribution and a channel, in diagrammatic terms.

Definition 3.4 For a state ω and, more generally, for a map f, in a CD-category, we say that it has *full support* when:

The normalisation $nrm(\omega)$ of a non-zero subdistribution $\omega \in \mathcal{D}_{\leq 1}(X)$ is defined to be $nrm(\omega) = \frac{1}{\|\omega\|} \cdot \omega$. We set $nrm(\mathbf{0}) = \mathbf{0} \in \mathcal{D}_{\leq 1}(X)$. The normalisation structure on $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ is then defined pointwise on subchannels $f : X \to \mathcal{D}_{\leq 1}(Y)$ by $nrm(f)(x) = nrm(f(x))$. It satisfies the axioms of Definition 3.1.

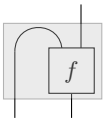
For each set X there is a comparator map $\nabla : X \times X \to \mathcal{D}_{\leq 1}(X)$ in $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$, namely $\nabla(x, x') = 1|x\rangle$ if $x = x'$, and $\nabla(x, x') = \mathbf{0}$ if $x \neq x'$. Comparators in $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ are cancellative. Concretely, this means the following for parallel subchannels $f, g : X \to \mathcal{D}_{\leq 1}(Y \times Z)$. The equation $f = g$ holds if $f(x)(y, z) = g(x)(y, z)$ for all $x \in X$, $y \in Y$, and $z \in Z$.

A subchannel $f : X \to \mathcal{D}_{\leq 1}(Y)$ has full support, according to Definition 3.4, if and only if $f(x)(y) > 0$ for all $x \in X$ and $y \in Y$. This can hold only if the set Y is finite. This notion differs from what is sometimes meant by f having full support, which is that for all $y \in Y$ there exists an $x \in X$ such that $f(x)(y) > 0$, see e.g. [13]. The notion of full support in Definition 3.4 for channels is more restrictive than this, since it requires that every distribution $f(x)$ has full support.

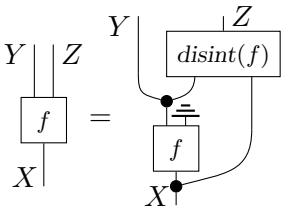
4 Disintegration and daggers

Disintegration is the technique of extracting a conditional probability $P(z \mid y)$ from a joint probability $P(y, z)$, such that $P(z \mid y) \cdot P(y) = P(y, z)$. More generally, this may be done in parameterised form, by extracting $P(z \mid y, x)$ from a joint probability $P(y, z \mid x)$, such that $P(z \mid y, x) \cdot P(y \mid x) = P(y, z \mid x)$. Crucially, disintegration involves normalisation. Its diagrammatic description below is now standard, see *e.g.* [29,25,15,12,6], building on earlier categorical formulations, for instance in [9,8]. The separation of normalisation and comparison for bending gives an intuitive description.

Thus, in a general CD-category with comparators and normalisation, disintegration involves the passage from a map $f\colon X \to Y \otimes Z$ with two outgoing wires into a map $\mathit{disint}(f)\colon Y \otimes X \to Z$ as on the right. We make the crucial property of disintegration explicit, following [29].

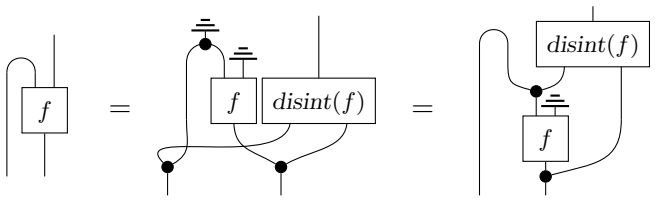
$\mathit{disint}(f) :=$ 

Theorem 4.1 *([29, Prop. 109]) In a CD-category with normalisation and cancellative comparators one can recover a map f from its disintegration in the following manner.*



i.e. $$P(y, z \mid x) = P(z \mid y, x) \cdot P(y \mid x). \tag{3}$$

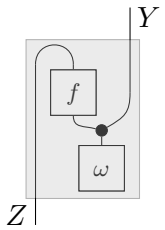
Proof. By applying the cancellativity of caps to the next line, where the first equation is the normalisation property on the right in (1) and the second equation is Frobenius (2).

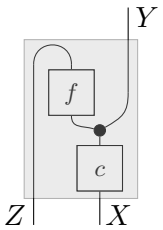


□

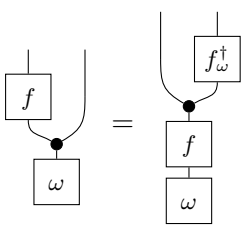
We turn to 'daggers', that is, to reversal of maps. This reversal corresponds to turning a conditional probability $P(y \mid x)$ upside down into $P(x \mid y)$. Such daggers are standardly defined with respect to a 'prior' distribution ω, see [8,6,12]. Here we define them also in parametrised form, with a channel as prior, on the right in (4).

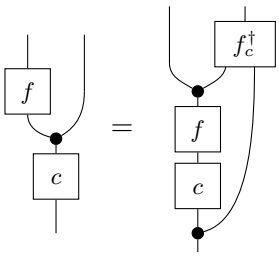
Definition 4.2 Let $f\colon Y \to Z$ be a map in a CD-category. For a state ω on Y and channel $c\colon X \to Y$ we define the dagger $f^\dagger_\omega\colon Z \to Y$ and the parametrised dagger $f^\dagger_c\colon Z \otimes X \to Y$ as disintegrations in (4).

$f^\dagger_\omega :=$ 

$f^\dagger_c :=$ 

(4)





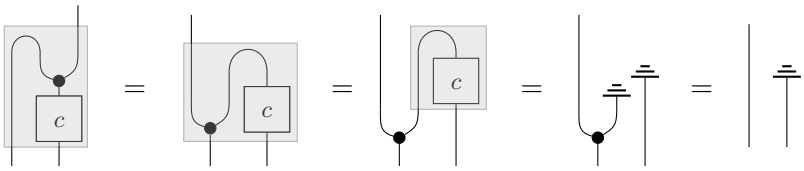
(5)

These daggers satisfy appropriate instantiations of the disintegration equation (3), which we make explicit above on the right.

We now come to our new result, for removal of shaded boxes. It plays a crucial role in the compositional reasoning for disintegration that is developed in this paper.

Proposition 4.3 *In a CD-category with normalisation and cancellative comparators, a state ω and a channel c, both with full support, satisfy:*

Proof. The equation for a state ω on the left is a special case, so we do the proof for the more general case on the right. It uses Frobenius (2), Definition 3.1 (i), and the fact that c has full support.



□

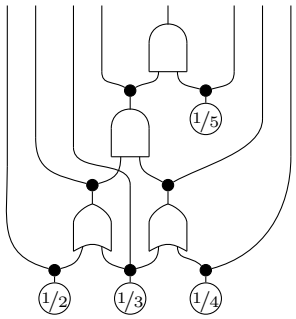
5 Applications to Bayesian networks

This section elaborates two examples of Bayesian networks in order to illustrate how the disintegration rules from the previous two sections can be applied in a compositional manner. These rules turn a Bayesian network with outputs only into a network that also has input wires, so that conditioning can be computed via composition, acting on the evidence as input. There is a fairly straightforward translation of traditional Bayesian network notation into string diagram, see *e.g.* [11,22]: the main difference is that copying is written explicitly as $\curlyvee$ in string diagrams.

5.1 Conditioning for fault trees

Fault trees form a graphical formalism that is used in safety and reliability engineering, see *e.g.* [23] for an overview (or also [34] for a string diagrammatic approach). They can be seen as Bayesian networks, with logical gates as conditional probability tables. We adapt the leading illustration from [28] and present it as a joint distribution τ, visualised on the right.

τ =



(6)

The encircled numbers r at the leafs correspond to coin distributions $\mathit{flip}(r) = r|1\rangle + (1-r)|0\rangle$ on the set $\mathbf{2} = \{0,1\}$. The above or-gates (with a curved bottom) and and-gates (with a straight bottom) are deterministic channels $oc, ac \colon \mathbf{2} \times \mathbf{2} \to \mathcal{D}(\mathbf{2})$ given by $oc(x,y) = 1\big|x \vee y\big\rangle$ and $ac(x,y) = 1\big|x \wedge y\big\rangle$. The joint distribution τ described in (6) is a distribution on $\mathbf{2}^8 = \mathbf{2} \times \cdots \times \mathbf{2}$.

