Leopold-Franzens Universität Innsbruck

Master Thesis

The

Causal Compatibility Problem

in

Categorical Probability

Author:
Andreas Klingler

Supervisor:
Ass.-Prof. Dr. Tobias Fritz

A thesis submitted
at the
Department of Mathematics

universität
innsbruck

August 18, 2022

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt durch meine eigenhändige Unterschrift, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe. Alle Stellen, die wörtlich oder inhaltlich den angegebenen Quellen entnommen wurden, sind als solche kenntlich gemacht. Die vorliegende Arbeit wurde bisher in gleicher oder ähnlicher Form noch nicht als Magister-/Master-/Diplomarbeit/Dissertation eingereicht.

Teile dieser Arbeit wurden veröffentlicht in [FK22]:

> *The d-separation criterion in Categorical Probability.*
> T. Fritz, A. Klingler.
> arXiv:2207.05740 (2022)

18. August 2022
Datum

Unterschrift

LEOPOLD-FRANZENS UNIVERSITÄT INNSBRUCK

Abstract

Fakultät für Mathematik, Informatik und Physik

Department of Mathematics

Master of Science

The Causal Compatibility Problem in Categorical Probability

by Andreas KLINGLER

Bayesian networks are probabilistic graphical models representing random variables via conditional probability distributions. They are causal structures given by a directed acyclic graph. In this setting, the d-separation criterion allows for deciding compatibility of a joint probability distribution with a Bayesian network via analyzing conditional independence conditions when having access to all variables in the model. However, checking d-separation on directed acyclic graphs is not particularly intuitive; it includes several case distinctions on the particular shapes of paths between nodes in the graph. Moreover, Bayesian networks traditionally include only certain causal structures, not covering models with inputs or symmetries.

In this work, we study causal compatibility in the language of categorical probability using the recently introduced framework of Markov categories. In particular, we introduce a string diagrammatic framework of generalized causal models together with a categorical notion of d-separation, which is simpler than the traditional one. Further, we present a proof of the d-separation criterion in this abstract setting.

Our results apply to probabilities in measure theory (with standard Borel spaces), Gaussian random variables, and finite probability distributions. This suggests that representing causal models with string diagrams is a more natural approach than using directed acyclic graphs.

Acknowledgements

First, I am extremely grateful to my supervisor Ass.-Prof. Dr. Tobias Fritz for his invaluable guidance and support throughout this exciting and challenging project. He always had an open door when I was stuck and prevented me from falling into common traps throughout the project. In addition, he always kept me motivated and provided me with many new insights in the field of Applied Category Theory.

Special thanks to Ass.-Prof. Dr. Gemma De las Cuevas and Univ.-Prof. Dr. Tim Netzer for the scientific and financial support, giving me the opportunity to explore new fields of research beyond my Ph.D. research topic. I am also grateful to all the members of our research group for the countless inspiring discussions, which allowed me to learn many new and different perspectives on this research topic.

I would also like to extend my thanks to my family for supporting me and making it possible to complete both Master's studies, physics and mathematics.

Thanks should also go to all my friends and colleagues who accompanied me through my studies for all the inspiring discussions.

Contents

Introduction **1**

1 Categorical Probability **3**
1.1 Markov categories 4
1.2 Construction of Markov categories: Kleisli categories 7
1.3 Existence of conditional Markov kernels 15
1.4 Independence and Conditional Independence 15

2 A categorical framework for causal models **19**
2.1 From directed acyclic graphs to string diagrams 19
2.2 The category of hypergraphs 23
2.3 gs-monoidal string diagrams 25
2.4 Free Markov categories 29
2.5 Generalized causal models 30

3 Characterizing causal compatibility for generalized causal models **35**
3.1 Categorical d-separation in string diagrams 37
3.2 The d-separation criterion 44

4 Conclusion and Outlook **55**

Bibliography **57**

A Construction of the Giry monad **61**

Introduction

A central task among many modern sciences is studying cause-effect relationships between observed quantities. In the last decades, graphical models like Bayesian networks have turned out to be a successful approach to such phenomena. Bayesian networks decompose a probability distribution into conditional probabilities according to a given causal structure represented by a directed acyclic graph.

Besides being a well-defined semantics for causal structures, Bayesian networks also allowed for developing tools to analyze causal relationships in observational data via the d-separation criterion when having access to all variables in the model. This criterion states that a probability distribution is compatible with a given causal structure if and only if every d-separation in the directed acyclic graph implies conditional independence in the probability distribution.

While conditional probabilities can be easily computed for discrete random variables using Bayesian calculus, this does not naively extend to continuous or mixed random variables. This ambiguity stems from the classical approach of probability theory based on Kolmogorov's measure-theoretic axioms, which constitute a low-level approach to probability theory. Operational concepts, like multipartite probability distributions, conditioning, or Bayesian inversion, are then built up from these axioms.

Markov categories [Fri20] constitute a different approach to probability theory. In contrast to measure theory, Markov categories are a top-down approach specifying how Markov kernels *compose* and *decompose* using a set of basic algebraic axioms. This gives a new view on probability theory which is completely agnostic about the usual measure theory behind it.

Several well-known results from classical measure-theoretic probability theory have been proven in the language of Markov categories, including de Finetti's theorem [FGP21], Kolmogorov's zero-one law [FR19], or the Blackwell-Sherman-Stein Theorem [Fri+20]. All of these proofs offer new insights into these results and provide a clean, unified proof, and include discrete and measure-theoretic probabilities as particular instances.

Beyond being a different approach to probability theory, category theory also offers string diagrams as a structural framework to represent causal models. In the setting of Markov categories, a string diagram reflects the information flow of variables present in a causal model. Every wire represents a variable, and every box is a mechanism that samples a new variable depending on its input information.

While boxes in a string diagram are usually understood as concrete stochastic mechanisms (for example finite stochastic maps or even measurable Markov kernels), these string diagrams can also be understood as a blueprint of a causal structure.

In this work, we introduce new types of graphical models in the language of string diagrams in Markov categories. In particular, we represent causal models as string diagrams instead of directed acyclic graphs as done in the classical setting of Bayesian networks.

This approach has several advantages. First, it is more general than the classical approach of Bayesian networks. Second, it simplifies known methods like the d-separation criterion. While directed acyclic graphs embed into the string diagrammatic framework, string diagrams can represent causal models with inputs and models with additional symmetries. Although more general, we also show that string diagrams lead to a simplified analysis of causal structures compared to classical Bayesian networks. We present a novel notion of categorical d-separation, specializing to the classical d-separation when considering Bayesian networks on directed acyclic graphs. This definition is much simpler than the original one since it corresponds to topological disconnectedness in string diagrams.

We prove that d-separation also decides causal compatibility in the language of string diagrams. Since we prove this result in the framework of Markov categories, this proof also includes continuous and mixed probability distributions which are usually treated in the measure-theoretic framework. Moreover, the proof in the setting of Markov categories highlights the necessary resources for the d-separation criterion to be applicable.

Our study suggests that string diagrams are a more natural approach to causal models than the traditional framework using directed acyclic graphs.

This work is structured as follows: In Chapter 1 we review the framework of Markov categories and highlight how the measure-theoretic approach to probability embeds into the framework. In Chapter 2 we introduce the string diagrammatic formalism representing causal structures. Finally, we study in Chapter 3 the causal compatibility problem in the language of Markov categories and string diagrams. In particular, we introduce the notion of d-separation on string diagrams and prove the d-separation criterion.

Chapter 1

Categorical Probability

Probability theory is traditionally based on Kolmogorov's measure theory axioms [KB18]. Despite its technicality, it opened the door for a rich theory leading to theoretical results and applications in many areas, including stochastic processes, theory of convergence and large deviations, stochastic partial differential equations, and machine learning.

Categorical probability is a complementary approach to the foundations of probability theory. Instead of building up probability by a low-level approach, the main goal of categorical probability is to formalize the theory via some basic algebraic rules satisfied by systems of Markov kernels.

In a recent paper, Fritz [Fri20] introduced the concept of Markov categories as a concrete approach to formalizing the behavior of Markov kernels building up on Cho and Jacobs [CJ19]. In this chapter, we review this approach, introducing Markov categories. Further, we will show that certain concrete classes of probability distributions like discrete probabilities, Gaussian probabilities, or arbitrary measure-theoretic probabilities fit into this framework by constructing concrete realizations thereof. Moreover, we will introduce essential notions of probability theory such as Bayesian disintegration or (conditional) independence in the language of Markov categories. This sheds light on these notions from an intuitive level and allows us to study probability theory completely from an abstract level.

We assume that the reader is familiar with the standard terminology of category theory, particularly *monoidal categories* and their string diagrammatic calculus. For a detailed treatment, we refer to Mac Lane's classic book [Mac78], Leinster's book [Lei14], as well as Perrone's wonderful lecture notes [Per19]. For a detailed exposition of monoidal categories and their applications, we refer to Fong's and Spivak's book [FS18b].

This chapter is structured as follows. In Section 1.1, we introduce the notion of Markov and gs-monoidal categories, the basic building block of categorical probability. In Section 1.2, we present an explicit construction of such categories. By applying this construction, we give a categorical description of different classes of probability spaces, including measurable Markov kernels on arbitrary measurable spaces, standard Borel measures on measurable spaces, and Gaussian probability theory. In Section 1.3, we review the axiom of conditioning. Finally, we introduce a categorical description of (conditional) independence in Section 1.4. Both latter concepts are essential for the compatibility criteria in Chapter 3.

Parts of Section 1.4 are based on [FK22, Section 5.2].

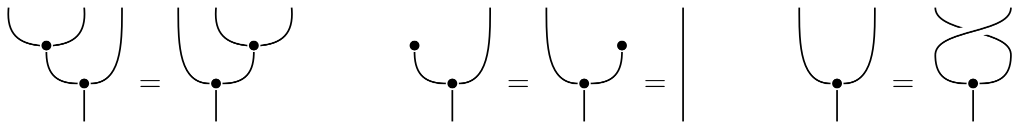
1.1 Markov categories

In the following, we define the notion of a Markov category and study concrete examples thereof. Intuitively, a Markov category is an abstract axiomatization of probability distributions together with modeling also the flow of information.

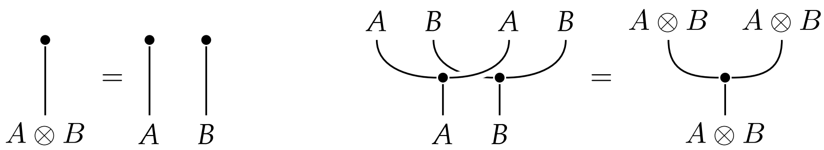
Definition 1.1 (gs-monoidal and Markov Category)**.**

(*i*) *A* gs-monoidal category $\mathbf{C}$ *is a symmetric monoidal category equipped with a comonoid structure for every object* $X \in \mathbf{C}$ *given by a counit* $\mathsf{del}_X : X \to I$ *and a comultiplication* $\mathsf{copy}_X : X \to X \otimes X$*. In the string diagrammatic notation these operations are depicted as*

and satisfy the commutative comonoid equations, diagrammatically given by


(1.1)

and is as well compatible with the comonoid structure, i.e.

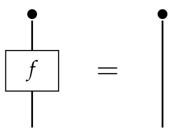


(1.2)

as well as

(1.3)

(*ii*) *A gs-monoidal category is called* Markov category *if* del *is in addition natural, i.e.*



(1.4)

Note that the "gs" in the definition refers to the intended interpretation of the comonoid structure since it stands for *garbage* (i.e. deletion) and *sharing* (i.e. copying). The additional condition for Markov categories in Equation (1.4) axiomatizes the possibility of marginalizing random variables in the context of probability theory. We refer to Example 1.3 and Section 1.2 for examples of Markov categories.

Intuitively, the objects of a Markov category can be understood as abstract sample spaces. Morphisms $f : X \to Y$ are understood as stochastic maps taking inputs $x \in X$ and sampling in Y depending on x. Markov categories additionally include the two basic operations applicable to every sampling space: copying and discarding information. The compatibility equations reflect the behavior of these morphisms in the operational setting. In particular, the naturality of del imposes the condition that the particular sampling process becomes irrelevant whenever the information becomes discarded immediately afterwards.

The introduced notions have already been studied in different contexts and names. To the best of our knowledge, gs-monoidal categories first appeared in Gadducci's PhD thesis [Gad96] and subsequently a paper joint with Corradini [CG99]. The main motivation in their works was the graphical notations for formal languages. Later, Golubtsov [Gol99] independently introduced a similar definition, already having applications to statistics in mind. Another work on string-diagrammatic approaches to probability was done by Coecke and Spekkens [CS11].

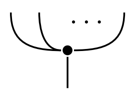
The first appearance of gs-monoidal categories in the context of probability theory is in Fong's Master thesis [Fon13]. However, no explicit definition and construction thereof is given. Cho and Jacobs [CJ19] introduce the notion of gs-monoidal categories in their present form, calling them CD (copy and delete) categories. In the same work, the authors also introduce Markov categories dubbing them *affine CD categories*.

The term Markov category is due to a paper by Fritz [Fri20], using the interpretation of morphisms as generalized Markov kernels. In the following, we will give a motivation by studying some concrete examples of Markov categories in Section 1.2.

Remark 1.2. Let us point mention some distinctive features of the introduced categories.

(i) Note that if Equation (1.4) is true, then the conditions in Equation (1.3) are automatically satisfied.

(ii) By the associativity of the comonoid structure there is an unambiguous way of a multipartite copy-morphism arising from multiple concatenations of copy_X. In the following, we will write



as a shorthand for this morphism.

(iii) $\mathsf{del}_I = \mathrm{id}_I$ implies that I is a terminal object since for every morphism $f : X \to I$ we have

$$f = \mathrm{id}_I \circ f = \mathsf{del}_I \circ f = \mathsf{del}_X \tag{1.5}$$

where we have used Equation (1.4) in the last step.

(iv) Since $\mathsf{del}_I = \mathrm{id}_I$, the naturality of del is equivalent to I being a terminal object. This follows immediately since for every morphism $f : X \to I$ we have

$$f = \mathsf{del}_I \circ f = \mathsf{del}_X.$$

△

In the following, we will present a representative example of a Markov category: the category of finite Markov kernels on finite sets.

Example 1.3. The Markov category consisting of finite sets as objects and finite Markov kernels as morphisms is called **FinStoch**. More precisely, for finite sets X and Y, a morphism $f : X \to Y$ is given by a function[1]

$$f : X \times Y \to \mathbb{R} : (x, y) \mapsto f(y|x) \tag{1.6}$$

such that $f(y|x) \geq 0$ and $\sum_{y \in Y} f(y|x) = 1$. Therefore, for a fixed value $x \in X$, $y \mapsto f(y|x)$ is again a probability distribution. Composition of two morphisms $f : X \to Y$, $g : Y \to Z$ is given by the *Chapman-Kolmogorov equation*

$$(g \circ f)(z|x) \coloneqq \sum_{y \in Y} g(z|y) \cdot f(y|x). \tag{1.7}$$

The symmetric monoidal structure on **FinStoch** is given on the level of objects by the Cartesian product. On the level of morphisms, we have the tensor product of Markov kernels (equivalently the tensor product of stochastic matrices)

$$g \otimes f : A \times B \to X \times Y \qquad \text{with} \qquad (g \otimes f)(xy|ab) \coloneqq g(x|a) \cdot f(y|b) \tag{1.8}$$

for morphisms $g : A \to X$, $f : B \to Y$.

The structure isomorphisms are inherited from the symmetric monoidal category **FinSet** (equipped with the Cartesian product and a unit object I) by applying the inclusion functor

$$\mathcal{F} : \mathbf{FinSet} \to \mathbf{FinStoch} : \quad \mathcal{F}X = X \quad \text{and} \quad \mathcal{F}f : (x, y) \mapsto \begin{cases} 1 & : \text{ if } y = f(x) \\ 0 & : \text{ else} \end{cases} \tag{1.9}$$

Moreover, it is immediate to show that **FinSet** is also a Markov category, where $\mathsf{del}_X : X \to I$ is the constant map and $\mathsf{copy}_X : X \to X \times X : x \mapsto (x, x)$ the copy map. Therefore, the induced maps via $\mathcal{F}$

$$\mathsf{del}_X(|x) = 1 \tag{1.10}$$

and

$$\mathsf{copy}_X(x_1, x_2|x) = \begin{cases} 1 & : \text{ if } x_1 = x_2 = x \\ 0 & : \text{ else.} \end{cases} \tag{1.11}$$

are a comonoid structure on **FinStoch** which satisfy the comonoid equations defining a Markov category. △

In the following section, we will introduce more sophisticated examples of Markov kernels, acting for example on continuous probability spaces or arbitrary measurable spaces. These constructions leverage a general procedure of building up Markov categories by using *Kleisli categories*.