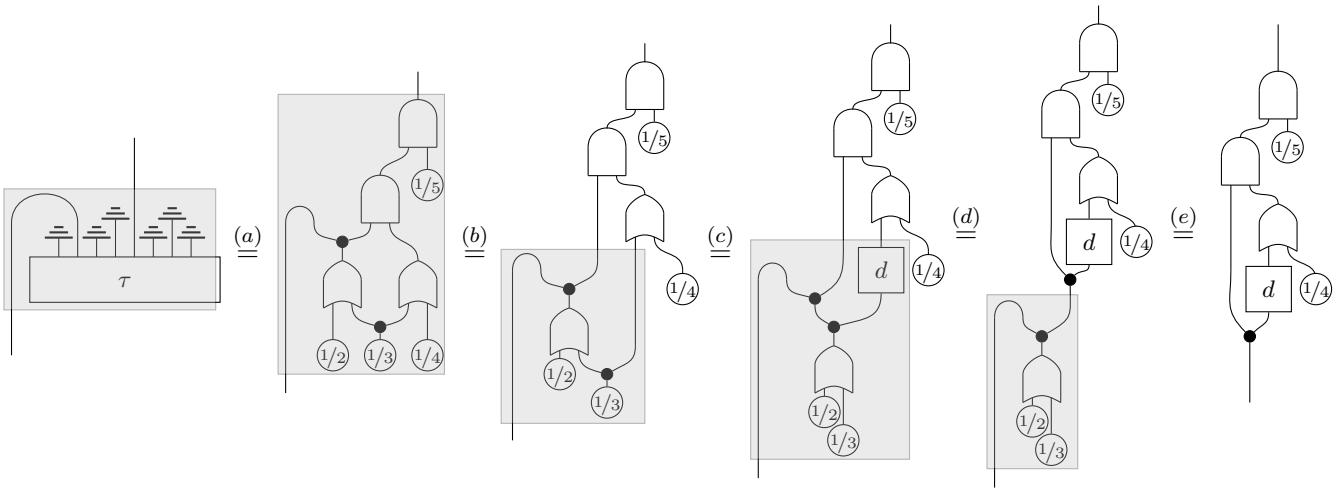


Fig. 1. Disintegration of the Bayesian fault tree network (6), via the introduction of a dagger channel d and via eventual removal of the shaded box, see the beginning of Section 5 for details.

Now suppose we wish to condition on the second wire from the left in (6), and wish to learn the effect on the fifth (top) wire. This means that we can discard all other wires. Figure 1 describes how disintegration

for this Bayesian (fault tree) network works. We elaborate on what happens in a step-by-step manner. Equation (a) on the left involves unpacking the description in (6) and removing the discarded wires. In step (b) channels are pulled out of the shaded box, using Definition 3.1 (i). Subsequently, in step (c), the channel d is introduced as dagger of $oc \circ (\mathit{flip}(1/2) \otimes \mathit{id}) \colon \mathbf{2} \to \mathcal{D}(\mathbf{2})$, with $\mathit{flip}(1/3)$ as prior, according to the left equation in (5). In step (d) the associativity of copying $\curlyvee$ is used and the (dagger) channel d is pulled out of the shaded box, again as in Definition 3.1 (i). Finally, the shaded box is removed entirely in step (e), via Proposition 4.3.

At this stage we have turned the original network (6) with outputs only into an 'open' network with both and input and ouput, at the right of Figure 1. It is not hard to actually compute what this network does, in a compositional manner. The dagger channel d is shown on the right, together with entire disintegrated network, both as channels $\mathbf{2} \to \mathcal{D}(\mathbf{2})$. This last outcome $0 \mapsto 1|0\rangle$ of the disintegrated network is not surprising: if one forces the point in the network where we condition to be 0, the two and-gates above it will produce 0 as output.

$$\begin{cases} d(1) = \frac{1}{2}|1\rangle + \frac{1}{2}|0\rangle \\ d(0) = 1|0\rangle \end{cases} \qquad \begin{cases} 1 \mapsto \frac{1}{8}|1\rangle + \frac{7}{8}|0\rangle \\ 0 \mapsto 1|0\rangle. \end{cases}$$

5.2 Conditioning for the 'Child' Bayesian network

We consider (part of) the 'Child' Bayesian network, a famous example from the literature [38] that is part of a standard repository of Bayesian networks [4]. It involves various medical possibilities and their statistical relationships for the diagnosis of partical child diseases.

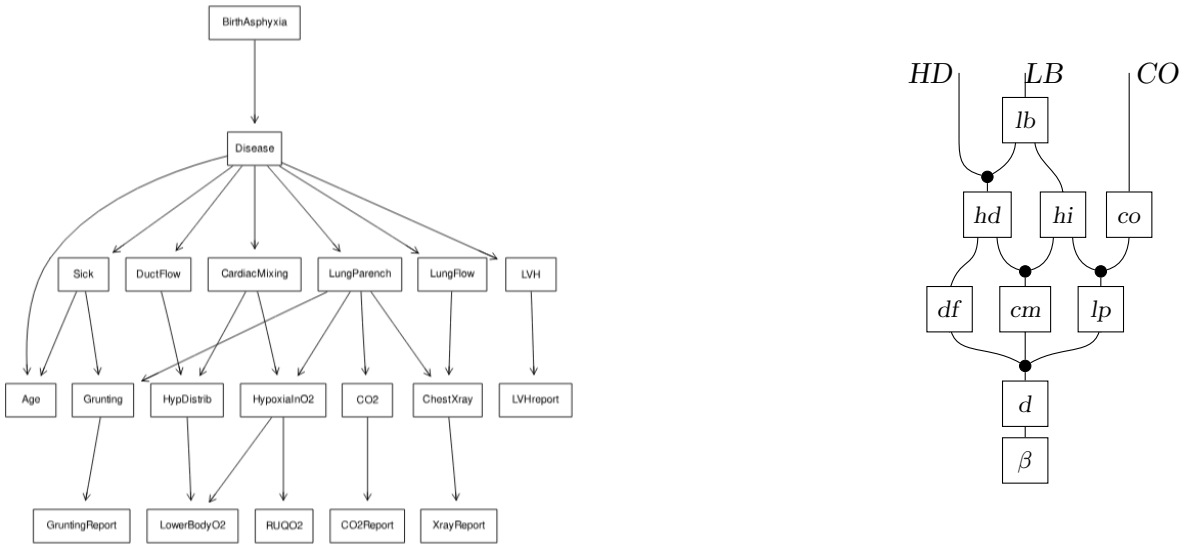


Fig. 2. The original child Bayesian network from [38] on the left (from top to bottom), and the relevant portion of it as string diagram on the right (from bottom to top).

The entire network is given on the left in Figure 2. We are interested in a particular disintegration $HD \times CO \to \mathcal{D}\big(LB\big)$, see below, so we formalise only the relevant part as a string diagram on the right.

The first type is for BirthAsphyxia (BA), which can be present or not. Further, there are sets Disease (DI), with six elements PFC, TGA, Fallot, PAIVS, TAPVD, Lung; DuctFlow (DF) with three elements Lt_to_Rt, None, Rt_to_Lt; CardiacMixing (CM) containing None, Mild, Complete, Transp; LungParench (LP) with Normal, Congested, Abnormal; CO2 (CO) containing Normal, Low, High; HypDistrib (HD) with Equal, Unequal; HypoxiaInO2 (HI) containing Mild, Moderate, Severe; and LowerBodyO2 (LB) containing elements '<5', '5-12', '12+'. We shall use abbreviations for the elements of these sets of the

[4] See www.bnlearn.com/bnrepository/discrete-medium.html#child

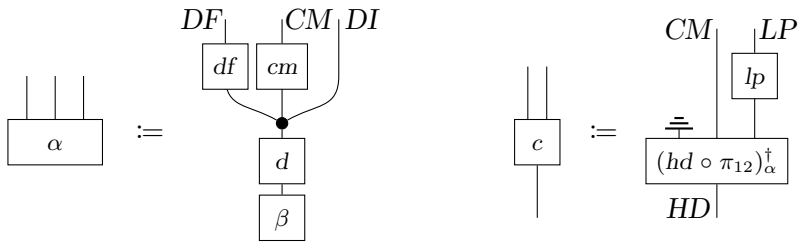
form:

$$\begin{array}{lll} BA = \{ba, ba^{\perp}\} & CM = \{no, mi, co, tr\} & HD = \{eq, eq^{\perp}\} \\ DI = \{pfc, tga, flt, pis, tpd, lng\} & LP = \{nr, cg, ab\} & HI = \{mi, mo, se\} \\ DF = \{lt, no, rt\} & CO = \{nr, lo, hi\} & LB = \{<5, 5\text{-}12, 12+\}. \end{array}$$

The string diagram on the right in Figure 2 involves a distribution $\beta = 0.1|\,ba\rangle + 0.9|\,ba^{\perp}\rangle$ and channels:

$$\begin{array}{c} BA \xrightarrow{d} \mathcal{D}(DI) \qquad DI \xrightarrow{df} \mathcal{D}(DF) \qquad DI \xrightarrow{cm} \mathcal{D}(CM) \qquad DI \xrightarrow{lp} \mathcal{D}(LP) \qquad LP \xrightarrow{co} \mathcal{D}(CO) \\ DF \times CM \xrightarrow{hd} \mathcal{D}(HD) \qquad CM \times LP \xrightarrow{hi} \mathcal{D}(HI) \qquad HD \times HI \xrightarrow{lb} \mathcal{D}(LB) \end{array} \tag{7}$$

The definitions of these channels can be found in Appendix A. Here we look only at how they are used in the Bayesian network on the right of Figure 2. The disintegration $HD \times CO \to \mathcal{D}(LB)$ that we are interested in is elaborated in Figure 3. It uses the following two abbreviations.



(8)

We leave it to the interested reader to recognise the various diagram transformations steps that are applied. We do point out that this disintegration uses two daggers, both with respect to a distribution α and with respect to a channel c, see Definition 4.2 for the two versions.

Eventually, at the bottom-right of Figure 3 we obtain a description of the channel $HD \times CO \to \mathcal{D}(LB)$. Using the detailed channel descriptions from the appendix it can be computed concretely as in (9). The precise medical relevance of these distributions is not our focus. What we do like to point out is that our approach yields the entire disintegration channel, for all inputs. In contrast, in conventional inference in Bayesian networks [32] one obtains outcomes for each of the above lines separately [5]. For instance, the six lines (9) of the extracted channel quickly show that the outcome is determined by the first input element (eq or $eq^{\perp}$) and not by the second input (nr, hi, lo).

$$\begin{array}{rcl} (eq, nr) & \longmapsto & 0.356\,|{<}5\rangle + 0.497\,|5\text{-}12\rangle + 0.147\,|12{+}\rangle \\ (eq, lo) & \longmapsto & 0.356\,|{<}5\rangle + 0.496\,|5\text{-}12\rangle + 0.147\,|12{+}\rangle \\ (eq, hi) & \longmapsto & 0.359\,|{<}5\rangle + 0.489\,|5\text{-}12\rangle + 0.152\,|12{+}\rangle \\ (eq^{\perp}, nr) & \longmapsto & 0.512\,|{<}5\rangle + 0.427\,|5\text{-}12\rangle + 0.061\,|12{+}\rangle \\ (eq^{\perp}, lo) & \longmapsto & 0.512\,|{<}5\rangle + 0.427\,|5\text{-}12\rangle + 0.0611\,|12{+}\rangle \\ (eq^{\perp}, hi) & \longmapsto & 0.505\,|{<}5\rangle + 0.432\,|5\text{-}12\rangle + 0.0631\,|12{+}\rangle. \end{array} \tag{9}$$

Moreover, the channel that we obtain in this way can be pre-computed, so that actual inferences of this kind can be done quickly. Further, if we have an evidence distribution on $HD \times CO$ we can apply the above channel (9) to it via pushforward (Kleisli extension). In this way we perform updating in the style of Jeffrey, see [18,19] for more information.

6 Applications to conditional independence

Disintegration is a technique that shows up in conditional independence arguments. This section will give an illustration, based on an example from the literature. In general, we say that for a map $h\colon Z \to X \otimes Y$

[5] We indeed checked these channel outcomes (9) with six separate inference queries using the *pgmpy* Python library from pgmpy.org.

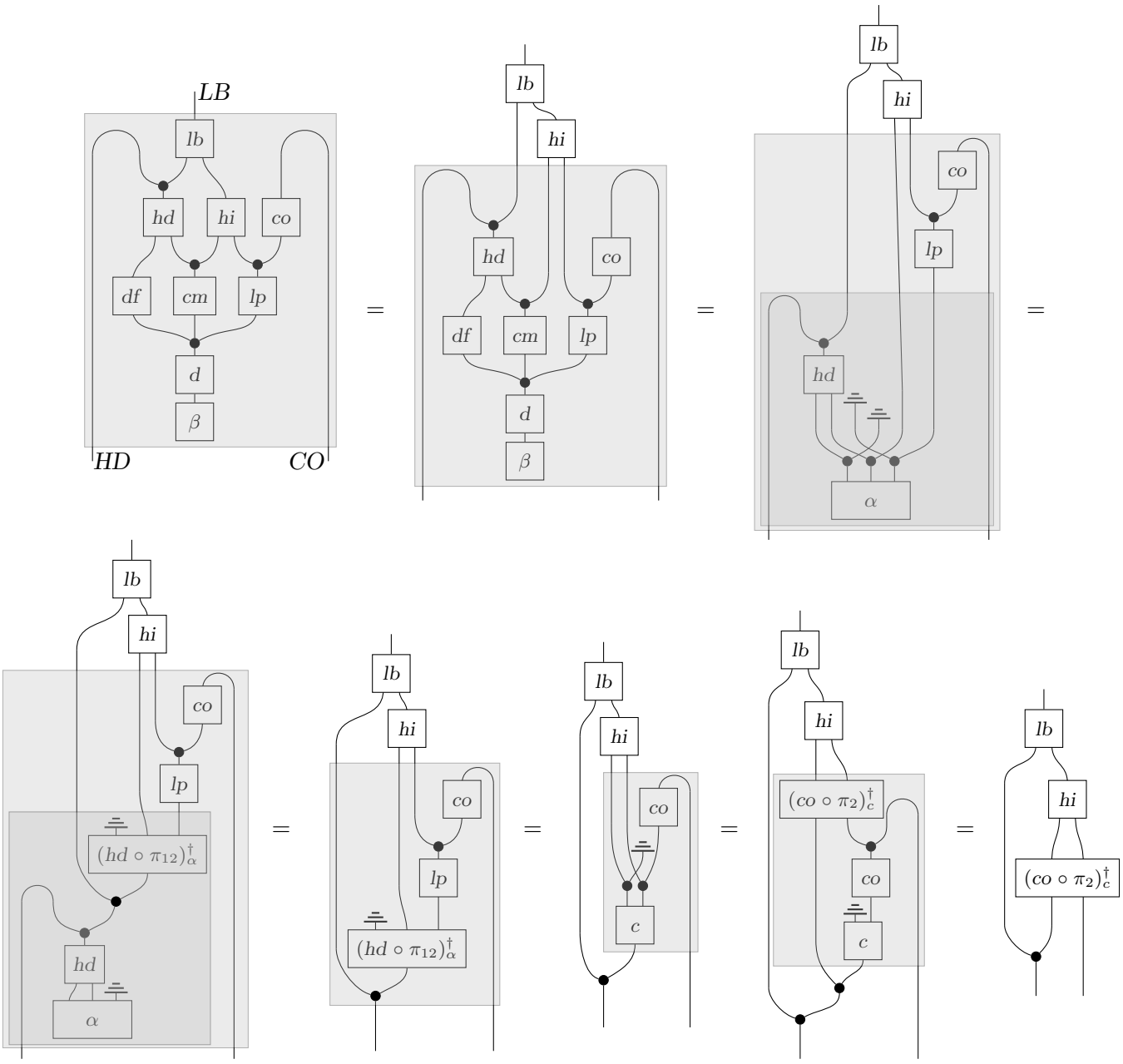
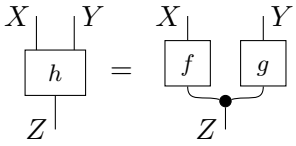


Fig. 3. The disintegration steps for the child Bayesian network on the right in Figure 2, using the abbreviations (8).

conditional independence of X, Y given Z holds, often written as $X \coprod Y \mid Z$, when:

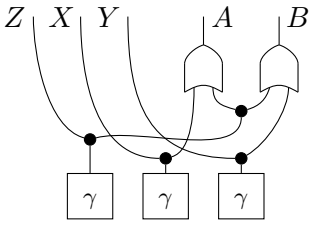


for some f, g. (10)

In [27, (CommonCause)], based on [2, Fig. 6(a)], a particular joint distribution σ on $\mathbf{2}^5$ is defined in a basic probabilistic programming language. We reproduce the code below on the left. It uses $\leftarrow$ for sampling and $\|$ for disjunction. The distribution defined by the program forms a string diagram of the form on the right below.

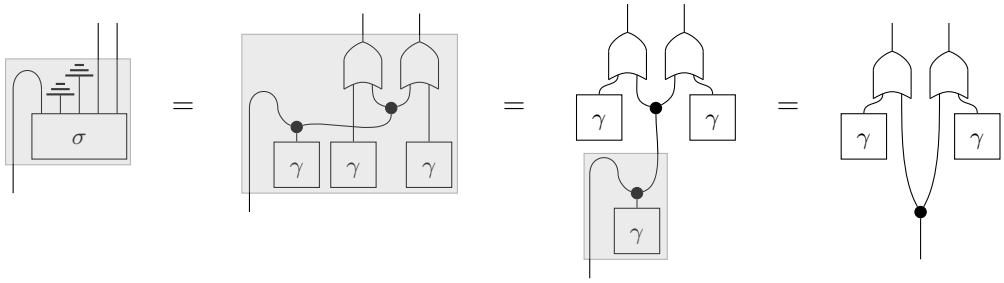
```
Z <- flip(1/2)
X <- flip(1/2)
Y <- flip(1/2)
A = X || Z
B = Z || Y
return (Z, X, Y, A, B)
```

σ =



where $\gamma = \mathit{flip}(\frac{1}{2})$.

It may be seen as a Bayesian network. The claim made in [2,27] is that $A \perp\!\!\!\perp B \mid Z$ via disintegration. A graphical proof looks as follows.



It uses Definition 3.1 (i) and Proposition 4.3. The final diagram above is clearly an instance of the diagram on the right in (10). The proof in [27] is much more concrete, and distinguishes whether the first coin flip (with type Z) produces 1 or 0.