[1] An equivalent way is describing f as a stochastic matrix, i.e. a nonnegative matrix $(f_{xy})_{x \in X, y \in Y}$ such that $\sum_{y \in Y} f_{xy} = 1$. Composition of morphisms is then simply matrix multiplication.

1.2 Construction of Markov categories: Kleisli categories

In this section, we review the standard approach of constructing Markov categories using Kleisli categories. This is mainly a review of [FL22, Section 3]. For an introduction to Kleisli categories, we refer to Perrone's lecture notes [Per19, Section 5].

Definition 1.4. *A* monad (T, η, μ) *on the category* $\mathbf{C}$ *is given by*

(*i*) *a functor* $T : \mathbf{C} \to \mathbf{C}$,

(*ii*) *a natural transformation* $\eta : \mathrm{id}_C \Longrightarrow T$, *called* unit,

(*iii*) *a natural transformation* $\mu : TT \Longrightarrow T$, *called* multiplication *or* composition.

which make the following three diagrams commute

$$
\begin{array}{ccc}
T & \overset{\eta T}{\Longrightarrow} & TT \\
& {\scriptstyle \mathrm{id}}\searrow\!\!\searrow & \Downarrow{\scriptstyle \mu} \\
& & T
\end{array}
\qquad
\begin{array}{ccc}
T & \overset{T\eta}{\Longrightarrow} & TT \\
& {\scriptstyle \mathrm{id}}\searrow\!\!\searrow & \Downarrow{\scriptstyle \mu} \\
& & T
\end{array}
\qquad
\begin{array}{ccc}
TTT & \overset{T\mu}{\Longrightarrow} & TT \\
{\scriptstyle \mu T}\Downarrow & & \Downarrow{\scriptstyle \mu} \\
TT & \overset{\mu}{\Longrightarrow} & T
\end{array}
\tag{1.12}
$$

The left and the middle diagram are called left *and* right unitality, *the right is called* associativity.

In the following, we will introduce a notion of monads with additional structure, called *symmetric monoidal monads*. These are monads respecting the symmetric monoidal structure of a category.

Definition 1.5. *Let* $\mathbf{C}$ *be a symmetric monoidal category. A* symmetric monoidal monad *is a monad* $T : \mathbf{C} \to \mathbf{C}$ *with unit* η *and multiplication* μ *together with a morphism*

$$\nabla_{X,Y} : TX \otimes TY \to T(X \otimes Y) \tag{1.13}$$

natural in both X *and* Y, *which turns* T *into a lax monoidal functor.*

Further, the natural transformations η, μ *must be monoidal transformations, i.e. the diagrams*

$$
\begin{array}{ccccc}
TTX \otimes TTY & \xrightarrow{\nabla_{TX,TY}} & T(TX \otimes TY) & \xrightarrow{T\nabla_{X,Y}} & TT(X \otimes Y) \\
\Big\downarrow{\scriptstyle \mu_X \otimes \mu_Y} & & & & \Big\downarrow{\scriptstyle \mu_{X\otimes Y}} \\
TX \otimes TY & & \xrightarrow{\qquad\qquad \nabla_{X,Y} \qquad\qquad} & & T(X \otimes Y)
\end{array}
\tag{1.14}
$$

and

$$
\begin{array}{ccc}
& X \otimes Y & \\
{\scriptstyle \eta_X \otimes \eta_Y}\swarrow & & \searrow{\scriptstyle \eta_{X\otimes Y}} \\
TX \otimes TY & \xrightarrow{\quad \nabla_{X,Y} \quad} & T(X \otimes Y)
\end{array}
\tag{1.15}
$$

commute for all objects X, Y *in* $\mathbf{C}$.

Note that the term *lax monoidal functor* refers to the existence of a natural transformation ∇ relating the objects $TX \otimes TY$ and $T(X \otimes Y)$. The term *monoidal transformation* then refers to the equivalent behavior of the components $\eta_{X\otimes Y}$ and $\eta_X \otimes \eta_Y$ as well as $\mu_{X\otimes Y}$ and $\mu_X \otimes \mu_Y$ via the structure map ∇.

In the following, we introduce the notion of a Kleisli category based on a monad (T, η, μ) on a category $\mathbf{C}$. While the objects are the same as in $\mathbf{C}$, a morphism from X to Y is given by a morphism $k : X \to TY$ in $\mathbf{C}$. These morphisms are called *Kleisli morphisms*. Kleisli composition $(h \circ k) : X \to TZ$ between Kleisli morphisms $k : X \to TY$ and $h : Y \to TZ$ arises from the monad structure as

$$X \xrightarrow{k} TY \xrightarrow{Th} TTZ \xrightarrow{\mu} TZ. \tag{1.16}$$

Definition 1.6 (Kleisli category)**.** *Let* (T, η, μ) *be a monad on* $\mathbf{C}$*. Then, the* Kleisli category of $\mathbf{C}$, *denoted* $\mathbf{Kl}(T)$*, is defined in the following way:*

(*i*) *The objects are the objects of* $\mathbf{C}$*.*

(*ii*) *The morphisms are the Kleisli morphisms of* T*.*

(*iii*) *The identity maps are given by the units* $\eta_X : X \to TX$ *for each object* X*.*

(*iv*) *The composition of morphisms is given by the Kleisli composition.*

$\mathbf{Kl}(T)$ is indeed a category, see for example [Per19, Definition 5.1.16].

For further use, we define the inclusion functor (also called *Kleisli adjunction*) $\mathcal{R}_T : \mathbf{C} \to \mathbf{Kl}(T)$ by $\mathcal{R}_T X = X$ for every object X and $\mathcal{R}_T f = \eta_Y \circ f$ for every morphism $f : X \to Y$. Given a symmetric monoidal monad T, it is possible to extend $\mathbf{Kl}(T)$ to a symmetric monoidal category. A proof of the following statement can be found for example in [Sea12, Proposition 1.2.2].

Proposition 1.7. *Let* $\mathbf{C}$ *be a symmetric monoidal category and* (T, η, μ, ∇) *be a symmetric monoidal monad on* $\mathbf{C}$*. Then the Kleisli category* $\mathbf{Kl}(T)$ *becomes symmetric monoidal by setting*

(*i*) *the tensor product of objects being the tensor product of* $\mathbf{C}$*.*

(*ii*) *the tensor product of morphisms* $f : X \to TA$ *and* $g : Y \to TB$ *given by*

$$X \otimes Y \xrightarrow{f\otimes g} TA \otimes TB \xrightarrow{\nabla_{A,B}} T(A \otimes B) \tag{1.17}$$

The structure morphisms are given via applying the functor $\mathcal{R}_T$ *to the structure morphisms on* $\mathbf{C}$*. Moreover, the inclusion functor* $\mathcal{R}_T : \mathbf{C} \to \mathbf{Kl}(T)$ *is strict symmetric monoidal.*

The last statement implies that if there is a distinguished comonoid structure on every object of $\mathbf{C}$, there is also a comonoid structure on $\mathbf{Kl}(T)$. In particular, the comonoids copy_X and del_X are given by

$$X \xrightarrow{\mathsf{copy}_X} X \otimes X \xrightarrow{\eta_{X\otimes X}} T(X \otimes X) \tag{1.18}$$

and

$$X \xrightarrow{\mathsf{del}_X} I \xrightarrow{\eta_X} T(I) \tag{1.19}$$

By functoriality it is immediate that the counit del_X and comultiplication copy_X on $\mathbf{Kl}(T)$ satisfy again Equation (1.1) and Equation (1.2). For Equation (1.4) we have to additionally assume I to be terminal and that $T(I) \cong I$, i.e. T is *affine*. Then, $T(I)$ is also terminal and therefore Equation (1.4) is automatically satisfied by Remark 1.2 (iv). This whole construction is summarized in the following statement. (see [Fri20, Corollary 3.2]).

Corollary 1.8. *Let* $\mathbf{C}$ *be a Markov category and* T *a symmetric monoidal affine monad. Then the Kleisli category* $\mathbf{Kl}(T)$ *is again a Markov category.*

Example 1.9. Although we directly constructed the Markov category **FinStoch** by defining its morphisms, one may also use Corollary 1.8 for construction. For this reason, we start with the category **Set** and define the discrete distribution monad

$$\mathcal{D} : \mathbf{Set} \to \mathbf{Set}$$

where for any object X in **Set**

$$\mathcal{D}X := \{p : X \to [0,1] \mid \mathrm{supp}(p) \text{ is finite}\}$$

and for any morphism $f : X \to Y$ we have $\mathcal{D}f : \mathcal{D}X \to \mathcal{D}Y$ with

$$(\mathcal{D}f)(p) := f_\star\, p := y \mapsto \sum_{\substack{x \in X \\ f(x)=y}} p(x) \tag{1.20}$$

for every $p \in \mathcal{D}X$.

The functor $\mathcal{D}$ is indeed a symmetric monoidal affine monad where the monoidal structure on **Set** is given by the Cartesian product and the unit η as well as multiplication μ are defined as

$$\eta_X : X \to \mathcal{D}X : x \mapsto \left(\delta_x := y \mapsto \begin{cases} 1 & : x = y \\ 0 & : x \neq y \end{cases}\right) \tag{1.21}$$

and

$$\mu_X : \mathcal{D}^2 X \to \mathcal{D}X : \mathcal{M} \mapsto \left(x \mapsto \sum_{p \in \mathcal{D}X} p(x)\mathcal{M}(p)\right). \tag{1.22}$$

We refer to [FP20, Section 6] for further details. △

We will now leverage Corollary 1.8 to construct different examples of Markov categories via the *Giry monad* — a probability monad on measurable spaces — that leads to the construction of the Markov categories **Stoch** and **BorelStoch**.

1.2.1 Measurable Markov kernels on measurable spaces

In this part, we construct the category **Stoch** of measurable Markov kernels between measurable spaces. We will obtain **Stoch** as the Kleisli category of the Giry monad introduced in [Gir82]. Before introducing the monad, we will give a brief overview of the main concepts of measure theory by introducing σ-algebras, measurable maps, and (probability) measures. We refer to [Bog07, Chapter 1] for a more detailed of these concepts.

Definition 1.10. *A collection Σ_X of subsets of X is called σ-*algebra *if it satisfies*

(i) $\emptyset, X \in \Sigma_X$

(ii) Σ_X *is closed under complements, i.e.* $A \in \Sigma_X \implies X \setminus A \in \Sigma_X$

(iii) Σ_X *is closed under countable union, i.e. for* $\{A_n\}_{n\in\mathbb{N}} \subseteq \Sigma_X$*, we have* $\bigcup_{n\in\mathbb{N}} A_n \in \Sigma_X$

Furthermore, if Σ_X is a σ-algebra, we call the tuple (X, Σ_X) a measurable space.

Intuitively, the concept of σ-algebras capture those sets which are observable and hence the definition of a measure as a function on a σ-algebra. Consider the example of rolling a die, i.e. $X = \{1, \ldots, 6\}$. While being able to measure arbitrary outcomes results in $\Sigma_X = \mathcal{P}(X)$, being only able to distinguish even or odd outcomes leads to

$$\Sigma_X = \Big\{\emptyset, \{1,3,5\}, \{2,4,6\}, X\Big\}.$$

While for finite and countable sets X one might always choose the power set $\mathcal{P}(X)$ as σ-algebra, in many situations a distinctive σ-algebra construction is necessary. For example, a consistent construction of the uniform probability measure on $[0,1]$ for arbitrary subsets of $[0,1]$ is not possible[2]; however it attains a well-defined measure on the Borel σ-algebra (Equation (1.27)) [Bog07, Section 1.7]. We continue by defining the notion of functions that are compatible with the structure of σ-algebras.

Definition 1.11. *Let (X, Σ_X) and (Y, Σ_Y) be two measurable spaces. A function $f : X \to Y$ is called* Σ_X – Σ_Y – measurable *if*

$$f^{-1}(\Sigma_Y) := \Big\{f^{-1}(B) : B \in \Sigma_Y\Big\} \subseteq \Sigma_X. \tag{1.23}$$

If it is clear from context, we will omit the particular σ-algebras when talking about measurability. The concrete definition of measurability allows for defining the category **Meas** as the category containing measurable spaces as objects and measurable functions as morphisms. As we will see in the following **Meas** is a symmetric monoidal category by defining a σ-algebra on the product space $X \times Y$. For this reason, let $\mathcal{A} \subseteq P(X, \Sigma_X)$ and define

$$\sigma(\mathcal{A}) := \bigcap_{\substack{\mathcal{A} \subseteq \Sigma \\ \sigma\text{-algebra}}} \Sigma \tag{1.24}$$

to be the smallest σ-algebra containing $\mathcal{A}$. We define the product σ-algebra of two measurable spaces (X, Σ_X), (Y, Σ_Y) by setting

$$\Sigma_X \otimes \Sigma_Y := \sigma\Big(\{B_1 \times B_2 : B_1 \in \Sigma_X, B_2 \in \Sigma_Y\}\Big).$$

[2] One such example of a *non-measurable* subset is the so-called *Vitali set* [Bog07, Example 1.7.7].

Note that $\Sigma_X \otimes \Sigma_Y$ is in particular the smallest σ-algebra such that the projection maps

$$\pi_X : X \times Y \to X \quad \text{and} \quad \pi_Y : X \times Y \to Y \tag{1.25}$$

are measurable. This definition of product makes **Meas** into a Cartesian category.

We now turn to the notion of (probability) measures on arbitrary measurable spaces.

Definition 1.12. *Let* (X, Σ_X) *a measurable space. A measure on* X *is a function* $\mu : \Sigma_X \to [0, \infty]$ *such that* $\mu(\emptyset) = 0$ *and* μ *is* σ*-additive, i.e. for every collection* $\{A_n\}_{n\in\mathbb{N}} \subseteq \Sigma_X$ *of disjoint measurable sets we have*

$$\mu\left(\bigsqcup_{n\in\mathbb{N}} A_n\right) = \sum_{n\in\mathbb{N}} \mu(A_n)$$

If, in addition, $\mu(X) = 1$*, we call* μ *a* probability measure.

We are now ready to construct the category of measurable Markov kernels **Stoch**. We first introduce the Giry monad $\mathcal{P}$ and then apply Corollary 1.8 to define **Stoch** as $\mathbf{Kl}(\mathcal{P})$.

We define the functor $\mathcal{P} : \mathbf{Meas} \to \mathbf{Meas}$ by first specifying its action on objects. Every measurable space (X, Σ_X) is mapped to the set $P(X, \Sigma_X)$ containing all probability measures on (X, Σ_X).

To make this functor meaningful, we have to equip the space of probability distributions with a reasonable notion of σ-algebra. Therefore, we define $\Sigma_{\mathcal{P}X}$ to be the coarsest σ-algebra on $P(X, \Sigma_X)$ which makes the evaluation map

$$\mathsf{eval}_A : P(X, \Sigma_X) \to [0, 1] : \mu \mapsto \mu(A)$$

measurable for every $A \in \Sigma_X$. We refer to Appendix A for a detailed study and a proof of the well-definedness of this σ-algebra.

We will use in the following the short-hand notation $\mathcal{P}X$ for $P(X, \Sigma_X)$ and $\mathcal{P}^2X$ for the set of probability distributions on the measurable space $(P(X, \Sigma_X), \Sigma_{\mathcal{P}X})$.

Moreover, we map every morphism in **Meas** (i.e. every measurable map $f : X \to Y$) to the pushforward $f_\star$, defined as

$$\mathcal{P}f \coloneqq f_\star : \mathcal{P}X \to \mathcal{P}Y : \mu \mapsto \mu \circ f^{-1}.$$

$\mathcal{P}$ can be extended to a monad by defining a unit $\eta : X \to P(X, \Sigma_X) : x \mapsto \delta_x$ mapping each point x to the Dirac distribution δ_x and a multiplication

$$\mu : \mathcal{P}^2X \to \mathcal{P}X : \mathcal{M} \mapsto \int_{P(X,\Sigma_X)} p\, \mathcal{M}(\mathrm{d}p)$$

where $\mathcal{M}$ is a probability measure on the space of probability measures $P(X, \Sigma_X)$. Note that this construction indeed looks very similar to the construction of the discrete probability monad. To define for example the multiplication μ, we just replaced the discrete distribution over probability distributions (called $\mathcal{M}$) by a probability measure on the set of probability measures. Further, we replaced the sum by an integral (see Equation (1.22)). For the proof that $\mathcal{P}$ is indeed a monad and the construction of the corresponding Kleisli category we refer to Appendix A.