7 Applications to Probabilistic Programming

We have informally used a probabilistic program in the last section to describe a joint distribution over five variables, and have written $\leftarrow$ for sampling. A fully-fledged Bayesian probabilistic programming language (*e.g.* [16,10,4]) adds to this the capacity to condition on data (`observe`), and perform inference (`normalize`). We refer to [30] for an introduction to the subject.

```
alice n = normalize $ do
 loc <- location
 prediction <- bob(n-1)
 condition
 (loc == prediction)
 return loc
```

```
bob 0 = location
bob n = normalize $ do
 loc <- location
 prediction <- alice n
 condition
 (loc == prediction)
 return loc
```

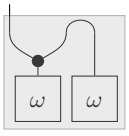
Fig. 4: Pseudocode for the coordination game

From a semantic point of view, probabilistic programs and string diagrams are tightly related and often inter-convertible. Observe can be interpreted as the comparator, and normalize as the shaded box. Conversely, a programming language can be seen as an internal language or type-theoretic calculus for statistical problems [24,39]. Equations such as in Proposition 4.3 can then be understood as admissible program transformations, which preserve the meaning of the program and may cause faster execution.

A striking feature of probabilistic programming is that `normalize` commands can be nested. Such *nested queries* can model agents making inferences about each other's cognitive processes. This kind of reasoning about reasoning is central to communication theory and to the theory of mind [40]. Equations for boxes thereby become tools for such "reasoning about reasoning" in the sense of [43].

First, for an arbitrary distribution $\omega \in \mathcal{D}(X)$ we can similarly define $\omega^2 \in \mathcal{D}(X)$ as on the right. This is the normalisation of the subdistribution $\sum_x \omega(x)^2 |x\rangle$. An iterated version ω^n will be used below. When n grows, this ω^n quickly becomes $1|x\rangle$, where $x \in \mathit{supp}(\omega)$ has the highest probability (if there is one).

$\omega^2 :=$

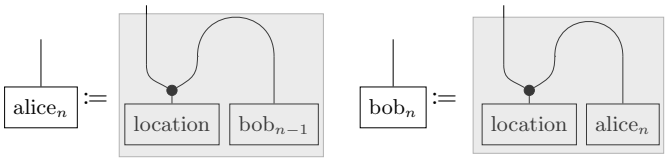


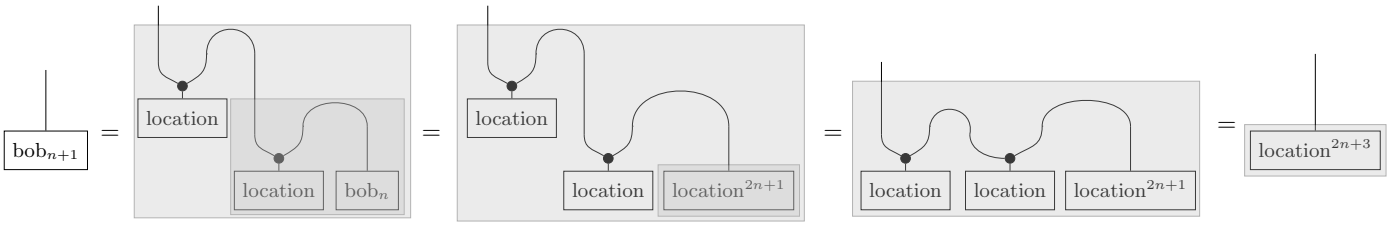
We reproduce a simple example from [40, Section 3.1], of a *coordination game*, originally due to [36]. Two agents (Alice and Bob) want to meet up at one of two locations but did not agree which one and can

not communicate any more. So they are forced to guess where the other person would go — which must recursively take their own behavior into account.

There is a prior distribution 'location' with popularity weights for places to meet. Each agent then simulates the thought process of the other, up to a specified recursion depth n. On Figure 4, we translate the problem into Haskell code with a probabilistic programming framework (such as *MonadBayes* [37]).

The mutually recursive functions `alice` and `bob` perform nested normalisation and inference. Mathematically, we capture the meaning of the program in the string diagrams on the right, where $\text{bob}_0 = \text{location}$. This can be evaluated in terms of the powers from Diagram (7). Via Lemma 3.3, we can eliminate (inner) nested boxes, see below, so that bob_n is equal to the normalisation of location^{2n+1}, where location^{2n+1} is the $(2n+1)$-fold power (comparison) of the location distribution with itself.



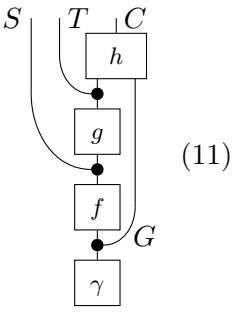


As a result, alice_n is the normalisation of the $2n$-power of location. In the limit $n \to \infty$, both Alice and Bob will end up going to the most popular place, that is, to $s \in \mathit{supp}(\text{location})$ with the highest probability.

8 Applications to causality

We elaborate a leading example from the causality literature in the current framework, from [33], see also *e.g.* [31]. We build on the approach from [21] using string-diagrammatic surgery for intervention. Although the example is well-known and extensively discussed elsewhere, our treatment does have an original element, namely the introduction of the dagger channel, see immediately after (13) for details.

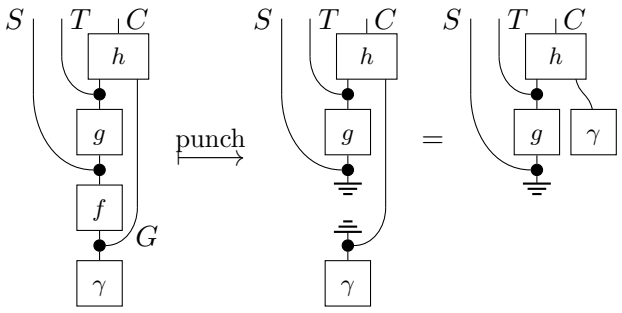
$$\begin{aligned}\sigma = \tfrac{1}{5}|s,t,c\rangle &+ \tfrac{1}{50}|s,t,c^{\perp}\rangle \\ &+ \tfrac{1}{20}|s,t^{\perp},c\rangle \\ &+ \tfrac{1}{10}|s,t^{\perp},c^{\perp}\rangle \\ &+ \tfrac{1}{50}|s^{\perp},t,c\rangle \\ &+ \tfrac{1}{100}|s^{\perp},t,c^{\perp}\rangle \\ &+ \tfrac{1}{10}|s^{\perp},t^{\perp},c\rangle \\ &+ \tfrac{1}{2}|s^{\perp},t^{\perp},c^{\perp}\rangle.\end{aligned} \qquad (11)$$



The example is about a possible causal relation between smoking S, cancer C, and tar T, with a possible confounding role played by genetic consitution G. It starts with a joint distribution $\sigma \in \mathcal{D}\big(S \times T \times C\big)$, involving sets $S = \{s, s^{\perp}\}$ for smoking and non-smoking, and similarly for cancer and tar: $C = \{c, c^{\perp}\}$ and $T = \{t, t^{\perp}\}$. We *know* what this σ is, for instance from some empirical study, see on the left above, and we *assume* that it has the shape described on the right.

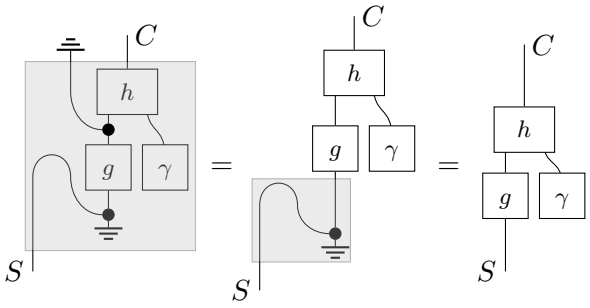
We see the confounding role via the assumed genetic distribution $\gamma \in \mathcal{D}(G)$. The assumed channels f, g, h and the distribution γ can not all be obtained from the joint distribution σ, but we can get something. For instance, the composite $f \circ \gamma$ becomes available when we marginalise out T and C. Similarly, the channel g can be extracted by disintegrating from S to T, see below for details.

We start with the causal surgery procedure, by punching a hole below the S copy bullet, as in:



The leaf $\perp\!\!\!\perp$ is used for a uniform distribution.

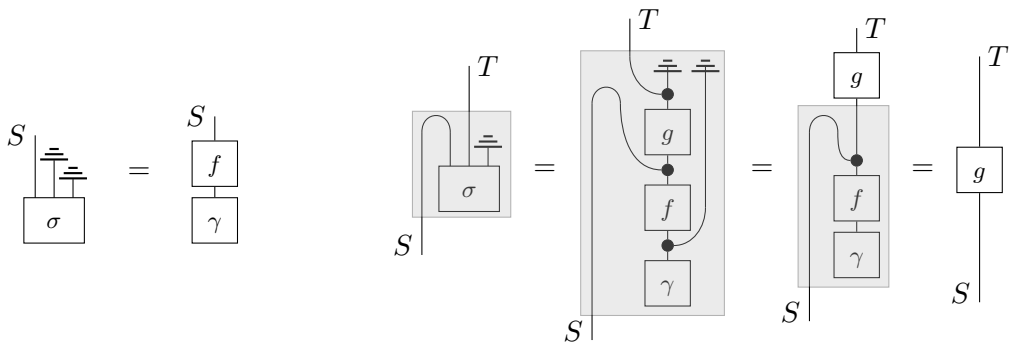
It is good to be aware that this punching happens in a hypothetical string diagram. But it is a way to get the required causal 'do' channel by discarding and disintegration, applied to the diagram above on the right. We calculate using the disintegration rules, especially the box removal rule from Proposition 4.3.



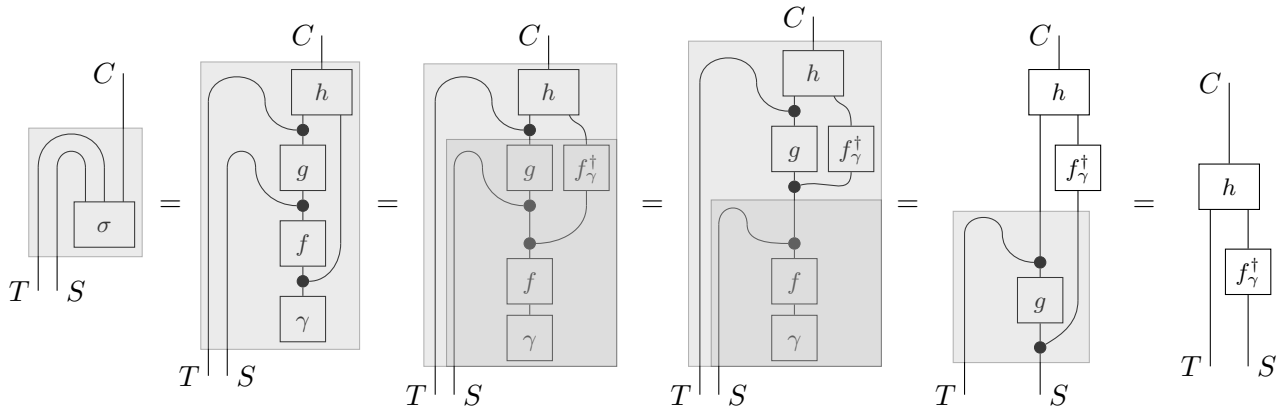
(12)

At this stage the situation is: we may not be able to extract the ingredients of the original joint distribution σ in (11), but we may be able to obtain the ingredients for the causal channel on the right in (12). This works, as we shall demonstrate, in several steps.

We first discard the second two wires of the joint distribution (11), giving the marginal on the left below. Next we extract the channel $g\colon S \to T$ from σ in, using again Proposition 4.3:

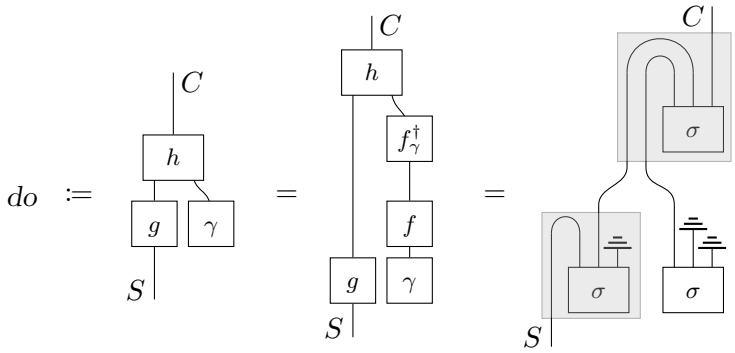


Finally, we extract the channel $S \times T \to \mathcal{D}(C)$ via disintegration:



The most complex step is the second one, which involves both (a) the change from one shaded box to two nested boxes, as in Lemma 3.3, and (b) the introduction of the dagger $f^\dagger_\gamma$ of the channel f, with the distribution γ as prior. The final equation also combined two steps, namely (a) pulling the copier out, as in Definition 3.1 (i), and (b) removing the channel g via Proposition 4.3.

We are now in a position to assemble the different pieces and obtain the causal smoking-to-cancer channel $do\colon S \to C$ from the distribution σ, in the following manner.



(13)

This assembling makes use of one crucial trick, namely the composite $f^\dagger_\gamma \circ f$ in the second equation. This allowed since $\gamma = f^\dagger_\gamma \circ f \circ \gamma$. This follows from the equation on the left in (5), by discarding the first wire on both sides. This do channel $do\colon S \to \mathcal{D}(C)$ in (13) yields the same outcomes as in [21]:

$$do(s) = 0.5423|c\rangle + 0.4577|c^\perp\rangle \qquad do(s^\perp) = 0.2535|c\rangle + 0.7465|c^\perp\rangle.$$

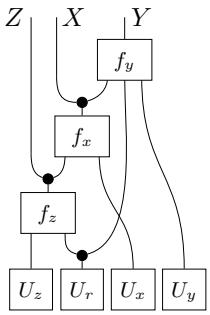
In the end, let us reconsider what happened. We started from a joint distribution $\sigma \in \mathcal{D}\big(S \times T \times C\big)$ in (11) and assumed a certain (inaccessible) internal structure. Via an intervention punch we removed the confounding influence and determined what the resulting intervention map from smoking to cancer would be. It turned out that we can identify this intervention map via purely external manipulations of the original joint distribution σ. The result is a composition, on the right in (13), of these manipulated portions of σ, obtained via disintegration and marginalisation. The resulting intervention channel was then actually computed, via these separate portions.

Finally, we observe that the choice of the mediating channel g matters. One could try to choose the identity channel $g = id$ to get a diagram of the shape (11). In that case, the channel $S \times T \to \mathcal{D}(C)$ could not be extracted, since the identity channel lacks full support. This shows that the notion of full support is crucial when considering causal identifiability.

9 Applications to counterfactuals

Counterfactual reasoning is quite common and follows a 'had' pattern that is easy to recognise, like in: had we left earlier, we would have avoided the traffic jam. Such phrases contain an antecedent (about leaving early) and a conclusion (ending up in a traffic jam), where the antecedent is factually false — we did not leave early — but is still used to draw a conclusion. Such reasoning patterns form a challenge in logic. They change a small part of reality, but leave many other things the same, and then come to a conclusion. In a probabilistic setting, this small change — the 'had' part of the counterfactual — can be modelled as an intervention, of the kind that we have seen in the previous section on causality. We shall describe counterfactual reasoning by duplicating the situation at hand, into a factual and counterfactual world, as proposed in [1] (see also [33]), in such a way that the probabilistic parts are shared, while the (deterministic) mechanisms in both worlds are duplicated but the same. This involves decomposing a probabilistic channel in terms of a deterministic channel together with an 'exogenous' distribution, in which all probability is concentrated. This separation is a general phenomenon that is described as randomness pushback in [12]. The exogenous distributions capture the unobserved background factors that are left unexplained and are simply accepted as they are. This approach is illustrated below in terms of string diagrams, where we shall, once again, exploit that we can simplify string diagrams after intervention, via the discard rules for $\overline{\top}$ and also that we can apply disintegration simplifications via rewriting.

A commonly used example in the (probabilistic) literature on counterfactuals involves people going to a party, with a question: had Bob not gone, would a scuffle not have happened, see *e.g.* [1,21]. We include a different, less familiar illustration, taken from [3, Ex. 27.2]. It involves a counterfactual of the form: had the patient been treated, would they have survived? In [3] the relevant channels are already decomposed into a deterministic channel and an exogenous distribution. In the represenation as string diagram below, there are four such distributions $\mathit{flip}(r) = r|1\rangle + (1-r)|0\rangle \in \mathcal{D}(\mathbf{2})$ with (roughly) the original names.



$$U_r = \mathit{flip}\big(\tfrac{1}{4}\big) \qquad U_z = \mathit{flip}\big(\tfrac{19}{20}\big) \qquad U_x = \mathit{flip}\big(\tfrac{9}{10}\big) \qquad U_y = \mathit{flip}\big(\tfrac{7}{10}\big)$$

$$f_z(a,b) = \begin{cases} 1\big|\,1\big\rangle & \text{if } a = b = 1 \\ 1\big|\,0\big\rangle & \text{otherwise} \end{cases}$$

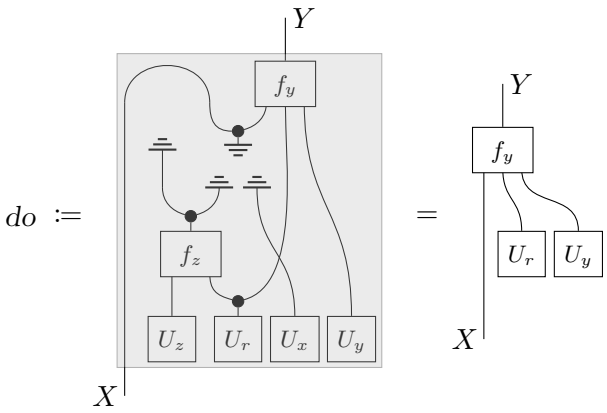
$$f_x(a,b) = \begin{cases} 1\big|\,1\big\rangle & \text{if } a = b = 1 \\ & \text{or } a = b = 0 \\ 1\big|\,0\big\rangle & \text{otherwise} \end{cases}$$

$$f_y(a,b,c) = \begin{cases} 1\big|\,1\big\rangle & \text{if } a = b = 1 \\ & \text{or } a = 0,\, b = c = 1 \\ & \text{or } a = b = c = 0 \\ 1\big|\,0\big\rangle & \text{otherwise} \end{cases}$$

There are three associated deterministic channels $f_z, f_x \colon \mathbf{2} \times \mathbf{2} \to \mathcal{D}\big(\mathbf{2}\big)$ and $f_z \colon \mathbf{2} \times \mathbf{2} \times \mathbf{2} \to \mathcal{D}\big(\mathbf{2}\big)$. They are defined on the left below, and used in the string diagram on the right. The labels / types X, Y, Z are all equal to $\mathbf{2}$, but with different names: Z is for symptoms, X is for treatment, and Y is for survival.