The monad $(\mathcal{P}, \mu, \eta)$ is further monoidal affine by defining the morphism

$$\nabla_{X,Y} : \mathcal{P}X \otimes \mathcal{P}Y \to \mathcal{P}\big(X \times Y\big) : (\mu, \nu) \mapsto \mu \otimes \nu.$$

which satisfies Equations (1.14) and (1.15). This allows for constructing $\mathbf{Stoch} := \mathbf{Kl}(P)$ as a symmetric monoidal category.

We will now analyze the morphisms in **Stoch** and their interpretation. A morphism $f : (X, \Sigma_X) \mapsto P(Y, \Sigma_Y)$ in **Stoch** is given by a map

$$f : \Sigma_Y \times X \to [0,1] : (A, x) \mapsto f(A|x)$$

satisfying that

(i) $f(-|x) : \Sigma_Y \to [0,1]$ is a probability measure on (Y, Σ_Y)

(ii) $f(S|-) : X \to [0,1]$ is measurable for every $S \in \Sigma_Y$

In other words, f is a *Markov kernel* with source (X, Σ_X) and target (Y, Σ_Y). Further, the Kleisli composition between $f : \Sigma_Y \times X \to [0,1]$ and $g : \Sigma_Z \times Y \to [0,1]$ leads to (a version of) the Chapman–Kolmogorov formula

$$g \circ f : \Sigma_Z \times X \to [0,1] : (A, x) \mapsto (g \circ f)(A|x) := \int_Y g(A|y) f(\mathrm{d}y|x). \tag{1.26}$$

Again, the Chapman-Kolmogorov formula in the measurable setting differs from the discrete setting only by replacing the sum with an integral and replacing the discrete probability distributions with probability measures.

For a full description of **Stoch** as a Markov category, it remains to define the copy and deletion maps. The copy map is defined via

$$\mathsf{copy}_{(X,\Sigma_X)} : (\Sigma_X \otimes \Sigma_X) \times X \mapsto [0,1] \qquad \mathsf{copy}_{(X,\Sigma_X)}(A \times B, x) = \begin{cases} 1 & \text{if } x \in A \cap B \\ 0 & \text{otherwise} \end{cases}$$

where $A \times B \in \Sigma_X \otimes \Sigma_X$ is a generating rectangle in the product σ-algebra. The deletion map is given by $\mathsf{del}_{(X,\Sigma_X)} := 1$. This is sensible since for any morphism $f : X \to Y \otimes Z$

$$\Big(\mathsf{del}_{(Y,\Sigma_Y)} \circ f\Big)(A|x) = \int_Y f(A, \mathrm{d}y|x),$$

where $A \in \Sigma_Z$. But this is precisely the marginal Markov kernel of Z given X.

1.2.2 Standard Borel spaces and measurable Markov kernels

In the following, we introduce the category **BorelStoch** as yet another Markov category smaller than **Stoch**. This category has some desirable properties **Stoch** is not having while being still quite general.

For a topological space $(X, \mathcal{T})$ we define the *Borel σ-algebra* to be the smallest σ-algebra containing all open sets of the topology, more precisely

$$\mathcal{B}(X) := \sigma(\mathcal{T}). \tag{1.27}$$

To construct **BorelStoch**, we restrict to a particular subclass of topological spaces, namely *Polish spaces*. A topological space is called Polish, if it is a complete separable metric space, i.e. there exists a metric generating the topology, every Cauchy sequence has a limiting point with respect to this metric, and it a countable, dense subset.

Definition 1.13. *A Borel measurable space $(X, \mathcal{B}(X))$ is called* standard Borel, *if X is a Polish space.*

This definition gives rise to a new category of measurable spaces, the category **BorelMeas**:

(i) Objects are standard Borel spaces.

(ii) Morphisms $f : X \to Y$ are measurable maps from X to Y.

Standard Borel spaces are closed under products, i.e. if $(X, \mathcal{B}(X))$ and $(Y, \mathcal{B}(Y))$ are standard Borel, then so is $X \times Y$.[3] In particular,

$$\mathcal{B}(X \times Y) := \sigma\Big(\big\{B_1 \times B_2 : B_1 \in \mathcal{B}(X), B_2 \in \mathcal{B}(Y)\big\}\Big). \tag{1.28}$$

This implies that **BorelMeas** is a symmetric monoidal full subcategory of **Meas**.[4] Moreover, the Giry monad $\mathcal{P}$ restricts to standard Borel spaces, i.e. $\mathcal{P}$: **BorelMeas** $\to$ **BorelMeas**. This follows from the fact that for every Polish space X, the space $P(X, \mathcal{B}(X))$ is again Polish. We refer to [Par67, Theorem 6.5] for details.

These two properties, namely **BorelMeas** being a Markov category (with the same copy and deletion map as in **Meas**), and $\mathcal{P}$ restricting to **BorelMeas** we can define the Markov category **BorelStoch** via Corollary 1.8.

In particular, **BorelStoch** is a full subcategory of **Stoch** since the objects restrict to standard Borel spaces while the morphisms between standard Borel spaces agree in both categories.

1.2.3 Gaussian probability theory

As a last example, we introduce the Markov category **Gauss**. This category captures all multivariate Gaussian probability distributions. Morphisms are linear transformations with additional Gaussian noise. In contrast to **Stoch** and **BorelStoch**, we construct **Gauss** not as the Kleisli category of a monad. We will directly build up **Gauss** specifying its objects and morphisms.[5]

Consider a Gaussian random variable $X \in \mathbb{R}^m$. Since linear transformations of Gaussians and addition of independent Gaussians remain Gaussian,

$$Y := MX + \xi$$

[3] For metrics d_1 on X and d_2 on Y, we have for example the metric $d := \max\{d_1, d_2\}$ on $X \times Y$. If X and Y are complete, then so is $X \times Y$.

[4] i.e. every object in **BorelMeas** is also in **Meas** and $\mathrm{Hom}_{\mathbf{BorelMeas}}(X, Y) = \mathrm{Hom}_{\mathbf{Meas}}(X, Y)$.

[5] In fact it is an open question whether **Gauss** can be obtained as a Kleisli category of a monad [Fri20, Section 6].

is a again Gaussian where M is a $n \times m$ matrix and ξ a Gaussian noise independent of X. This motivates the definition of a conditional distribution of Y in terms of $X \in \mathbb{R}^n$ via a triple (M, a, A) where M is the indicated linear transformation and ξ is uniquely specified via its expectation value $\mathbf{E}[\xi] = a$ and covariance matrix $\mathbf{Var}[\xi] = A$ defined as

$$A_{ij} = \mathbf{E}[\xi_i \xi_j] - \mathbf{E}[\xi_i] \cdot \mathbf{E}[\xi_j]$$

For this reason, we define the monoidal category **Gauss** in the following way:

(i) Objects are given by the monoid $(\mathbb{N}, +)$.

(ii) Morphisms $n \to m$ are specified by tuples (M, a, A) where $M \in \mathbb{R}^{m \times n}$, $a \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times m}$ is positive semidefinite.

It remains to define a formal composition rule. Therefore, let $(N, b, B) : m \to k$ be a second morphism representing the conditional distribution $Z = NY + \eta$. Representing Z in terms of X, we obtain

$$Z = NMX + N\xi + \eta.$$

The composed linear transformation is specified by $N \cdot M$ and for the Gaussian noise $N\xi + \eta$ we obtain

$$\mathbf{E}[N\xi + \eta] = N\mathbf{E}[\xi] + \mathbf{E}[\eta] = Na + b$$

and

$$\mathbf{Var}[N\xi + \eta] = N^t \mathbf{Var}[\xi] N + \mathbf{Var}[\eta] = N^t A N + B$$

where we have used in the first equation that ξ and η are independent.

This motivates to define the composition rule as

$$(N, b, B) \circ (M, a, A) := (NM, Na + b, NAN^t + B).$$

The composition $\circ$ is associative and the identity morphism is given by $(\mathrm{id}_n, 0, 0)$.

We now show that **Gauss** is a symmetric monoidal category. While the monoidal product on objects is given by $+$, we define the tensor product morphism acting on two separate variables by

$$(M, a, A) \otimes (N, b, B) := (M \oplus N, a \oplus b, A \oplus B).$$

This is a well-defined composition of Gaussian distributions since $\mathbf{E}[(\xi, \eta)] = (\mathbf{E}[\xi], \mathbf{E}[\eta]) = a \oplus b$ and

$$\mathbf{Var}[(\xi, \eta)] = \begin{pmatrix} \mathbf{Var}[\xi] & 0 \\ 0 & \mathbf{Var}[\eta] \end{pmatrix} = A \oplus B$$

since ξ and η are independent.

Finally, defining $\mathsf{del}_n : n \to 0$ by its only morphism $(0, 0, 0)$ and

$$\mathsf{copy}_n := \left(\begin{pmatrix} \mathrm{id}_n \\ \mathrm{id}_n \end{pmatrix}, 0, 0 \right)$$

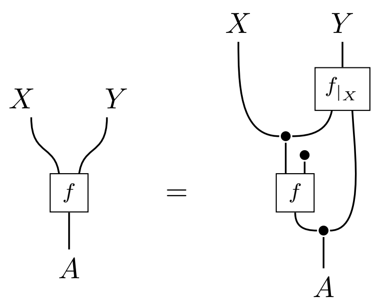
makes **Gaus** into a Markov category, where copy_n maps a vector $X \in \mathbb{R}^n$ to $(X, X) \in \mathbb{R}^{2n}$.

1.3 Existence of conditional Markov kernels

One essential task in studying causal compatibility is the computation of conditional distributions. More precisely, given a stochastic map outputting two random variables (i.e. sampling for example in a space $X \times Y$), we want to find procedures first sampling the random variable in X and then in Y.

For this reason, we introduce this behavior as an additional condition for Markov categories, calling it the existence of conditionals. In the following, we briefly review the definition following [Fri20, Section 11].

Definition 1.14. *Let* $\mathbf{C}$ *be a Markov category. We say that* $\mathbf{C}$ *has* conditionals *if for every morphism* $f : A \to X \otimes Y$*, there is a morphism* $f_{|X}$ *such that*



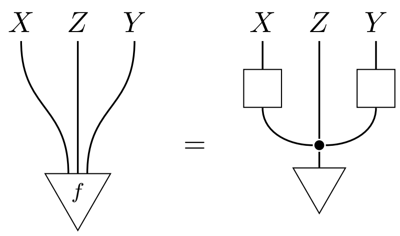
In a categorical setting, this has been studied first by Cho and Jacobs [CJ19] in the special case of probability distributions, where the authors call it *admitting disintegration*. Subsequently, Fritz [Fri20] generalized the notion to arbitrary Markov kernels.

Examples of categories having conditionals are **FinStoch**, **Gauss** as well as **BorelStoch**. However, **Stoch** does not have conditionals. For proofs of these statements, we refer to [Fri20, Examples 11.6–11.8] and references therein. Therefore, the causal compatibility criteria will be only applicable to the former three Markov categories.

1.4 Independence and Conditional Independence

One further central notion in the d-separation criterion for causal compatibility is conditional independence. In the following, we will introduce two notions of conditional independence in the setting of Markov categories. Conditional independence in a string diagrammatic language was first defined and studied by Cho and Jacobs [CJ19] for mere probability distributions and later extended to arbitrary morphisms by Fritz [Fri20]. In the following, we will briefly review the notion for probability distributions following [Fri20, Section 12].

Definition 1.15. *A morphism* $\psi : I \to X \otimes Z \otimes Y$ *displays* the conditional independence $X \perp Y \mid Z$ *if*



(1.29)

According to this definition, a finite probability distribution $P : X \times Y \times Z \to [0,1]$ in **FinStoch** shows the conditional independence $X \perp Y \mid Z$ if

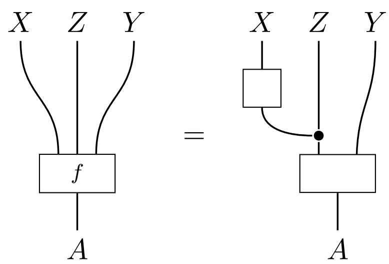
$$P(x,y,z) = P(x|z) \cdot P(y|z) \cdot P(z).$$

This coincides with the classical notion of conditional independence.

The string-diagrammatic definition satisfies the well-known semigraphoid properties [Stu06, Section 2.2.2], as shown in [CJ19, Proposition 6.10] for Markov categories with conditionals and later extended to arbitrary Markov categories [Fri20, Lemma 12.5].

With this in mind, we now introduce a novel notion of conditional independence for morphisms with nontrivial inputs. This notion is the key ingredient of the d-separation criterion presented in Chapter 3, and it is the categorical generalization of the *transitional conditional independence* introduced recently by Forré [For21, Definition 3.1].

Definition 1.16. *A morphism $f : A \to X \otimes Y \otimes Z$ in* **C** *displays the* conditional independence $X \perp Y \mid Z$ *if there exists a factorization of the form*



Remark 1.17.

(i) Note that the above definition of conditional independence is not symmetric, i.e. $X \perp Y \mid Z$ does not necessarily imply $Y \perp X \mid Z$.

Consider for example in **FinStoch** the spaces $X = Y = Z = A = \{0,1\}$ and the conditional distribution

$$P(x,y,z|a) = P(x,z) \cdot P(y|a) \qquad x \in X, y \in Y, z \in Z, a \in A$$

where $P(x,z) = \frac{1}{2}\delta_{x,z}$ and $P(y|a) = \delta_{y,a}$. This distribution displays $X \perp Y \mid Z$ since

$$P(x,y,z|a) = P(x|z) \cdot P(y,z|a)$$

with $P(x|z) = \delta_{x,z}$ and $P(y,z|a) = \frac{1}{2}\delta_{y,a}$. It does not display $Y \perp X \mid Z$ since assuming

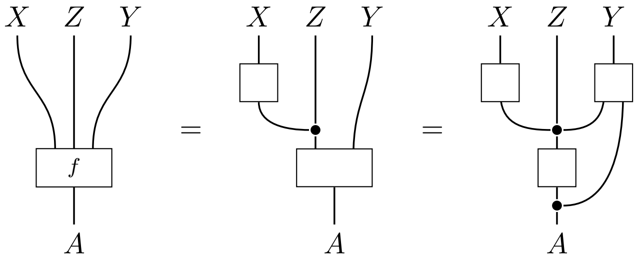
$$P(x,y,z|a) = P(y|z) \cdot P(x,z|a)$$

would imply that for $x = z$ the 2×2 identity matrix

$$I_2 = (\delta_{y,a})_{y,a\in\{0,1\}} = \Big(2 \cdot P(x,y,z|a)\Big)_{y,a\in\{0,1\}} = \Big(2 \cdot P(y|z) \cdot P(x,z|a)\Big)_{y,a\in\{0,1\}}$$

i.e. I_2 has rank 1 which is a contradiction.

(ii) If **C** has conditionals then $X \perp Y \mid Z$ rewrites to



which highlights again the asymmetry. Moreover, if $\mathcal{A}$ is trivial, then the conditional independence coincides with Definition 1.15. △

Due to the asymmetry, the output Y might contain information about the input A which cannot be retrieved just from Z. On the other hand, the output in X is generated using only the information from the output in Z. The local Markov property that we will use in Definition 3.12 explicitly highlights this asymmetry: the output of a box (corresponding to X) is independent of its non-descendants (Y) given its inputs (Z). Every global input is non-descendant to any box; however, not every global input wire is directly an input of the box itself.

Chapter 2

A categorical framework for causal models

Causal models constitute a framework to study dependencies between different observables. The most prominent example is Bayesian networks, whose causality structures are represented by directed acyclic graphs (DAGs).

In this section, we introduce a different approach in the language of Markov categories. This new framework builds up on the idea of Bayesian networks, however, it is represented by string diagrams instead of DAGs. As we will see throughout this thesis, this brings two main advantages compared to the classical approach with DAGs:

- ▷ The string diagrammatic formalism is more general than the DAG approach, i.e. it contains a broader range of causal models compared to the DAG-framework.
- ▷ The notion of d-separation on DAGs, which is used to decide causal compatibility, simplifies in the setting of string diagrams to a more intuitive formulation.