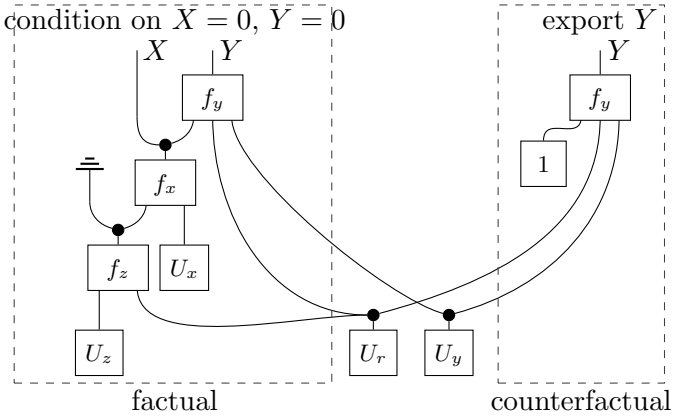
First we are interested in a causal connection between treatment X and survival Y, where we need to take the confounding influence via U_r into account. We thus remove f_X via a punch, discard Z, then

disintegrate the result to get a direct connection from X to Y. This gives the following situation.

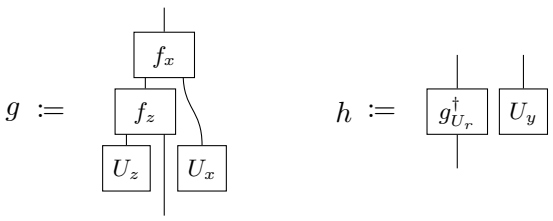


The equation on the right is obtained via some obvious re-wiring and via the by now familiar application of Definition 3.1 (i) and Proposition 4.3. This 'do' channel $do\colon X \to Y$ can now be computed, as $do(1) = \frac{1}{4}|1\rangle + \frac{3}{4}|0\rangle$ and $do(0) = \frac{2}{5}|1\rangle + \frac{3}{5}|0\rangle$. This outcome is as in [3]. It shows that the treatment (input 1 for do) does not really have effect of survival (output 1).

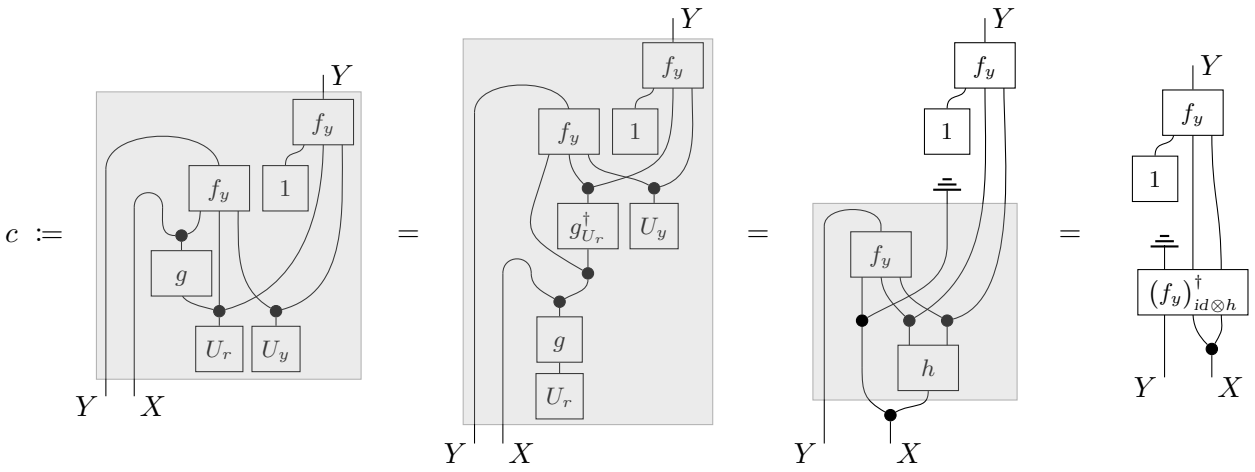
Next, we look at the counterfactual question. There is a patient who did not get treatment and died. Thus we can ask, had the patient been treated, would they have survived? We form a twinned diagram, coupling the factual and the counterfactual situation via the exogenous distributions:



The factual side on the left describes the patient's situation with no treatment ($X = 0$) and no survival ($Y = 0$). On the counterfactual side on the right, treatment is enforced, via the input 1 to f_y and we like to learn about (export) the survival probability via the Y-wire. In order to increase readability we will use the following abbreviations.



The disintegration of the combination of the above two dashed diagrams, for factual and counterfactual, happens in



The abbreviation g helps to simplify the situation in the first shaded box. The dagger of this g is introduced in the second shaded box. In the next step, box removal is applied, formally after using Lemma 3.3. At the same time, the abbreviation for h is introduced and the copy is pulled out at the bottom, using Definition 3.1 (i). Now we have a situation where a dagger with respect to a channel can be introduced, as in (4) on the right, namely the dagger of f_y with respect to the channel $id \otimes h$.

Concretely, the counterfactual channel $c\colon X \times Y \to \mathcal{D}(Y)$, with $X = Y = \mathbf{2}$, is:

$$c(1,1) = 1\big|1\big\rangle \qquad c(0,1) = \tfrac{49}{454}\big|1\big\rangle + \tfrac{405}{454}\big|0\big\rangle \qquad c(1,0) = 1\big|0\big\rangle \qquad c(0,0) = \tfrac{1}{46}\big|1\big\rangle + \tfrac{45}{46}\big|0\big\rangle.$$

In [3] only the latter outcome $\frac{1}{46} \approx 0.0217$ occurs. Our disintegration approach yields the entire channel, with all possible inputs. The low output at $(0,0)$ of about 2% survival chance, had the treatment been given, confirms the earlier finding that the treatment has little positive effect. We conclude with an interpretation of the different outcomes of the channel c. The first two columns below describe the factual situation.

factual treatment	**factual survival**	**survival chance had treatment been given**
1	1	1
1	0	0
0	1	$\frac{49}{454}$
0	0	$\frac{1}{46}$

The third line is intriguing. It describes the situation where the patient survived without treatment, and where the survival probability had the treatment been given is only $\frac{49}{454} \approx 11\%$. Apparently, in this situation, the treatment has a negative effect, lowering the chance of survival.

10 Concluding remarks

This paper argues that inference in Bayesian networks, via conditioning / disintegration, can be done in a compositional manner. This is achieved by adding Proposition 4.3 to the calculus introduced in [29], allowing one to directly compute with open models. This extends the range of applications of graphical techniques for probabilistic reasoning. It may become an ingredient of explainable AI, following the lines of [41]. All our applications involve discrete probability, but the approach is sufficiently abstract to allow generalisation to suitable categories for continuous probability. This is not trivial, for instance because the cancellativity of caps that we use in Definition 3.2 does not hold for Borel spaces, so a different approach is needed. A possible approach would be to directly develop a theory of least disintegrations without

relying on least normalisations and cancellative comparators. Partial Markov categories may provide a good setting for this, following [25,26].

One further line of work is to develop automation for the rewriting of string diagrams that happens in this paper. Another line of work is to integrate the theory of partial Markov categories with (partial) effectuses [7,5,17], where there is more logical structure.

References

[1] Balke, A. and J. Pearl, *Probabilistic evaluation of counterfactual queries*, in: B. Hayes-Roth and R. Korf, editors, *Proc. 12th Nat. Conf. on Artificial Intelligence*, pages 230–237, AAAI Press / The MIT Press (1994).
https://doi.org/10.1145/3501714.3501733

[2] Bao, J., S. Docherty, J. Hsu and A. Silva, *A bunched logic for conditional independence*, in: *Logic in Computer Science*, IEEE, Computer Science Press (2021).
https://doi.org/10.1109/LICS52264.2021.9470712

[3] Bareinboim, E., J. Correa, D. Ibeling and T. Icard, *On Pearl's Hierarchy and the Foundations of Causal Inference*, pages 507–556, Assoc. for Computing Machinery (2022).
https://doi.org/10.1145/3501714.3501743

[4] Bingham, E., J. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall and N. Goodman, *Pyro: deep universal probabilistic programming* **20(1)**, pages 973–978 (2019).
https://doi.org/10.5555/3322706.3322734

[5] Cho, K., *Total and partial computation in categorical quantum foundations*, in: C. Heunen, P. Selinger and J. Vicary, editors, *Quantum Physics and Logic (QPL) 2015*, Elect. Proc. in Theor. Comp. Sci., pages 116–135.
https://doi.org/10.4204/EPTCS.195.9

[6] Cho, K. and B. Jacobs, *Disintegration and Bayesian inversion via string diagrams*, Math. Struct. in Comp. Sci. **29(7)**, pages 938–971 (2019).
https://doi.org/10.1017/s0960129518000488

[7] Cho, K., B. Jacobs, A. Westerbaan and B. Westerbaan, *An introduction to effectus theory* (2015). See https://arxiv.org/abs/1512.05813.

[8] Clerc, F., F. Dahlqvist, V. Danos and I. Garnier, *Pointless learning*, in: J. Esparza and A. Murawski, editors, *Foundations of Software Science and Computation Structures*, number 10203 in Lect. Notes Comp. Sci., pages 355–369, Springer, Berlin (2017).
https://doi.org/10.1007/978-3-662-54458-7_21

[9] Culbertson, J. and K. Sturtz, *A categorical foundation for Bayesian probability*, Appl. Categorical Struct. **22(4)**, pages 647–662 (2014).
https://doi.org/10.1007/s10485-013-9324-9

[10] Cusumano-Towner, M., F. Saad, A. Lew and V. Mansinghka, *Gen: a general-purpose probabilistic programming system with programmable inference*, in: *Proc. of the 2002 ACM SIGPLAN Conf. on Programming Language Design and Implementation (PLDI)*, pages 221–236, ACM (2019).
https://doi.org/10.1145/3314221.3314642

[11] Fong, B., *Causal Theories: A Categorical Perspective on Bayesian Networks*, Master's thesis, Univ. of Oxford (2012).
https://doi.org/10.48550/arXiv.1301.6201

[12] Fritz, T., *A synthetic approach to Markov kernels, conditional independence, and theorems on sufficient statistics*, Advances in Math. **370**, page 107239 (2020).
https://doi.org/10.1016/J.AIM.2020.107239

[13] Fritz, T., T. Gonda, A. Lorenzin, P. Perrone and D. Stein, *Absolute continuity, supports and idempotent splitting in categorical probability* (2023).
https://doi.org/10.48550/arXiv.2308.00651

[14] Fritz, T. and A. Klingler, *The d-separation criterion in categorical probability*, J. Mach. Learn. Res. **24** (2023), ISSN 1532-4435.
https://jmlr.org/papers/v24/22-0916.html

[15] Fritz, T., A. Klingler, D. McNeely, A. Shah-Mohammed and Y. Wang, *Hidden Markov models and the Bayes filter in categorical probability* (2024).
https://doi.org/10.48550/arXiv.2401.14669

[16] Goodman, N. and A. Stuhlmüller, *The design and implementation of probabilistic programming languages* (2014). https://dippl.org.

[17] Jacobs, B., *New directions in categorical logic, for classical, probabilistic and quantum logic*, Logical Methods in Comp. Sci. **11(3)** (2015).
https://doi.org/10.2168/lmcs-11(3:24)2015

[18] Jacobs, B., *The mathematics of changing one's mind, via Jeffrey's or via Pearl's update rule*, Journ. of Artif. Intelligence Research **65**, pages 783–806 (2019).
https://doi.org/10.1613/jair.1.11349

[19] Jacobs, B., *Learning from what's right and learning from what's wrong*, number 351 in Elect. Proc. in Theor. Comp. Sci., pages 116–133 (2021).
https://doi.org/10.4204/EPTCS.351.8

[20] Jacobs, B., *Structured probabilitistic reasoning* (2025). Book, in preparation, see http://www.cs.ru.nl/B.Jacobs/PAPERS/ProbabilisticReasoning.pdf.

[21] Jacobs, B., A. Kissinger and F. Zanasi, *Causal inference via string diagram surgery: A diagrammatic approach to interventions and counterfactuals*, Math. Struct. in Comp. Sci. **31(5)**, pages 553–574 (2021).
https://doi.org/10.1017/S096012952100027X

[22] Jacobs, B. and F. Zanasi, *The logical essentials of Bayesian reasoning*, in: G. Barthe, J.-P. Katoen and A. Silva, editors, *Foundations of Probabilistic Programming*, pages 295–331, Cambridge Univ. Press (2021).
https://doi.org/10.1017/9781108770750.010

[23] Jimenez, L., T. Heskes and M. Stoelinga, *Fault trees, decision trees, and binary decision diagrams: A systematic comparison*, in: B. Castanier, M. Cepin, D. Bigaud and C. Berenguer, editors, *Proc. 31st Europ. Safety and Reliability Conference*, pages 673–680 (2021).
https://doi.org/10.3850/978-981-18-2016-8_241-cd

[24] Lavore, E. D., B. Jacobs and M. Roman, *A simple formal language for probabilistic decision problems* (2024).
https://doi.org/10.48550/arXiv.2410.10643

[25] Lavore, E. D. and M. Roman, *Evidential decision theory via partial Markov categories*, in: *Logic in Computer Science*, IEEE, Computer Science Press (2023).
https://doi.org/10.1109/LICS56636.2023.10175776

[26] Lavore, E. D., M. Román, P. Sobociński and M. Széles, *Order in partial Markov categories*, in: *Mathematical Foundations of Programming Semantics* (2025).
https://doi.org/10.48550/ARXIV.2507.19424

[27] Li, J., A. Ahmed and S. Holtzen, *Lilac: A modal separation logic for conditional probability*, Programming Language Design and Implementation **7**, pages 148–171 (2023).
https://doi.org/10.1145/3591226

[28] Lopuhaä-Zwakenberg, M., *Fault tree reliability analysis via squarefree polynomials.*, in: F. Mayo, L. Pires and E. Seidewitz, editors, *Proc. 12th Int. Conf. on Model-Based Software and Systems Engineering*, pages 39–49, Scitepress (2024).
https://doi.org/10.5220/0012334000003645

[29] Lorenz, R. and S. Tull, *Causal models in string diagrams* (2023).
https://doi.org/10.48550/arXiv.2304.07638

[30] Meent, J.-W. v. d., , B. Paige, H. Yang and F. Wood, *An introduction to probabilistic programming* (2021).
https://doi.org/10.48550/arXiv.1809.10756

[31] Nielsen, M., *If correlation doesn't imply causation, then what does?* (2012). Online report, published Jan. 23, 2012.
http://www.michaelnielsen.org/ddi/if-correlation-doesnt-imply-causation-then-what-does/

[32] Pearl, J., *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Graduate Texts in Mathematics 118, Morgan Kaufmann (1988).
https://doi.org/10.1016/C2009-0-27609-4

[33] Pearl, J., *Causality. Models, Reasoning, and Inference*, Cambridge Univ. Press, 2nd edition (2009).
https://doi.org/10.1017/CBO9780511803161

[34] Piedeleu, R., M. Torres-Ruiz, A. Silva and F. Zanasi, *A complete axiomatisation of equivalence for discrete probabilistic programming* (2024).
https://doi.org/10.48550/arXiv.2408.14701

[35] Piedeleu, R. and F. Zanasi, editors, *An Introduction to String Diagrams for Computer Scientists*, Elements in Applied Category Theory, Cambridge Univ. Press (2025).
https://doi.org/10.1017/9781009625715

[36] Schelling, T., *The Strategy of Conflict*, Harvard Univ. Press (1980), ISBN: 9780674840317

[37] Ścibior, A., Z. Ghahramani and A. Gordon, *Practical probabilistic programming with monads*, in: *Proc. 2015 ACM SIGPLAN Symp. on Haskell*, pages 165–176, ACM (2015).
https://doi.org/10.1145/2804302.2804317

[38] Spiegelhalter, D., A. Dawid, S. Lauritzen and R. Cowell, *Bayesian analysis in expert systems*, Statistical Science **8(3)**, pages 219–247 (1993).
https://doi.org/10.1214/ss/1177010888

[39] Stein, D., *Structural Foundations for Probabilistic Programming Languages*, Ph.D. thesis, Cambridge Univ. (2022). Available via https://dario-stein.de/notes/thesis.pdf.

[40] Stuhlmüller, A. and N. Goodman, *Reasoning about reasoning by nested conditioning: Modeling theory of mind with probabilistic programs*, Cognitive Systems Research **28**, pages 80–99 (2014).
https://doi.org/10.1016/j.cogsys.2013.07.003

[41] Tull, S., R. Lorenz, S. Clark, I. Khan and B. Coecke, *Towards compositional interpretability for XAI* (2024).
https://doi.org/10.48550/arXiv.2406.17583

[42] Yin, Y. and J. Zhang, *Markov categories, causal theories, and the do-calculus* (2022).
https://arxiv.org/abs/2204.04821

[43] Zhang, Y. and N. Amin, *Reasoning about 'reasoning about reasoning': Semantics and contextual equivalence for probabilistic programs with nested queries and recursion*, in: *Princ. of Programming Languages*, ACM Press (2022).
https://doi.org/10.1145/3498677

A Channel definitions in the child example

For completeness, we include the details of the channels / boxes (7) in the child Bayesian network example in Section 5. These details are used to calculate the actual disintegration (9).

$$LP \xrightarrow{co} \mathcal{D}(CO) \qquad DF \times CM \xrightarrow{hd} \mathcal{D}(HD) \qquad CM \times LP \xrightarrow{hi} \mathcal{D}(HI) \qquad HD \times HI \xrightarrow{lb} \mathcal{D}(LB).$$

First, the channel $d\colon BA \to \mathcal{D}(DI)$ is given by the following two equations.

$$\begin{aligned} d(ba) &= 0.2\big|\,pfc\,\big\rangle + 0.3\big|\,tga\,\big\rangle + 0.25\big|\,flt\,\big\rangle + 0.15\big|\,pis\,\big\rangle + 0.05\big|\,tpd\,\big\rangle + 0.05\big|\,lng\,\big\rangle \\ d(ba^{\perp}) &= 0.03\big|\,pfc\,\big\rangle + 0.34\big|\,tga\,\big\rangle + 0.3\big|\,flt\,\big\rangle + 0.23\big|\,pis\,\big\rangle + 0.05\big|\,tpd\,\big\rangle + 0.05\big|\,lng\,\big\rangle \end{aligned}$$