Moreover, the categorical formulation allows for studying causal compatibility independently of the concrete class of probabilities since all results are proven on the level of Markov categories. In particular, all results of deciding causal compatibility apply to morphisms in **FinStoch**, **Gauss**, and **BorelStoch**. Moreover, this approach avoids using any measure theory since the statements are proven diagrammatically.

This chapter is structured as follows: In Section 2.1 we introduce the idea of causal models as string diagrams and motivate its differences from the classical framework using DAGs. The rest of this chapter is then devoted to constructing the appropriate category capturing causal models. We introduce the category of hypergraphs in Section 2.2, which is the basic building block to defining gs-monoidal string diagrams in Section 2.3. In Section 2.4, we then review the construction of free Markov categories and finally show in Section 2.5 their relation to causal models.

Parts of Section 2.1 are based on [FK22, Section 1], and the rest of this chapter is based on [FK22, Section 4].

2.1 From directed acyclic graphs to string diagrams

The traditional framework of Bayesian networks is represented by directed acyclic graphs (DAGs). These graphs encode the underlying causal structure. Each node $v \in V$ of a DAG G corresponds to a variable X_v, and each directed arrow $w \to v$ corresponds to a direct possible causal dependence of the variable X_v on X_w.

A joint probability distribution is compatible with a given causal structure G if P factorizes into

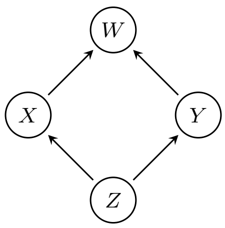
$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P(X_i \mid \mathsf{Pa}(X_i))$$

where $\mathsf{Pa}(X_i) = \{X_j : G \text{ contains the arrow } j \to i\}$ is set of parents of X_i relative to the graph G.

In this section, we introduce a different way of parametrizing causal models, based on string diagrams. This idea goes back to a work of Fong [Fon13] and has appeared in the meantime in several other works [Ris20; RW21; Gao22; JKZ19].

As already presented in the last chapters, string diagrams arise as a natural tool in symmetric monoidal categories and, therefore, also in the context of Markov categories. In this chapter, we will leverage this representation and introduce a Markov category whose morphisms will be the basic framework to address causal models in the language of categorical probability.

Rather than Markov kernels, the morphisms in a free Markov category are the string diagrams themselves, i.e. all "networks" that can be built by plugging together a set of boxes. In this way, string diagrams constitute generalized causal models. In particular, we will see that string diagrams can represent arbitrary DAG causal models. Consider for example the following DAG:

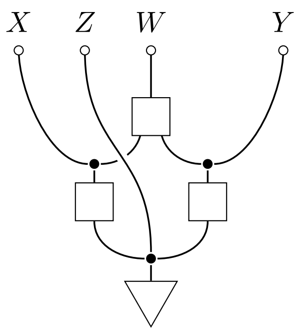


(2.1)

A discrete probability distribution P is compatible with this structure if

$$P(x,y,z,w) = P(w \mid x,y) \cdot P(x \mid z) \cdot P(y \mid z) \cdot P(z).$$

As a string diagram, this causal structure looks like this:



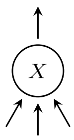
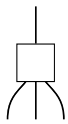
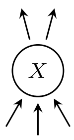
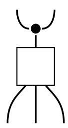
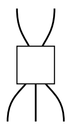
(2.2)

Each loose wire[1] represents a random variable where the name of the wire indicates the name of the variable. Further, each variable has a corresponding type, a placeholder for the concrete probability space in a concrete Markov kernel. Unless necessary, we will not explicitly mention the type of each variable.

In our setting, every wire is connected to an ”output“, representing a variable that is ”observed“ rather than marginalized over. Note that every variable then becomes an output in exactly one way. Throughout this thesis, we call such diagrams *pure blooms* (see Definition 2.7).

Every box in a pure bloom string diagram corresponds to one vertex in the DAG with inputs coming from all of its parents[2]. For example, X and Y causally influence W, so the box outputting W in the string diagram has two inputs corresponding to X and Y. Moreover, the copy-map allows sharing of the same random variable multiple times. For example, Z causally influences X and Y, therefore the corresponding copy map in the string diagram copies Z three times, one for X, one for Y, and one as an output.

The following table elucidates upon the relation between nodes in a DAG, boxes in the string diagram, and the corresponding conditional probability distribution. In particular, it shows that DAGs are only capable of representing particular local processes in a causal model.

	DAG	string diagram	conditional distribution
one output			$P(X\|ABC)$
ident. outputs			$P(XX\|ABC)$
diff. outputs	✗		$P(XY\|ABC)$

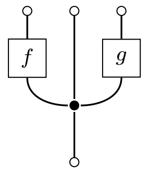
Using string diagrams as generalized causal models allow us to go beyond the DAG approach in several directions:

▷ String diagrams in Markov categories describe Markov kernels instead of just probability distributions. Therefore, the string diagram language allows for modeling causal structures with

[1]Note that we use the term ”wire“ as referring to an entire connected piece of circuitry, i.e. traversing a black dot in the diagram does not leave the wire.

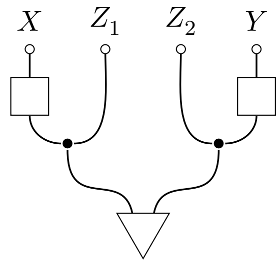
[2]This correspondence was already observed in Fong's master thesis [Fon13]; however no formal construction of free Markov categories was given.

inputs, such as



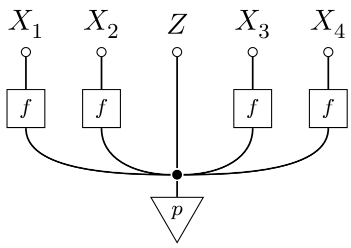
This describes a causal structure in which the input variable at the bottom does not have any particular distribution itself.

- ▷ As indicated in the table, boxes in the string diagram can have more than a single output wire. Therefore, we have a framework to represent causal structures like



which are not native to the DAG framework (see Example 3.16(i) for a detailed discussion of this structure).

- ▷ String diagrams allow for the use of identical boxes multiple times. In particular, we can represent symmetric causal structures, for example,

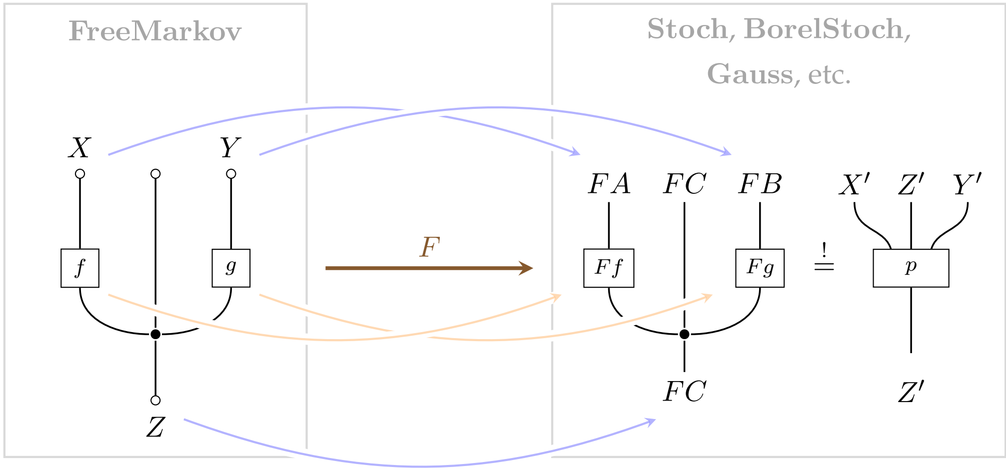


represents a causal structure in which one random variable with distribution p causally influences four others with the additional constraint that the causal mechanism must be the same for all four. Further, in this situation the types of the variables $X_1, \ldots, X_4$ must be the same.

2.1.1 Causal compatibility for Markov categories

A distribution is compatible with a causal model if it can be written as a composition in precisely the way specified by the corresponding string diagram. In other words, every type W in the string diagram is mapped to a concrete probability space FW and every box f to a concrete Markov kernel Ff.

In the category theoretic language this is captured in the following way: A morphism p in a concrete Markov category is compatible with a causal structure φ, if there exists a functor F such that $p = F\varphi$. Intuitively, this functor acts as follows:



where p is the given morphism in a concrete Markov category and **FreeMarkov** the free Markov category whose morphisms capture the causal models. Further, X has type A, Y has type B, and Z has type C.

The notion of free Markov categories are the tailored notion for all of these purposes. These categories contain precisely all morphisms which arise from a generating set of morphisms. A morphism in this category is then understood as a causal model. We divide the construction into three main parts: First, we will introduce a categorical formulation of hypergraphs as a basic structural tool. Placing appropriate restrictions we leverage them to give a combinatorial description of morphisms, including a gs-monoidal structure. This gives rise to a universal construction of a free gs-monoidal category. Finally, we review the universal construction of free Markov categories introduced in [FL22].

This chapter mainly reviews the construction of free gs-monoidal categories and free Markov categories presented in [FL22]. It introduces the necessary tools to study the causal compatibility problem in Chapter 3. A similar construction was given independently in [MZ22].

2.2 The category of hypergraphs

In the following, we introduce the category of hypergraphs **Hyp** following [Bon+16]. This category captures all (directed) hypergraphs as objects and an appropriate notion of morphisms between hypergraphs.

For this purpose, we start by defining a particular type of index category. Let **I** be the category defined in the following way:

(i) The set of objects is given by $\{(k, \ell) \mid k, \ell \in \mathbb{N}\} \cup \{\star\}$

(ii) Besides the identity morphisms, for every (k, ℓ) there are $k + \ell$ different morphisms

$$\mathsf{in}_1, \ldots, \mathsf{in}_k, \mathsf{out}_1, \ldots, \mathsf{out}_\ell : (k, \ell) \to \star.$$

Note that it is not necessary to specify any composition rule in **I** since no compositions exists except the trivial ones.

Definition 2.1 (Hypergraph category)**.** *A functor* $G : \mathbf{I} \to \mathbf{Set}$ *is called a* hypergraph*. We define the functor category*[3]

$$\mathbf{Hyp} := \mathbf{Set}^{\mathbf{I}}$$

Intuitively, the functor G characterizes our common interpretation of (directed) hypergraphs in the following way:

(i) $W(G) := G(\star)$ is the vertex set (or in string diagrammatic language the set of wires)

(ii) $B_{k,\ell}(G) := G((k,\ell))$ is the set of edges with k inputs and ℓ outputs (or string diagrammatically the set of boxes)

(iii) $G(\mathsf{in}_i)$ specifies the wire connected to the ith input of every box.

(iv) $G(\mathsf{out}_j)$ specifies the vertex connected to the jth output of every box.

Note that the set of boxes and the set of wires can be chosen to be infinite; however, the number of inputs and outputs are always finite. An example of a pictorial representation of a hypergraph is shown in Figure 2.1.

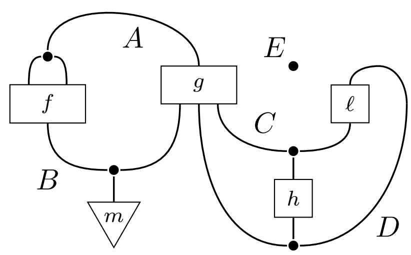


Fig. 2.1. Pictorial representation of a hypergraph with vertex set $\{A, B, C, D, E\}$ and edge set $\{f, g, h, m, n\}$. The vertices are represented by wires and the edges are represented by boxes whose input vertices are wires from the bottom and output vertices are wires from the top. For example, the edge f has one input connected to vertex B and two outputs both connected to vertex A. Vertex E is not connected to any edge.

For a box $b \in B_{k,\ell}(G)$ and a wire $A \in W(G)$, we define the cardinalities

$$\mathsf{in}(b, A) := \left|\{j \in \{1, \ldots, \ell\} : \mathsf{in}_j(b) = A\}\right|,$$
$$\mathsf{out}(b, A) := \left|\{i \in \{1, \ldots, k\} : \mathsf{out}_i(b) = A\}\right|$$

counting the number of incoming or outgoing wires of type A in the box b. Further, we also denote the sets of inputs and outputs as

$$\mathsf{in}(b) := \{\mathsf{in}_i(b) : i \in \{1, \ldots, \ell\}\},$$
$$\mathsf{out}(b) := \{\mathsf{out}_i(b) : i \in \{1, \ldots, k\}\},$$

where repeated wires are counted only once.

[3] A functor category $\mathbf{D}^{\mathbf{C}}$ is defined as follows: The objects are given by all functors $\mathbf{C} \to \mathbf{D}$. Morphisms are the natural transformations between functors $\alpha : F \Rightarrow G$. Composition is given by the *vertical composition* of natural transformations, i.e. for $\alpha : F \Rightarrow G$ and $\beta : G \Rightarrow H$, we have $\beta \circ \alpha : F \Rightarrow H$ with $(\beta \circ \alpha)_X := \beta_X \circ \alpha_X$ (see [Lei14, Construction 1.3.6] or [Per19, Section 1.4.5] for details).

Let us next analyze the morphisms in the category **Hyp**. A morphism $\alpha : F \to G$ is precisely a natural transformation $\alpha : F \Rightarrow G$ since **Hyp** is a functor category. Natural transformations are fully determined by their components; in this situation,

$$\alpha_\star : W(F) \to W(G) \quad \text{and} \quad \alpha_{(k,l)} : B_{(k,\ell)}(F) \to B_{(k,\ell)}(G) \quad \text{for all } k, \ell \in \mathbb{N}$$

satisfying the naturality conditions, i.e. the following diagrams commute:

$$\begin{array}{ccc} B_{k,\ell}(F) & \xrightarrow{\mathsf{in}_i} & W(F) \\ \downarrow{\scriptstyle \alpha_{k,\ell}} & & \downarrow{\scriptstyle \alpha_{k,\ell}} \\ B_{k,\ell}(G) & \xrightarrow{\mathsf{in}_i} & W(G) \end{array} \qquad \begin{array}{ccc} B_{k,\ell}(F) & \xrightarrow{\mathsf{out}_j} & W(F) \\ \downarrow{\scriptstyle \alpha_{k,\ell}} & & \downarrow{\scriptstyle \alpha_{k,\ell}} \\ B_{k,\ell}(G) & \xrightarrow{\mathsf{out}_j} & W(G). \end{array}$$

Put differently, every morphism in **Hyp** is a structure-preserving map between hypergraphs. More precisely, if box f is incident to wire A in its ith input in the hypergraph F, then the same applies to their images with respect to α in the hypergraph G.

As already pointed out, **Hyp** might contain an infinite set of boxes and wires. For the rest of this thesis, we will mainly restrict to *finite hypergraphs*, i.e. functors G where $W(G)$ and

$$B(G) \coloneqq \coprod_{k,\ell \in \mathbb{N}} B_{k,\ell}(G).$$

are finite sets.[4] We will denote the corresponding category as **FinHyp**.

2.3 gs-monoidal string diagrams

The pictorial representation of hypergraphs already indicates their potential use for modeling causal structures in a categorical framework. For the rest of this chapter, we use hypergraphs to construct free Markov categories generated by a fixed hypergraph Σ. Free Markov categories are Markov categories whose morphisms are string diagrams built up from the boxes in Σ.

However, several apparent problems prohibit us from using hypergraphs directly as a framework representing string diagrams:

- ▷ Hypergraphs can contain loops, i.e. an output wire of a box might be indirectly connected to one of its inputs and therefore feed back.
- ▷ While splitting a wire into two represents copying of values and makes sense in any Markov category, merging of wires as in Figure 2.1 does not make sense. In other words, a Markov category is, in general, no hypergraph category (according to [FS18a]) due to its missing Frobenius generators.
- ▷ A hypergraph has no intrinsic input and output. This additional information has to be additionally specified via a cospan over hypergraphs.

We resolve these issues by restricting to acyclic and left monogamous hypergraphs and by representing gs-monoidal string diagrams by cospans thereof (see Definition 2.2).

[4] Note that this is a stronger condition compared to considering the category $\mathbf{FinSet}^{\mathbf{I}}$. In the latter category, the set $B(G)$ might still be infinite since the restriction of being a finite set only applies to all $B_{k,\ell}(G)$.