Next, we have three channels with DI as domain. First, $df\colon DI \to \mathcal{D}(DF)$ is determined by:

$$\begin{aligned} df(pfc) &= 0.15\big|\,lt\,\big\rangle + 0.05\big|\,no\,\big\rangle + 0.8\big|\,rt\,\big\rangle \\ df(tga) &= 0.1\big|\,lt\,\big\rangle + 0.8\big|\,no\,\big\rangle + 0.1\big|\,rt\,\big\rangle \\ df(flt) &= 0.8\big|\,lt\,\big\rangle + 0.2\big|\,no\,\big\rangle \\ df(pis) &= 1\big|\,lt\,\big\rangle \\ df(tpd) &= 0.33\big|\,lt\,\big\rangle + 0.33\big|\,no\,\big\rangle + 0.34\big|\,rt\,\big\rangle \\ df(lng) &= 0.2\big|\,lt\,\big\rangle + 0.4\big|\,no\,\big\rangle + 0.4\big|\,rt\,\big\rangle. \end{aligned}$$

Next there is $cm\colon DI \to \mathcal{D}\big(CM\big)$ of the form:

$$\begin{array}{rcl}
cm(pfc) &=& 0.4\big|\,no\,\big\rangle + 0.43\big|\,mi\,\big\rangle + 0.15\big|\,co\,\big\rangle + 0.02\big|\,tr\,\big\rangle \\
cm(tga) &=& 0.02\big|\,no\,\big\rangle + 0.09\big|\,mi\,\big\rangle + 0.09\big|\,co\,\big\rangle + 0.8\big|\,tr\,\big\rangle \\
cm(flt) &=& 0.02\big|\,no\,\big\rangle + 0.16\big|\,mi\,\big\rangle + 0.8\big|\,co\,\big\rangle + 0.02\big|\,tr\,\big\rangle \\
cm(pis) &=& 0.01\big|\,no\,\big\rangle + 0.02\big|\,mi\,\big\rangle + 0.95\big|\,co\,\big\rangle + 0.02\big|\,tr\,\big\rangle \\
cm(tpd) &=& 0.01\big|\,no\,\big\rangle + 0.03\big|\,mi\,\big\rangle + 0.95\big|\,co\,\big\rangle + 0.01\big|\,tr\,\big\rangle \\
cm(lng) &=& 0.4\big|\,no\,\big\rangle + 0.53\big|\,mi\,\big\rangle + 0.05\big|\,co\,\big\rangle + 0.02\big|\,tr\,\big\rangle.
\end{array}$$

And we also have $lp\colon DI \to \mathcal{D}\big(LP\big)$.

$$\begin{array}{rcl}
lp(pfc) &=& 0.6\big|\,nr\,\big\rangle + 0.1\big|\,cg\,\big\rangle + 0.3\big|\,ab\,\big\rangle \\
lp(tga) &=& 0.8\big|\,nr\,\big\rangle + 0.05\big|\,cg\,\big\rangle + 0.15\big|\,ab\,\big\rangle \\
lp(flt) &=& 0.8\big|\,nr\,\big\rangle + 0.05\big|\,cg\,\big\rangle + 0.15\big|\,ab\,\big\rangle \\
lp(pis) &=& 0.8\big|\,nr\,\big\rangle + 0.05\big|\,cg\,\big\rangle + 0.15\big|\,ab\,\big\rangle \\
lp(tpd) &=& 0.1\big|\,nr\,\big\rangle + 0.6\big|\,cg\,\big\rangle + 0.3\big|\,ab\,\big\rangle \\
lp(lng) &=& 0.03\big|\,nr\,\big\rangle + 0.25\big|\,cg\,\big\rangle + 0.72\big|\,ab\,\big\rangle
\end{array}$$

The next channel is $co\colon LP \to \mathcal{D}\big(CO\big)$ consisting of three distributions:

$$\begin{array}{rcl}
co(nr) &=& 0.8\big|\,nr\,\big\rangle + 0.1\big|\,lo\,\big\rangle + 0.1\big|\,hi\,\big\rangle \\
co(cg) &=& 0.65\big|\,nr\,\big\rangle + 0.05\big|\,lo\,\big\rangle + 0.3\big|\,hi\,\big\rangle \\
co(ab) &=& 0.45\big|\,nr\,\big\rangle + 0.05\big|\,lo\,\big\rangle + 0.5\big|\,hi\,\big\rangle
\end{array}$$

The firnal three channels with a product set as domain. First, $hd\colon DF \times CM \to \mathcal{D}\big(HD\big)$ is given by:

$$\begin{array}{rcl}
hd(lt, no) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(lt, mi) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(lt, co) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(lt, tr) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(no, no) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(no, mi) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(no, co) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(no, tr) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(rt, no) &=& 0.05\big|\,eq\,\big\rangle + 0.95\big|\,eq^{\perp}\,\big\rangle \\
hd(rt, mi) &=& 0.5\big|\,eq\,\big\rangle + 0.5\big|\,eq^{\perp}\,\big\rangle \\
hd(rt, co) &=& 0.95\big|\,eq\,\big\rangle + 0.05\big|\,eq^{\perp}\,\big\rangle \\
hd(rt, tr) &=& 0.5\big|\,eq\,\big\rangle + 0.5\big|\,eq^{\perp}\,\big\rangle
\end{array}$$

The next one is $hi\colon CM \times LP \to \mathcal{D}(HI)$.

$$\begin{aligned}
hi(no, nr) &= 0.93\big|\, mi\,\big\rangle + 0.05\big|\, mo\,\big\rangle + 0.02\big|\, se\,\big\rangle \\
hi(no, cg) &= 0.15\big|\, mi\,\big\rangle + 0.8\big|\, mo\,\big\rangle + 0.05\big|\, se\,\big\rangle \\
hi(no, ab) &= 0.7\big|\, mi\,\big\rangle + 0.2\big|\, mo\,\big\rangle + 0.1\big|\, se\,\big\rangle \\
hi(mi, nr) &= 0.1\big|\, mi\,\big\rangle + 0.8\big|\, mo\,\big\rangle + 0.1\big|\, se\,\big\rangle \\
hi(mi, cg) &= 0.1\big|\, mi\,\big\rangle + 0.75\big|\, mo\,\big\rangle + 0.15\big|\, se\,\big\rangle \\
hi(mi, ab) &= 0.1\big|\, mi\,\big\rangle + 0.65\big|\, mo\,\big\rangle + 0.25\big|\, se\,\big\rangle \\
hi(co, nr) &= 0.1\big|\, mi\,\big\rangle + 0.7\big|\, mo\,\big\rangle + 0.2\big|\, se\,\big\rangle \\
hi(co, cg) &= 0.05\big|\, mi\,\big\rangle + 0.65\big|\, mo\,\big\rangle + 0.3\big|\, se\,\big\rangle \\
hi(co, ab) &= 0.1\big|\, mi\,\big\rangle + 0.5\big|\, mo\,\big\rangle + 0.4\big|\, se\,\big\rangle \\
hi(tr, nr) &= 0.02\big|\, mi\,\big\rangle + 0.18\big|\, mo\,\big\rangle + 0.8\big|\, se\,\big\rangle \\
hi(tr, cg) &= 0.1\big|\, mi\,\big\rangle + 0.3\big|\, mo\,\big\rangle + 0.6\big|\, se\,\big\rangle \\
hi(tr, ab) &= 0.02\big|\, mi\,\big\rangle + 0.18\big|\, mo\,\big\rangle + 0.8\big|\, se\,\big\rangle.
\end{aligned}$$

As last channel we have $lb\colon HD \times HI \to \mathcal{D}(LB)$.

$$\begin{aligned}
lb(eq, mi) &= 0.1\big|\, {<}5\,\big\rangle + 0.3\big|\, 5\text{-}12\,\big\rangle + 0.6\big|\, 12{+}\,\big\rangle \\
lb(eq, mo) &= 0.3\big|\, {<}5\,\big\rangle + 0.6\big|\, 5\text{-}12\,\big\rangle + 0.1\big|\, 12{+}\,\big\rangle \\
lb(eq, se) &= 0.5\big|\, {<}5\,\big\rangle + 0.4\big|\, 5\text{-}12\,\big\rangle + 0.1\big|\, 12{+}\,\big\rangle \\
lb(eq^{\perp}, mi) &= 0.4\big|\, {<}5\,\big\rangle + 0.5\big|\, 5\text{-}12\,\big\rangle + 0.1\big|\, 12{+}\,\big\rangle \\
lb(eq^{\perp}, mo) &= 0.5\big|\, {<}5\,\big\rangle + 0.45\big|\, 5\text{-}12\,\big\rangle + 0.05\big|\, 12{+}\,\big\rangle \\
lb(eq^{\perp}, se) &= 0.6\big|\, {<}5\,\big\rangle + 0.35\big|\, 5\text{-}12\,\big\rangle + 0.05\big|\, 12{+}\,\big\rangle.
\end{aligned}$$