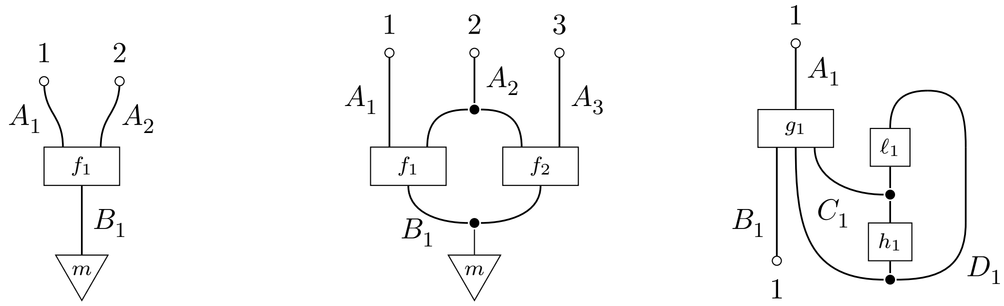
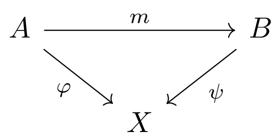


Fig. 2.2. Examples of hypergraphs in $\mathbf{FinHyp}/\Sigma$ where Σ is the hypergraph from Figure 2.1. The hypergraph morphisms are given by mapping the vertices A_i to A in both examples as well as mapping the two separate boxes f two the only morphism f in Σ. The second hypergraph is not left-monogamous since A_2 arises from two outputs, the third hypergraph is not acyclic.

We will start by introducing the notion of slice categories. Given a category $\mathbf{C}$ and an object X in $\mathbf{C}$, the slice category $\mathbf{C}/X$ is defined as follows:

(i) Objects are given by morphisms $\varphi : A \to X$

(ii) Morphisms from $\varphi : A \to X$ to $\psi : B \to X$ are given by morphisms $m : A \to B$ such that the diagram

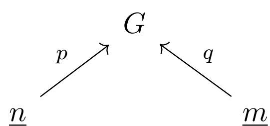


commutes.

Given a finite hypergraph Σ, the slice category $\mathbf{FinHyp}/\Sigma$ is given by hypergraphs G with a morphism $\varphi : G \to \Sigma$. In other words, any G can be understood as a hypergraph that is labeled by boxes and wires in Σ. We refer to Figure 2.2 for examples. These already show one feature explicitly, namely the possibility of having identical morphisms in one string diagram multiple times.

We are now ready to define the main ingredient for free gs-monoidal and free Markov categories, namely gs-monoidal string diagrams.

Definition 2.2. *Given a hypergraph Σ. A* gs-monoidal string diagram *for Σ is a cospan in $\mathbf{FinHyp}/\Sigma$ of the form*



(2.3)

satisfying that

(i) G is acyclic, *i.e. there is no family of wires $(A_0, \dots, A_{n-1})$ and boxes $(f_0, \dots, f_{n-1})$ such that*

$$\mathsf{in}(f_i, A_i) \geq 1 \quad \textit{and} \quad \mathsf{out}(f_i, A_{i+1}) \geq 1$$

where addition is taken modulo n.

(*ii*) *G is* left monogamous, *i.e. for every wire W we have*

$$|p^{-1}(W)| + \sum_{f \in E(G)} \mathsf{out}(f, W) \leq 1 \tag{2.4}$$

In the above definition, we have made use of the *discrete hypergraph* $\underline{n}$ which is defined via $W(\underline{n}) = \{1, \ldots, n\}$ and $B(\underline{n}) = \emptyset$. Hence, the morphism $p : \underline{n} \to G$ in the cospan assigns different wires in G to $1, \ldots, n$. These labels specify the global input wires in the hypergraph. Similarly, $q : \underline{m} \to G$ maps labels to the m different output wires in the hypergraph.

Pictorially, an acyclic hypergraph does not contain a family of wires which form a loop. Further, left monogamy requires that every wire in the hypergraph arises as either a global input or as an output of a box in precisely one way, ensuring that no "merging" of wires occurs. See Figure 2.2 for an illustration of these properties. We define the hypergraph morphism $\mathsf{type} : G \to \Sigma$ by mapping the wires A_i to A in both examples and mapping the two distinct boxes f_i to the only morphism f in Σ, etc. The first and second hypergraphs are acyclic, while the third one is not. The first and third hypergraphs are left monogamous, while the second one is not since A_2 is an output of two boxes. Finally, we have $n = 0$ in the first two cases so that the left cospan leg p is trivial, while the right leg q maps each number i to the ith overall output wire as counted from left to right.

The notion of gs-monoidal string diagrams is the main ingredient defining a gs-monoidal category whose morphisms are freely generated by the boxes and wires in Σ. We define the category $\mathbf{FreeGS}_\Sigma$ as follows:

- ▷ Objects are given by the morphisms $\sigma : \underline{n} \to \Sigma$ for $n \in \mathbb{N}$.
- ▷ Morphisms are the "isomorphism classes" of gs-monoidal diagrams.

Each object in $\mathbf{FreeGS}_\Sigma$ corresponds to an element of the free monoid generated by $W(G)$ or equivalently a word over the alphabet $W(G)$. Note that we will never spell out the labeling σ explicitly because it is usually clear from the string diagram notation. The pushout of the cospans defines the composition of two gs-monoidal diagrams. In detail, given two cospans $\underline{n} \to G \leftarrow \underline{m}$ and $\underline{m} \to H \leftarrow \underline{r}$ we have

$$\begin{array}{ccccccc} & & & G +_{\underline{m}} H & & & \\ & & \overset{u}{\nearrow} & & \overset{v}{\nwarrow} & & \\ & G & & & & H & \\ \overset{p}{\nearrow} & & \overset{q}{\nwarrow} & & \overset{f}{\nearrow} & & \overset{h}{\nwarrow} \\ \underline{n} & & & \underline{m} & & & \underline{r} \end{array} \tag{2.5}$$

This can be understood on the level of string diagrams as follows: The composed string diagram of two separate diagrams G and H arises by gluing the output wires of G together with the input wires of H.

To show that $\mathbf{FreeGS}_\Sigma$ is indeed a category, it is left to show that the composition of two gs-monoidal string diagrams is again gs-monoidal, i.e. the underlying hypergraph is left monogamous as well as acyclic. We will not further elaborate on this here; instead, we refer to [FL22, Lemma 3.8] for a proof.

As the name already suggests, $\mathbf{FreeGS}_\Sigma$ is also a gs-monoidal category. The symmetric monoidal product on the level of objects is given by the coproduct in $\mathbf{Set}$. This is realized by the disjoint union, i.e.

$$G + H : \mathbf{I} \to \mathbf{Set}$$

is defined as

$$(G + H)(\star) := G(\star) \sqcup H(\star) \quad \text{and} \quad (G + H)((k, l)) := G((k, l)) \sqcup H((k, l))$$

together with the coproduct morphisms $(G + H)(\mathsf{in}_i) := G(\mathsf{in}_i) + H(\mathsf{in}_i)$ as well as $(G + H)(\mathsf{out}_j) := G(\mathsf{out}_j) + H(\mathsf{out}_j)$.

Applying this procedure to the discrete graphs $\underline{n}$ and $\underline{m}$ we obtain the graph $\underline{n+m}$. Therefore, we define the monoidal structure on the morphisms as

$$\left(\underline{n} \xrightarrow{p} G \xleftarrow{q} \underline{m} \right) \otimes \left(\underline{k} \xrightarrow{r} H \xleftarrow{s} \underline{\ell} \right) := \underline{n+k} \xrightarrow{p+r} G + H \xleftarrow{q+s} \underline{m+\ell} \tag{2.6}$$

String diagrammatically the monoidal product of two hypergraphs in this setting is the union of both diagrams without any connection in between. This precisely reflects the string diagrammatic representation of $f \otimes g$ in a symmetric monoidal category.

Finally, the comonoid structure is given by the cospans

$$\underline{n} \xrightarrow{\mathrm{id}} \underline{n} \xleftarrow{\mathrm{id+id}} \underline{n+n} \quad \text{and} \quad \underline{n} \xrightarrow{\mathrm{id}} \underline{n} \longleftarrow \emptyset \tag{2.7}$$

which satisfy the commutativity relations in Equations (1.1), (1.2), and (1.3).

Finally, we present the main theorem of this section, which shows that $\mathbf{FreeGS}_\Sigma$ is the appropriate definition for our purpose. In particular, the following theorem states that $\mathbf{FreeGS}_\Sigma$ is the free gs-monoidal category generated by the morphisms which are represented as boxes in the hypergraph Σ.

For this purpose, note that every monoidal category can be seen as a hypergraph whose wires are the objects and whose boxes with k inputs and ℓ outputs are the morphisms from a k-fold tensor product to an ℓ-fold tensor product of objects in $\mathbf{C}$. Formally, this assignment defines a functor

$$\mathsf{hyp} : \mathbf{MonCat} \to \mathbf{Hyp}$$

where $\mathbf{MonCat}$ is the category of monoidal categories.

Theorem 2.3. *Let $\mathbf{C}$ be a strict gs-monoidal category whose monoid of objects is freely generated by $W(\Sigma)$. Restricting along the morphism*

$$\Sigma \to \mathsf{hyp}(\mathbf{FreeGS}_\Sigma)$$

there is a bijection between

(*i*) *identity-on-objects strict gs-monoidal functors* $\mathbf{FreeGS}_\Sigma \to \mathbf{C}$

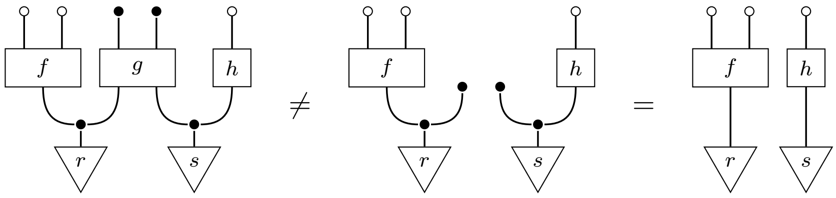
(*ii*) *identity-on-wires hypergraph morphisms* $\Sigma \to \mathsf{hyp}(\mathbf{C})$

For a proof and a different 2-categorical universal property we refer to [FL22, Theorem 4.1, Corollary 4.7].

Intuitively, this result can be understood as follows: Consider a functor in (i) which is completely determined by mapping morphisms in $\mathbf{FreeGS}_\Sigma$ to morphisms in $\mathbf{C}$. This functor induces a unique hypergraph morphism in (ii). In other words, there is a unique labeling of boxes in Σ using boxes in $\mathsf{hyp}(\mathbf{C})$. Conversely, labeling boxes by morphisms in $\mathbf{C}$ induces a unique gs-monoidal functor in (i). This bijection implies that any labeling is already fully characterized by labels on the generating set of morphisms.

2.4 Free Markov categories

$\mathbf{FreeGS}_\Sigma$ is typically not a Markov category. For instance, the first step in the following transformation in the first step does not hold in general since the (cospans of) hypergraphs are not isomorphic. In contrast, the second equation does hold:



(2.8)

In the following, we define the free Markov category $\mathbf{FreeMarkov}_\Sigma$ by taking a quotient of $\mathbf{FreeGS}_\Sigma$ which enforces Equation (1.4), so that also the first equation above becomes true.

Definition 2.4. *Let*

$$\varphi \quad := \quad \underline{n} \xrightarrow{\;p\;} G \xleftarrow{\;q\;} \underline{m} \tag{2.9}$$

be a gs-monoidal string diagram.

(*i*) *A box $b \in B(G)$ is called* eliminable *if each output of b gets discarded, i.e. if for every $W \in W(G)$ such that* $\mathsf{out}(b, W) > 0$ *we have*

  (*a*) $q^{-1}(W) = \emptyset$.

  (*b*) $\mathsf{in}(b', W) = 0$ *for every box $b' \in B(G)$.*

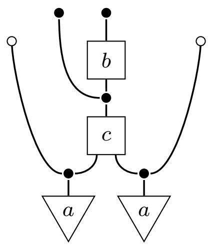
(*ii*) φ *is called* normalized *if it contains no eliminable boxes.*

Every gs-monoidal string diagram has a normalized version obtained by iteratively applying the rule of Equation (1.4) to any eliminable box. Since every diagram is finite, this procedure terminates after finitely many steps, and we reach the normalized version. In addition, this diagram is unique since the order of elimination does not matter.

The free Markov category $\mathbf{FreeMarkov}_\Sigma$ is now defined as $\mathbf{FreeGS}_\Sigma$, but with morphisms restricted to the normalized gs-monoidal string diagrams. The composition of morphisms is then defined as composition in $\mathbf{FreeGS}_\Sigma$ followed by normalization since the composition of two normalized diagrams need not be normalized. See [FL22] for details.

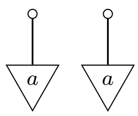
Example 2.5. The morphism

$\varphi \quad =$



is not normalized, since the output of b gets discarded. Applying Equation (1.4), also the output of c gets discarded. Therefore the normalization of φ is

$\mathsf{norm}(\varphi) \quad =$



△

In general, normalizing a gs-monoidal string diagram defines a strict gs-monoidal functor

$$\mathsf{norm} : \mathbf{FreeGS}_\Sigma \to \mathbf{FreeMarkov}_\Sigma$$

that is identity-on-objects.

2.5 Generalized causal models

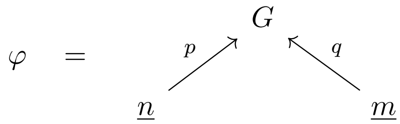
We now introduce the notion of a generalized causal model and define when a morphism in a Markov category is considered compatible with a generalized causal model.

Definition 2.6 ([FL22, Definition 7.1])**.** *Given a hypergraph* Σ*, a* generalized causal model *is a normalized gs-monoidal string diagram* (*see Definition* 2.2) *over* Σ *such that* q *is injective.*

Intuitively, a generalized causal model is a morphism in $\mathbf{FreeMarkov}_\Sigma$ where the injectivity of q ensures that each wire is connected to at most one output. This lets us identify any global input and any global outputs with a wire in $W(G)$ (see Notation 2.10). In the traditional terminology of random variables, the injectivity of q guarantees that different outputs correspond to different variables.

One relevant subclass of generalized causal models is the class of pure blooms. These morphisms represent causal models in which all variables are observed, i.e. every wire is an output in precisely one way, such as in Example 2.8(i). This notion of causal models are introduced in [FL22]. We refer to this source for a detailed study of pure blooms, in particular in the context of bloom-circuitry factorizations.

Definition 2.7. *Let φ be a generalized causal model represented by a gs-monoidal string diagram*



Then φ is called pure bloom *if q is a bijection on wires.*

Pure bloom morphisms will be of special interest in Chapter 3. While the soundness[5] of the d-separation criterion holds for arbitrary generalized causal models (see Corollary 3.11), completeness holds only for pure bloom morphisms (see Theorem 3.13).

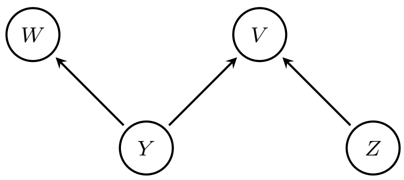
Example 2.8. In the following, we will study examples and non-examples of (pure bloom) generalized causal models.

(i) Let $G = (\mathcal{V}, A, s, t)$ be a directed acyclic graph where $\mathcal{V}$ is the set of nodes, A is the set of arrows, and $s, t : A \to \mathcal{V}$ are the source and target maps. The causal model generated by G can be implemented as a morphism in $\textbf{FreeMarkov}_\Sigma$ by defining Σ by setting the set of generating objects as $\mathcal{V}$ and morphisms being of the form

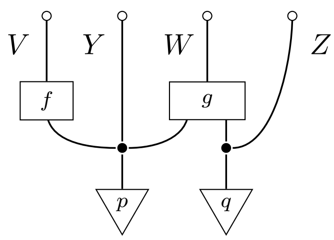
$$\bigotimes_{W \in \mathsf{An}(V)} W \to V$$

where $\mathsf{An}(V)$ is the set of ancestors of $V \in \mathcal{V}$ in G. The causal structure is then the induced hypergraph Σ itself.

For example, the causal structure



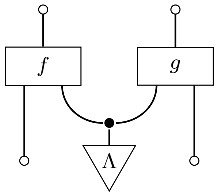
reads as a string diagram as follows



In this situation, every wire of the gs-monoidal string diagram is automatically connected to an output, i.e. φ is a pure bloom morphism. In particular, every string diagram arising from a DAG is a pure bloom morphism without any input wires.

[5] i.e. the existence of a link between d-separation and conditional independence is a necessary criterion for causal compatibility with a causal model.

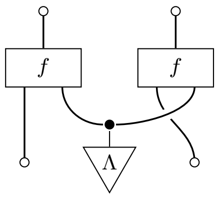
(ii) Unlike causal models, the morphisms in $\mathbf{FreeMarkov}_\Sigma$ model Markov kernels instead of mere probability distributions. More precisely, they allow for equipping Markov kernels with an additional internal structure rather than only probability distributions as the DAG setting does. One relevant example in physics in this context are Bell scenarios which are morphisms of the form



(2.10)

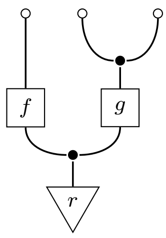
where Λ is a shared hidden variable between two separated parties and A, B are the local inputs which might not have a fixed distribution.

(iii) An additional built-in feature of generalized causal models is the possibility of including symmetry constraints in the morphism. For example, considering Equation (2.10) with the additional constrained $f = g$ leads to the morphism



(2.11)

(iv) The gs-monoidal string diagram



is a non-example to generalized causal models. This follows since a single wire is connected to multiple outputs. We exclude such examples to avoid technical problems regarding categorical d-separation in Section 3.1. △

To define causal compatibility, we make the following assumption for the rest of this thesis for convenience:

Assumption 2.9. *Throughout,* $\mathbf{C}$ *is a strict Markov category.*

Although most examples like **FinStoch**, **BorelStoch**, or **Stoch** fail strictness, this does not exclude these examples since we can always work with a strictification instead [Fri20, Theorem 10.17], which satisfies Assumption 2.9. On the other hand, our free Markov categories $\mathbf{FreeMarkov}_\Sigma$ already satisfy this condition "on the nose". In any case, Assumption 2.9 is a useful convenience that holds without loss of generality.

Notation 2.10. *For the rest of this thesis, we will assume that* φ *is a generalized causal model with*

$$\varphi \quad := \quad \begin{array}{ccccc} & & G & & \\ & {\scriptstyle p} \nearrow & & \nwarrow {\scriptstyle q} & \\ \underline{n} & & & & \underline{m} \end{array}$$

which becomes a cospan in $\mathbf{FinHyp}/\Sigma$ *through* $\mathsf{type} : G \to \Sigma$.

We identify inputs and outputs with the wires they map to under p *and* q *and refer to them as such. In particular, we define*

$$\mathsf{in}(\varphi) := p(\underline{n}) \subseteq W(\varphi) \tag{2.12}$$

$$\mathsf{out}(\varphi) := q(\underline{m}) \subseteq W(\varphi) \tag{2.13}$$

for the set of all input/output wires. If φ *is a pure bloom morphism, then* $\mathsf{out}(\varphi) = W(\varphi)$.

Note that φ is a morphism

$$\varphi : \bigotimes_{i=1}^{n} \mathsf{type}(p(i)) \longrightarrow \bigotimes_{j=1}^{m} \mathsf{type}(q(j))$$

in $\mathbf{FreeMarkov}_{\Sigma}$.

In the following, we present the notion of causal compatibility for a generalized causal model φ. Intuitively, a morphism f in any Markov category $\mathbf{C}$ is compatible with φ if we can plug in a morphism from $\mathbf{C}$ into every box in $B(\Sigma)$ in such a way that the composite is exactly f, and such that the global input and output wires of φ correspond to a given tensor factorization of the domain and codomain of f:

Definition 2.11 (Compatibility)**.** *For* Σ *a hypergraph, let* φ *be a generalized causal model. Let further*

$$f : \bigotimes_{i=1}^{n} W_i' \to \bigotimes_{j=1}^{m} V_j'$$

be a morphism in any Markov category $\mathbf{C}$ *satisfying Assumption 2.9, equipped with a fixed tensor decomposition of its domain and codomain as indicated.*

We call f compatible with φ *if there exists a strict Markov functor*[6] $F : \mathbf{FreeMarkov}_{\Sigma} \to \mathbf{C}$ *such that:*

(*i*) *We have*

$$W_i' = F(\mathsf{type}(p(i))), \qquad V_j' = F(\mathsf{type}(q(j))) \tag{2.14}$$

for all input indices $i = 1, \ldots, n$ *and output indices* $j = 1, \ldots, k$.

(*ii*) $f = F(\varphi)$.

This generalizes the functorial definition of causal compatibility as first studied by Fong [Fon13].

Note that the functor F must assign to every type (i.e. a wire in Σ) a corresponding object in the category $\mathbf{C}$. This implies that wires in $W(G)$ with identical types must map to the same object in $\mathbf{C}$.

[6] i.e. a strict symmetric monoidal functor that preserves the comonoid structure.

For example, one may consider a situation where f is a probability distribution with no inputs, and all output variables are real-valued. In this case, we have $V_j' = \mathbb{R}$ for all j, and one may want to consider a causal model φ in which all wires are similar.

Similarly, the hypergraph morphism $\mathtt{type} : G \to \Sigma$ assigns a specific "type" box in Σ to each box in G. This means that under F, any two boxes with the same type must map to the same morphism in $\mathbf{C}$. This is why generalized causal models, in our sense, can naturally incorporate the condition that several causal mechanisms must be the same, namely when choosing the types enforcing this.

In the following, we denote for every wire $X \in W(G)$ in φ the corresponding object $F(\mathtt{type}(X))$ in $\mathbf{C}$ by X'. Similarly, for every set of wires $\mathcal{W} \subseteq W(G)$ in φ, we denote the corresponding multiset of wires in $\mathbf{C}$ by $\mathcal{W}'$. For the rest of this thesis, we will associate this multiset with the corresponding tensor product in $\mathbf{C}$ obtained by tensoring its elements, where we ignore the question of how to order the factors.

Chapter 3

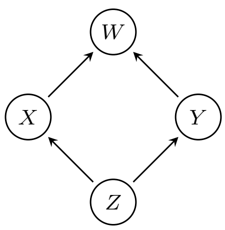
Characterizing causal compatibility for generalized causal models

Causal models are not only efficient representations of probability distributions, but they also unravel causal relationships between variables. However, checking compatibility with certain models might be a challenging task. In this chapter, we present the d-separation criterion as a way to decide causal compatibility for causal models with access to all variables (i.e. no hidden variables) or in the language of string diagrams for pure bloom causal models.

The main goal of this chapter is to prove that the d-separation criterion [Pea09, Section 1.2.3] correctly detects causal compatibility not just in discrete probability but in all Markov categories with conditionals and, therefore, also probability distributions in **Gauss** or **BorelStoch**. To this end, we introduce a novel, categorical notion of d-separation phrased in terms of the connectedness of the gs-monoidal string diagram representing the causal model. We show that this notion coincides with the classical notion of d-separation whenever the latter applies.

The notion of d-separation for DAGs is a criterion relating conditional independence of a probability distribution to the causal compatibility with an underlying DAG. In this chapter, we present a notion of d-separation from a different perspective, namely a categorical notion of d-separation. This notion looks very different, is much simpler than the classical notion, and applies to generalized causal models. However, we prove that it coincides with the classical one when considering causal models on DAGs.

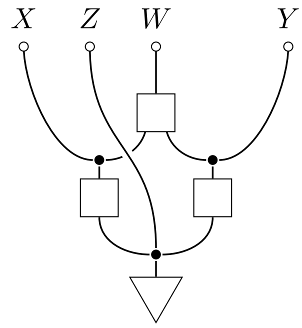
An output wire Z *categorically d-separates* the output X from output Y if X and Y become disconnected upon *marginalizing* over all wires that are not involved in the d-separation relation and *removing* the wire Z. Consider for example, again the DAG



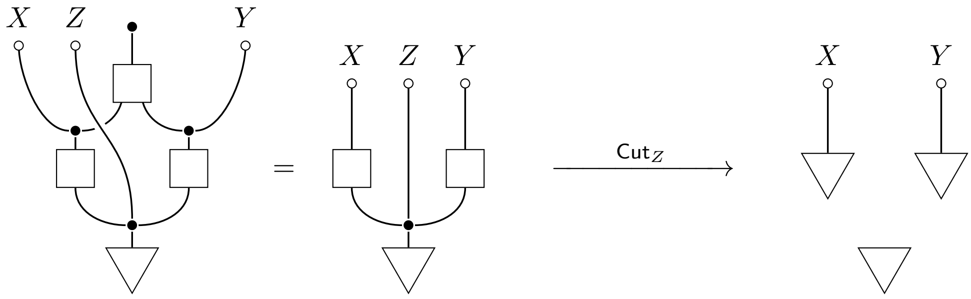
already introduced in Section 2.1. Z classically d-separates X from Y, based on the fact that the only paths between X and Y are the collider $X \to W \leftarrow Y$ and the fork $X \leftarrow Z \to Y$.[1] In the corresponding

[1]We define classical d-separation in the language of string diagrams in Definition 3.4. For an introduction to the traditional notion of d-separation in the language of DAGs, we refer to [Pea09, Section 1.2.3].

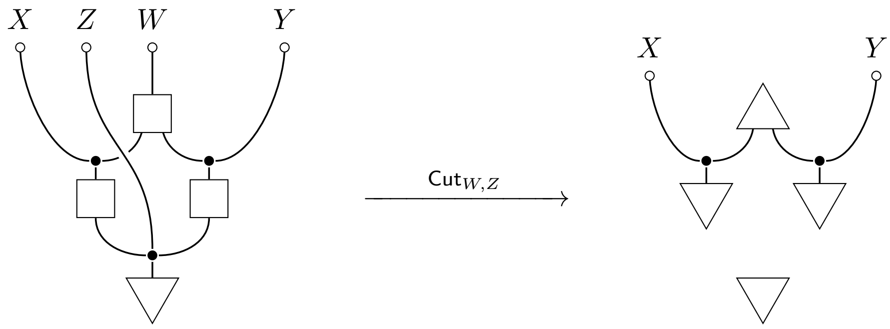
string diagram,



we witness categorical d-separation by first marginalizing over W, then removing the Z wire, and finally observing that X and Y are disconnected, pictorially:



On the other hand, X is not d-separated from Y by W and Z due to the collider $X \to W \leftarrow Y$. In the string diagram, this is apparent since upon removing the wires Z and W,



X and Y are still connected.

In Section 3.2, we prove that the categorical d-separation criterion applies to generalized causal models in Markov categories. First, we show in Corollary 3.11 the soundness of the criterion and in Theorem 3.13 the completeness of the criterion. More specifically, the soundness states that if a morphism is compatible with a generalized causal structure (according to Definition 2.11) then categorical d-separation in the gs-monoidal string diagram implies conditional independence of the compatible morphism (according to Definition 1.16). Hence, conditional independence for every d-separated triple of wires is necessary for compatibility. This link between d-separation and conditional independence is known as the *global Markov property* (see Definition 3.12). For the completeness, we show that even the weaker *local Markov property* suffices for compatibility of a morphism with a causal model. A central assumption for the proof is the existence of conditionals (see Definition 1.14); hence, this d-separation criterion applies to discrete variables, Gaussian random variables, and random variables arising from a distribution in a standard Borel space. Moreover, it generalizes the classical d-separation criterion also in the sense of allowing random processes with inputs, i.e. Markov kernels instead of mere probability distributions.

This chapter is structured as follows. In Section 3.1 we rigorously define categorical d-separation and show that it sensibly generalizes classical d-separation. Finally, we show in Section 3.2 the main result of this chapter. First, we show that d-separation implies conditional independence for compatible morphisms (Corollary 3.11). Second, we show that d-separation completely determines causal compatibility for pure bloom causal models (Theorem 3.13).

This chapter is mainly based on [FK22, Section 1, Section 6].

3.1 Categorical d-separation in string diagrams

In the following, we will introduce the notion of categorical d-separation. As already pointed out, categorical d-separation is based on checking disconnectedness on the string diagram which arises from the causal model by removing a set of wires.

Therefore, we will start by formalizing the cutting procedure on gs-monoidal string diagrams. Consider a gs-monoidal string diagram

$$\varphi \quad = \quad \underline{n} \xrightarrow{\;p\;} G \xleftarrow{\;q\;} \underline{m}$$

and a set of output wires $\mathcal{Z} \subseteq \mathsf{out}(\varphi)$, we define a new gs-monoidal string diagram $\mathsf{Cut}_{\mathcal{Z}}(\varphi)$ in the following way:

(i) Its underlying hypergraph H contains the same set of boxes $B(H) = B(G)$ and those wires which are not in $\mathcal{Z}$, i.e. $W(H) = W(G) \setminus \mathcal{Z}$.

(ii) Accordingly, the input and output wires of each box in $B(H)$ remain the same up to all wires in $\mathcal{Z}$ which are removed. Hence the arities of the boxes are lowered correspondingly.

(iii) We remove the in- and outputs connected to the wires in $\mathcal{Z}$ by lowering the cardinality of the discrete hypergraphs $\underline{n}$ and $\underline{m}$ accordingly.

This procedure leads to the cospan

$$\mathsf{Cut}_{\mathcal{Z}}(\varphi) \quad := \quad \underline{n'} \xrightarrow{\;p'\;} H \xleftarrow{\;q'\;} \underline{m'}$$

Note that $\mathsf{Cut}_{\mathcal{Z}}(\varphi)$ is usually not a morphism in $\mathbf{FreeMarkov}_{\Sigma}$ anymore. This is because the boxes in $\mathsf{Cut}_{\mathcal{Z}}(\varphi)$ are generally not in Σ. Further the obtained string diagram also misses normalization. However, it can still be understood as a morphism in the gs-monoidal category $\mathbf{FreeGS}_{H}$.

Before defining categorical d-separation, we introduce some notation regarding paths on wires in string diagrams.

Definition 3.1. *Let φ be a gs-monoidal string diagram in $\mathbf{FreeGS}_{\Sigma}$.*

(*i*) *An* undirected path *between two wires* $X, Y \in W(G)$ *is a sequence of wires*

$$X = W_1, W_2, \dots, W_n, W_{n+1} = Y$$

together with a sequence of boxes $b_1, \dots, b_n \in B(G)$ *such that*

$$\mathsf{in}(b_i, W_i) + \mathsf{out}(b_i, W_i) \geq 1 \quad \textit{and} \quad \mathsf{in}(b_i, W_{i+1}) + \mathsf{out}(b_i, W_{i+1}) \geq 1.$$

If there exists an undirected path between X *and* Y*, then we write* $X - Y$.

(*ii*) *For two wires* $A, B \in W(G)$*, we write* $A \to B$ *if there exists a box* $b \in B(G)$ *such that*

$$\mathsf{in}(b, A) = 1 \quad \textit{and} \quad \mathsf{out}(b, B) = 1. \tag{3.1}$$

(*iii*) *For two wires* $A, B \in W(G)$*, we write* $A \twoheadrightarrow B$ *if there exists a sequence of wires* $W_1, \dots, W_n \in W(G)$ *such that*

$$A \to W_1 \to \dots \to W_n \to B. \tag{3.2}$$

A directed path in φ only allows input to output traversals. In contrast, an undirected path may traverse a box not just from input to output or vice versa but also from input to input or output to output.

We are now ready to define the notion of d-separation. The intuitive idea behind categorical d-separation was communicated to us by Rob Spekkens (see also [FL22, Remark 7.2]).

Definition 3.2 (Categorical d-separation)**.** *Let* φ *be a generalized causal model. For three disjoint sets of output wires* $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subseteq \mathsf{out}(\varphi)$*, we say that* $\mathcal{Z}$ d-separates $\mathcal{X}$ *and* $\mathcal{Y}$ *if*

$$\mathsf{Cut}_{\mathcal{Z}}(\varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}})$$

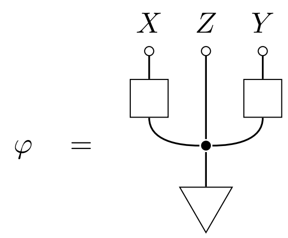
has no undirected path between any output in $\mathcal{X}$ *and any output in* $\mathcal{Y}$.

The string diagram $\varphi_{\mathcal{W}}$ is a shortcut for the marginal of φ on $\mathcal{W} \subseteq \mathsf{out}(\varphi)$, i.e. $\varphi_{\mathcal{W}} := \mathsf{norm}(\mathsf{del}_{\mathcal{W}^c} \circ \varphi)$.

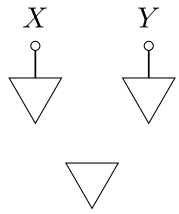
Intuitively, categorical d-separation can be also understood as topological disconnectedness of the string diagrams containing the output sets $\mathcal{X}$ and $\mathcal{Y}$.

Example 3.3. The following examples constitute the basic components of "classical" d-separation and illustrate the simplicity of categorical d-separation. In all cases, the unlabeled boxes denote distinct generators, i.e. distinct boxes in the generating hypergraph Σ.

(i) Fork: consider the morphism

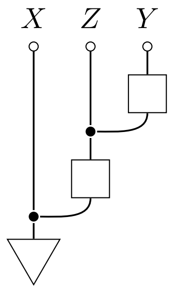


Then Z d-separates X from Y since

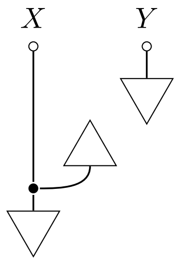
$\mathsf{Cut}_Z(\varphi) \quad =$ 

has disconnected X and Y.

(ii) Chain: consider the morphism

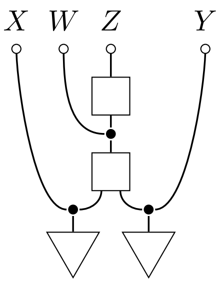
$\varphi \quad =$ 

Then Z d-separates X from Y since

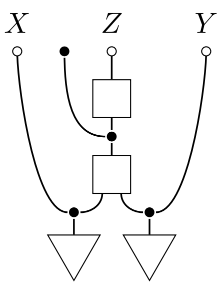
$\mathsf{Cut}_Z(\varphi) \quad =$ 

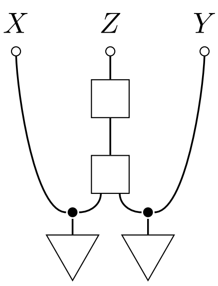
has disconnected X and Y.

(iii) Collider: consider the morphism

$\varphi \quad =$ 

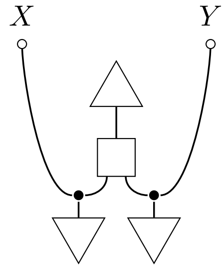
Then Z does *not* d-separate X from Y since we have

$\varphi_{X,Z,Y} \quad =$ 

$=$ 

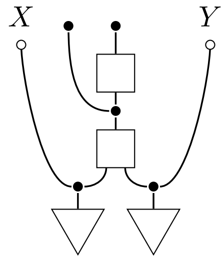
and therefore

$$\mathsf{Cut}_Z(\varphi_{X,Z,Y}) \quad =$$

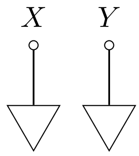


which still contains an undirected path $X - Y$. The same reasoning applies when $\mathcal{Z} = \{W\}$ or $\mathcal{Z} = \{W, Z\}$. However, if $\mathcal{Z} = \emptyset$, then

$$\varphi_{X,Y} \quad =$$



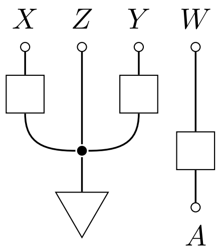
$$=$$



which disconnects X and Y. Therefore $\emptyset$ d-separates X and Y.

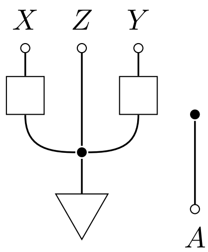
(iv) Consider the morphism

$$\varphi \quad =$$



The normalized marginal $\varphi_{X,Y,Z}$ is given by

$$\varphi_{X,Y,Z} \quad =$$



which again shows that Z d-separates X from Y since cutting Z makes X and Y disconnected. △

In the following, we will translate the notion of classical d-separation into the language of string diagrams and then show that both concepts, classical and categorical d-separation, coincide.

In order to define the classical notion of d-separation, we note that certain gs-monoidal string diagrams have an underlying DAG (see Section 2.1). A central ingredient in the classical formulation of d-separation is the notion of undirected paths on DAGs. However, translating these notions to string diagrams does not produce an undirected path in the sense of Definition 3.1.

For this reason, we use the term *DAG path* to refer to an undirected path in a DAG, i.e. to a sequence of wires connected by boxes from input to output or vice versa (but *not* from input to input or output to output). Moreover, we define the set of ancestor wires, given a set of wires, as

$$\mathsf{An}(\mathcal{X}) = \{U \in W(G) : \exists X \in \mathcal{X} \text{ such that } U \twoheadrightarrow X\}.$$

and the set of descendant wires as

$$\mathsf{Dec}(\mathcal{X}) = \{U \in W(G) : \exists X \in \mathcal{X} \text{ such that } X \twoheadrightarrow U\}.$$

Note that $\mathcal{X} \subseteq \mathsf{An}(\mathcal{X}), \mathsf{Dec}(\mathcal{X})$ since $X \twoheadrightarrow X$ holds by definition.

Since the definition of classical d-separation is traditionally given in the setting of DAGs (see for example [Pea09, Definition 1.2.3]), we now restrict to those gs-monoidal string diagrams that arise from an underlying DAG. In a causal structure represented by a DAG, it is (implicitly) assumed that every node or variable has its own causal mechanism associated with it; in our framework, this means that every box has exactly one output. Moreover, DAGs have no global inputs, which implies in our framework that $\mathsf{in}(\varphi) = \emptyset$. Finally, every variable in a DAG is usually assumed to be accessible.[2] This translates to string diagrams by assuming that every wire is connected to an output, i.e. the causal model is pure bloom.

Definition 3.4 (Classical d-separation)**.** *Let φ be a pure bloom causal model with $\mathsf{in}(\varphi) = \emptyset$ and such that every box has exactly one output. Then:*

(*a*) *A DAG path p in φ is called d-*separated *by a set of wires $\mathcal{Z} \subseteq \mathsf{out}(\varphi)$ if:*

  (*i*) *p contains a chain $W \to Z \to U$ or a fork $W \leftarrow Z \to U$ for some $Z \in \mathcal{Z}$.*

  (*ii*) *p contains a collider $W \to M \leftarrow U$ where $M \notin \mathsf{An}(\mathcal{Z})$.*

(*b*) *$\mathcal{X}$ is d-separated from $\mathcal{Y}$ by $\mathcal{Z}$ if every DAG path between every $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ is d-separated by $\mathcal{Z}$.*

We will now prove the equivalence of categorical d-separation with classical d-separation for the class of causal models on which the latter is defined. This implies that categorical d-separation is a suitable generalization of classical d-separation, which is simpler than the original one.

To this end, we first have to show a preparatory lemma.

Lemma 3.5. *Let φ be a pure bloom causal model, $b \in B(G)$ a box in φ, and $\mathcal{W} \subseteq \mathsf{out}(\varphi)$ a subset of its wires. The following statements are equivalent:*

(*i*) $\mathsf{out}(b) \cap \mathsf{An}(\mathcal{W}) = \emptyset$.

(*ii*) *b gets discarded in $\varphi_{\mathcal{W}} = \mathsf{norm}(\mathsf{del}_{\mathcal{W}^c} \circ \varphi)$.*

Proof. (*ii*) $\Longrightarrow$ (*i*): To prove the contrapositive, assume $\exists A \in \mathsf{out}(b)$ such that $A \in \mathsf{An}(\mathcal{W})$. Then there is a path $A \twoheadrightarrow W$ with $W \in \mathcal{W}$. Since W is still an overall output that does not get discarded, this path is still valid in $\mathsf{del}_{\mathcal{W}^c} \circ \varphi$. Therefore b remains in $\mathsf{norm}(\mathsf{del}_{\mathcal{W}^c} \circ \varphi)$.

[2] This holds, unless we consider DAGs with a specified set of latent variables. However, the classical d-separation criterion does not apply to this setting.

(i) $\implies$ *(ii)*: Consider the set $\mathsf{Dec}(\mathsf{out}(b))$. By assumption, we have that $\mathsf{Dec}(\mathsf{out}(b)) \cap \mathsf{An}(\mathcal{W}) = \emptyset$. We show that the box b gets discarded in $\mathsf{norm}(\mathsf{del}_{\mathsf{Dec}(\mathsf{out}(b))} \circ \varphi)$, which is enough because of $\mathcal{W}^c \supseteq \mathsf{Dec}(\mathsf{out}(b))$. By definition of $\mathsf{Dec}(\mathsf{out}(b))$, there exists a final box[3] $\hat{b}$ such that $\mathsf{out}(\hat{b}) \subseteq \mathsf{Dec}(\mathsf{out}(b))$. This shows that $\hat{b}$ gets discarded in $\mathsf{norm}(\mathsf{del}_{\mathsf{Dec}(\mathsf{out}(\hat{b}))} \circ \varphi)$.

Define $\widetilde{\varphi} := \mathsf{norm}(\mathsf{del}_{\mathsf{Dec}(\mathsf{out}(\hat{b}))} \circ \varphi)$. Repeating the above procedure with $\widetilde{\varphi}$, we arrive after a finite number of steps at b being a final box. Since it is then eliminable after composing with $\mathsf{del}_{\mathsf{Dec}(\mathsf{out}(b))}$, it no longer appears in the normalization. □

We will now show the equivalence between categorical d-separation and classical d-separation in the cases where φ represents a causal structure given by a DAG.

Proposition 3.6. *Both concepts of d-separation coincide on pure bloom causal models φ with $\mathsf{in}(\varphi) = \emptyset$ and in which every box has exactly one output.*

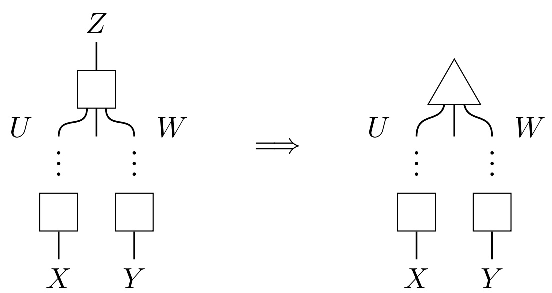
Proof. To make the proof more intuitive, we introduce the term *d-connected* as the negation of d-separated (in either version).

We start by showing that classical d-connectedness implies categorical d-connectedness. Let p be a DAG path between some $X \in \mathcal{X}$ and some $Y \in \mathcal{Y}$, which witnesses that $\mathcal{Z}$ makes $\mathcal{X}$ and $\mathcal{Y}$ be d-connected in the classical sense, which means that the following hold:

(i) For every chain $W \to M \to U$ or fork $W \leftarrow M \to U$ in p, we have $M \notin \mathcal{Z}$.

(ii) For every collider $W \to M \leftarrow U$ in p, we have $M \in \mathsf{An}(\mathcal{Z})$.

For simplicity, we also assume without loss of generality that p contains only one wire twice from $\mathcal{X}$ and $\mathcal{Y}$ each, say X and Y, respectively. Then this p can also be interpreted as an undirected path in φ, but generally not in $\varphi_{\mathsf{cut}} := \mathsf{Cut}_{\mathcal{Z}}(\varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}})$, since it may traverse wires that are not in φ_{cut}. However, we now show that there still is an undirected path p' between X and Y in φ_{cut}. By the above assumption (i), if p contains a wire $Z \in \mathcal{Z}$, then it has to arise from a collider $U \to Z \leftarrow W$ in p. Removing wire Z from p still defines a valid *undirected* path between X and Y, pictorially:



We prove that the path p' obtained by removing all wires in $\mathcal{Z}$ from p like this is an undirected path in φ_{cut}, which implies categorical d-connectedness. To this end, it only remains to show that each wire in p' is an existing wire in φ_{cut}, which we do as follows:

(i) X and Y themselves are still in φ_{cut}.

(ii) Every $Z \in \mathcal{Z}$ in p is part of a collider $U \to Z \leftarrow W$ as above, so that $U, W \in \mathsf{An}(\mathcal{Z})$. This implies that U and W survive in $\varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}}$ by Lemma 3.5.

[3]As defined in [FL22], a final box is one whose outputs are global outputs of φ without further copy or discard. Such a box always exists since φ is pure bloom and normalized (compare [FL22, Lemma 4.6]).

(iii) Since U and W are themselves either the middle node in a chain or fork or the start or end of p, we have $U, W \notin \mathcal{Z}$. This implies $U, W \in \mathsf{An}(\mathcal{Z}) \setminus \mathcal{Z}$, and therefore U and W survive also in φ_{cut}.

(iv) For every chain $W \to M \to U$ in p, if U survives in φ_{cut}, then so does M (since it survives in $\varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}}$ and $M \notin \mathcal{Z}$).

(v) For every fork $W \leftarrow M \to U$ in p, if U or W survives in φ_{cut}, then so does M (since it survives in $\varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}}$ and $M \notin \mathcal{Z}$).

Since the wires in p' are exactly those of p minus some of the colliders, we can start with the first two observations and then apply the latter two repeatedly on any segment bounded by colliders or the starting node X or the final node Y in order to conclude that all wires in p' are present in φ_{cut}. This concludes one direction of the proof.

The converse direction of showing that categorical d-connectedness implies classical d-connectedness works similarly. Let p be an undirected path between $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ in φ_{cut}. We assume without loss of generality that all wires in p are distinct. Furthermore, we also assume without loss of generality that p is of the form

$$X \twoheadleftarrow A - B \twoheadrightarrow Y, \tag{3.3}$$

where every wire that is in between A and B is not contained in $\mathsf{An}(\mathcal{X})$ or $\mathsf{An}(\mathcal{Y})$, or equivalently that every wire in p that is also in $\mathsf{An}(\mathcal{X})$ is directly reached from X by output-to-input traversals in p, and similarly for all wires in $\mathsf{An}(\mathcal{Y})$. This property can be achieved by taking every wire in p which is additionally in $\mathsf{An}(\mathcal{X})$ and replace it by the path from X to it by a sequence of output-to-input traversals, and similarly for every wire in $\mathsf{An}(\mathcal{Y})$. Note that this replacement may involve changing the starting and ending wires X and Y as well.

In order to turn p into a DAG path p' that witnesses classical d-separation, we need to remove all direct input-to-input traversals of a box in p; direct output-to-output traversals cannot occur due to the assumption that every box has exactly one output. We can hence simply add to p the unique output wire of every box that has an input-to-input traversal in p, and we obtain a valid DAG path p'.

It remains to verify the conditions on chains, forks and colliders. Clearly p' does not contain any chain $W \to Z \to U$ or fork $W \leftarrow Z \to U$ with $Z \in \mathcal{Z}$, since such a configuration cannot occur in p to begin with. For a collider $W \to M \leftarrow U$, the unique box which outputs M must be contained in $\varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}}$, and therefore be in $\mathsf{An}(\mathcal{X} \cup \mathcal{Y} \cup \mathcal{Z})$ by Lemma 3.5. However, M being in $\mathsf{An}(\mathcal{X})$ or $\mathsf{An}(\mathcal{Y})$ violates the assumption that p is of the form (3.3). Therefore M has to be in $\mathsf{An}(\mathcal{Z})$, showing the collider condition (ii). □

A necessary criterion for categorical d-separation is given by constraints on in- and outputs of the boxes in the causal model. We will record this in the following lemma. These constraints are the essential ingredient to prove conditional independence in Lemma 3.9.

Lemma 3.7. *Let φ be a pure bloom causal model and $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subseteq \mathsf{out}(\varphi)$ a partition of all output wires such that $\mathcal{Z}$ categorically d-separates $\mathcal{X}$ and $\mathcal{Y}$. Then every box $b \in B(G)$ in φ satisfies at least one of the cases:*

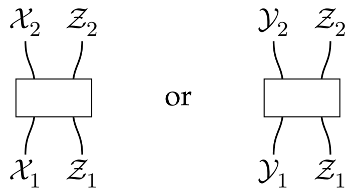
(i) $\mathsf{in}(b), \mathsf{out}(b) \subseteq \mathcal{X} \cup \mathcal{Z}$.

(ii) $\mathsf{in}(b), \mathsf{out}(b) \subseteq \mathcal{Y} \cup \mathcal{Z}$.

Proof. Suppose there exist $Y \in \mathcal{Y} \cap \mathsf{out}(b)$ and $X \in \mathcal{X} \cap \mathsf{out}(b)$. In that case, these wires are still in the output of b in φ_{cut}, and this contradicts the assumed disconnectedness of φ_{cut} with respect to $\mathcal{X}$ and $\mathcal{Y}$. Since $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ form a partition, this shows that either $\mathsf{out}(b) \subseteq \mathcal{X} \cup \mathcal{Z}$ or $\mathsf{out}(b) \subseteq \mathcal{Y} \cup \mathcal{Z}$.

Proving $\mathsf{out}(b) \subseteq \mathcal{X} \cup \mathcal{Z} \Rightarrow \mathsf{in}(b) \subseteq \mathcal{X} \cup \mathcal{Z}$ and $\mathsf{out}(b) \subseteq \mathcal{Y} \cup \mathcal{Z} \Rightarrow \mathsf{in}(b) \subseteq \mathcal{Y} \cup \mathcal{Z}$ works similarly, and this then proves the statement. □

Pictorially, Lemma 3.7 shows that if $\mathcal{Z}$ d-separates $\mathcal{X}$ and $\mathcal{Y}$, then every box b in φ is of the form



where $\mathcal{X}_i \subseteq \mathcal{X}, \mathcal{Y}_i \subseteq \mathcal{Y}, \mathcal{Z}_i \subseteq \mathcal{Z}$.

3.2 The d-separation criterion

In the following, we show that d-separation implies conditional independence for any generalized causal model. We first prove this result for a partition of wires in a pure bloom causal model in Lemma 3.9. We then refine it to any disjoint collection of wires in Corollary 3.11 in any generalized causal model. Finally, we show in Theorem 3.13 that d-separation fully characterizes causal compatibility for pure bloom causal models in all Markov categories with conditionals. Throughout, we also use the following convenient notation:

Notation 3.8. *Suppose a morphism f in $\mathbf{C}$ is compatible with a causal model φ in the sense of Definition 2.11. In that case, we refer to the wires of φ to indicate conditional independence instead of the objects in the tensor factorization of f. In other words, instead of writing $\mathcal{X}' \perp \mathcal{Y}' \mid \mathcal{Z}'$, we simply write $\mathcal{X} \perp \mathcal{Y} \mid \mathcal{Z}$.*

Here, each $W' = F(\mathsf{type}(W))$ is the object in $\mathbf{C}$ associated with the wire W by the causal model functor F (see Definition 2.11).

We will now show the soundness of d-separation under the additional conditions that φ is a pure bloom causal model and that the d-separation is among a tripartition of wires. As shown later in Corollary 3.11, this result remains true even without these assumptions.

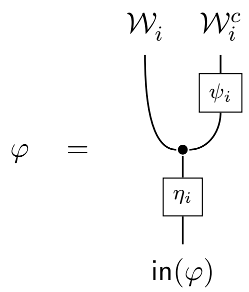
Lemma 3.9. *Let $\mathbf{C}$ be a strict Markov category with conditionals, and let φ be a pure bloom causal model. Further, let $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subseteq \mathsf{out}(\varphi)$ be a partition of wires in φ such that $\mathsf{in}(\varphi) \subseteq \mathcal{Y} \cup \mathcal{Z}$ and $\mathcal{X}$ and $\mathcal{Y}$ are d-separated by $\mathcal{Z}$.*

If a morphism f in $\mathbf{C}$ is compatible with φ, then $\mathcal{X} \perp \mathcal{Y} \mid \mathcal{Z}$ (as in Definition 1.16).

Proof. Choose a total ordering of all boxes $b_1, \ldots, b_{k-1} \in B(G)$ and a chain of sets of wires in $\mathsf{out}(\varphi)$,

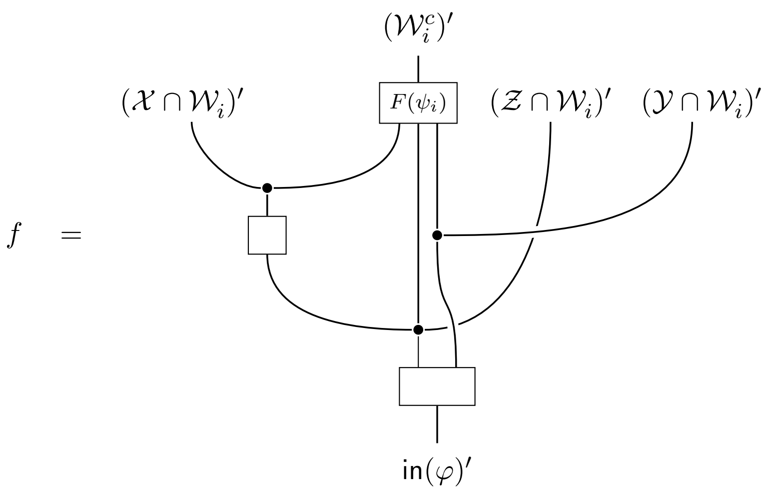
$$\mathsf{in}(\varphi) = \mathcal{W}_1 \subseteq \ldots \subseteq \mathcal{W}_k = \mathsf{out}(\varphi),$$

such that $\mathcal{W}_{i+1} = \mathsf{out}(b_i) \cup \mathcal{W}_i$ and $\mathsf{An}(\mathcal{W}_i) = \mathcal{W}_i$. Note that there is a factorization



in $\mathbf{FreeMarkov}_\Sigma$, where η_i is again a pure bloom and η_1 is an identity morphism. The existence of such a chain of sets follows easily by induction on the number of boxes based on the existence of a final box.[4]

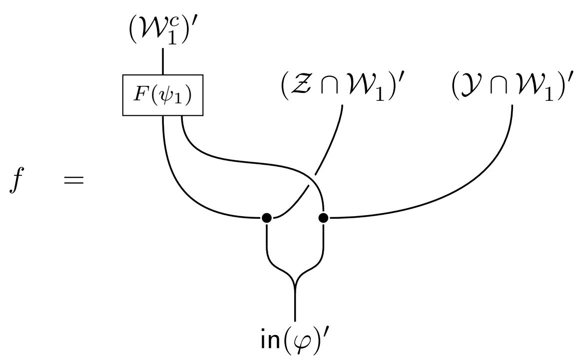
Then for every $i \in \{1, \ldots, k\}$, we show the existence of a decomposition



(3.4)

Since $\mathcal{W}_n^c = \emptyset$, setting $i = k$ proves the desired statement.

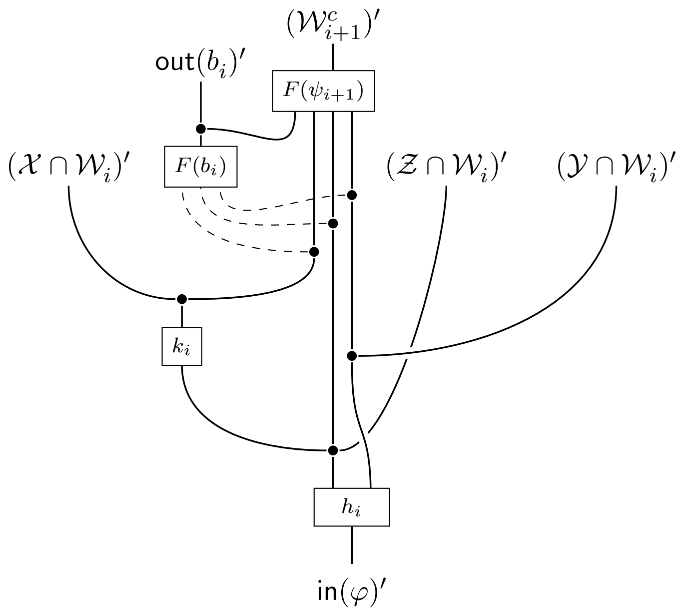
We prove this stronger claim by induction on i. The start of the induction at $i = 1$ is trivial since η_1 is the identity and therefore



[4] Note that restricting to causal models arising from DAGs, this statement amounts to the standard fact that every DAG gives rise to a topological ordering where $a \preccurlyeq b$ if $a \in \mathsf{An}(\{b\})$ (see for example [TM11, Section 5.7]). The maximal element of this ordering then corresponds to the final box in the string diagram.

since $\mathcal{Y} \cap \mathcal{W}_1 = \mathcal{Y} \cap \mathsf{in}(\varphi) = \emptyset$. For the induction step, we prove the statement at $i+1$. Since φ is pure bloom and $\mathsf{An}(\mathcal{W}_{i+1}) = \mathcal{W}_{i+1}$, we can peel off the box b_i with $\mathcal{W}_{i+1} = \mathsf{out}(b_i) \cup \mathcal{W}_i$ from ψ_i, so as to achieve the decomposition

$f \quad =$

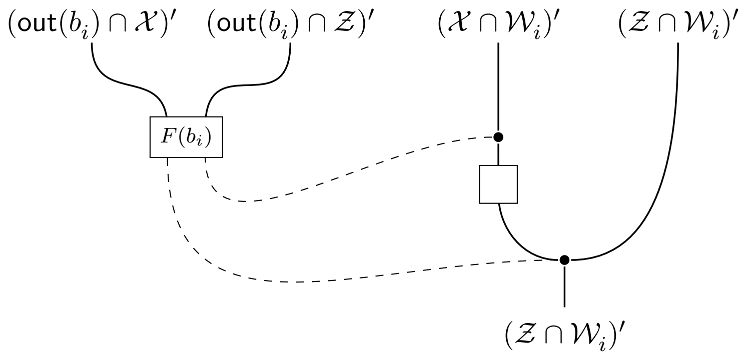


(3.5)

where we have used the induction assumption to obtain a decomposition as in the lower half, and the dashed wires indicate that only some of them may be present since the inputs of b_i are an unspecified subset of $\mathcal{W}_i$. By Lemma 3.7, we have to distinguish two cases:

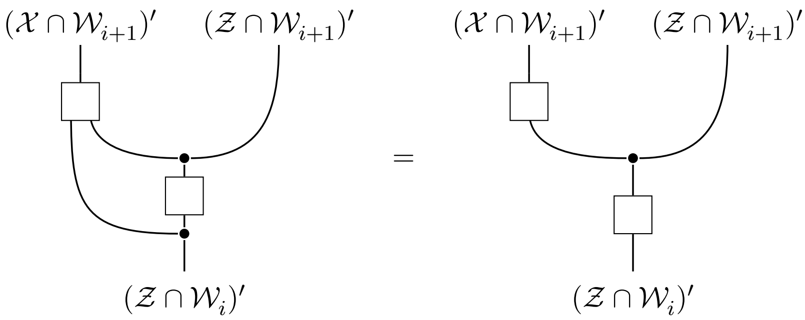
(i) $\mathsf{in}(b_i), \mathsf{out}(b_i) \subseteq \mathcal{X} \cup \mathcal{Z}$. Then, the third dashed wire in the above decomposition of f is not needed, and we consider the morphism

$g \quad :=$



which is part of that decomposition. By the existence of conditionals, we can rewrite g in the form

$g \quad =$



where both lower boxes can be refined with an internal structure consisting of carrying $(\mathcal{Z} \cap \mathcal{W}_i)'$ forward on a separate wire, but this internal structure is not relevant for the remainder of the proof. Substituting this form of g into Equation (3.5), i.e. replacing the morphism k_i there with the left box here and merging the lower box here with h_i there, proves the induction step.

(ii) $\mathsf{in}(b_i), \mathsf{out}(b_i) \subseteq \mathcal{Y} \cup \mathcal{Z}$. Then, the first dashed wire in the above decomposition of f is not needed, and we can merge $F(b_i)$ with h_i, which shows the statement. □

We will now generalize Lemma 3.9 to all generalized causal models and to arbitrary disjoint sets $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ which do not necessarily partition the set of all wires.

Lemma 3.10. *Let φ be a generalized causal model and $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z} \subseteq \mathsf{out}(\varphi)$ a tripartition of output wires in φ such that $\mathsf{in}(\varphi) \subseteq \mathcal{Y} \cup \mathcal{Z}$ and such that $\mathcal{Z}$ categorically d-separates $\mathcal{X}$ and $\mathcal{Y}$. Then there exists a tripartition of wires $\widetilde{\mathcal{X}} \supseteq \mathcal{X}$, $\widetilde{\mathcal{Y}} \supseteq \mathcal{Y}$, $\mathcal{Z}$ in the pure bloom version $\varphi_{\text{pure-bloom}}$ of φ*[5] *such that*

$$\mathcal{Z} \text{ d-separates } \widetilde{\mathcal{X}} \text{ and } \widetilde{\mathcal{Y}} \qquad \text{in } \varphi_{\text{pure-bloom}}$$

Proof. With $\varphi_{\text{cut}} := \mathsf{Cut}_{\mathcal{Z}}(\varphi_{\text{pure-bloom}})$, define

$$\widetilde{\mathcal{X}} := \{U \in \mathsf{out}(\varphi_{\text{cut}}) : \exists X \in \mathcal{X} : X - U \text{ in } \varphi_{\text{cut}}\} \supseteq \mathcal{X}$$

to be the connected component of $\mathcal{X}$ in φ_{cut}, and

$$\widetilde{\mathcal{Y}} := \mathsf{out}(\varphi_{\text{pure-bloom}}) \setminus \left(\widetilde{\mathcal{X}} \cup \mathcal{Z}\right) \supseteq \mathcal{Y}.$$

By definition, $\widetilde{\mathcal{X}}, \widetilde{\mathcal{Y}}, \mathcal{Z}$ form a tripartition of wires in $\varphi_{\text{pure-bloom}}$. Moreover, $\widetilde{\mathcal{X}}$ and $\widetilde{\mathcal{Y}}$ are categorically d-separated by $\mathcal{Z}$ since any path in $\mathsf{Cut}_{\mathcal{Z}}(\varphi_{\text{pure-bloom}})$ is valid in $\mathsf{Cut}_{\mathcal{Z}}(\varphi)$ and vice versa. □

Corollary 3.11. *Let $\mathbf{C}$ be a strict Markov category with conditionals, and let φ be a generalized causal model.*[6] *Further, let $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subseteq \mathsf{out}(\varphi)$ be disjoint sets of output wires in φ such that $\mathsf{in}(\varphi) \subseteq \mathcal{Y} \cup \mathcal{Z}$ and $\mathcal{X}$ and $\mathcal{Y}$ are d-separated by $\mathcal{Z}$. If f is compatible with φ, then $\mathcal{X} \perp \mathcal{Y} \mid \mathcal{Z}$.*

In this statement, we use another standard convention: when the disjoint sets $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ do not partition the set of wires of φ, then the conditional independence $\mathcal{X} \perp \mathcal{Y} \mid \mathcal{Z}$ is to be understood as Definition 1.16 applied to the corresponding marginal $f_{\mathcal{X}', \mathcal{Y}', \mathcal{Z}'}$ rather than to f itself.

Proof. We prove this statement by reducing it to the case of pure bloom causal models treated in Lemma 3.9.

Consider the restricted causal model $\psi := \varphi_{\mathcal{X},\mathcal{Y},\mathcal{Z}}$ and its compatible morphism $g := F(\psi)$ obtained from the compatibility of f with φ, which is a marginal of f. By the definition of categorical d-separation, $\mathcal{Z}$ d-separates $\mathcal{X}$ and $\mathcal{Y}$ also in ψ. Let $\psi_{\text{pure-bloom}}$ be the pure bloom version of ψ. Since g is compatible with ψ, we can extend g to a pure bloom version

$$g_{\text{pure-bloom}} := F(\psi_{\text{pure-bloom}})$$

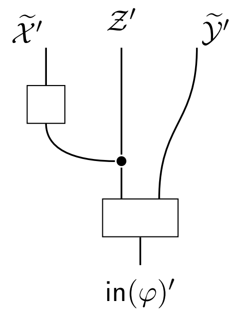
[5]The pure-bloom version $\varphi_{\text{pure-bloom}}$ is obtained by copying each wire so to make it into an output. It is part of the bloom-circuitry factorization of [FL22].

[6]In this situation φ does not need to be pure bloom.

of which g is a marginal.

By Lemma 3.10, for $\psi_{\text{pure-bloom}}$ there is a tripartition of output wires $\widetilde{\mathcal{X}} \supseteq \mathcal{X}, \widetilde{\mathcal{Y}} \supseteq \mathcal{Y}, \mathcal{Z}$ such that $\mathcal{Z}$ d-separates $\widetilde{\mathcal{X}}$ and $\widetilde{\mathcal{Y}}$. Since $\mathbf{C}$ has conditionals, Lemma 3.9 provides us with a decomposition of the form

$g_{\text{pure-bloom}} \quad =$



By marginalizing over $\widetilde{\mathcal{X}}' \setminus \mathcal{X}', \widetilde{\mathcal{Y}}' \setminus \mathcal{Y}'$ in $g_{\text{pure-bloom}}$, we obtain the desired conditional independence for the marginal $f_{\mathcal{X}',\mathcal{Y}',\mathcal{Z}'}$. □

Note that this result includes the soundness of the classical d-separation criterion in the classical case of discrete random variables in Bayesian networks.[7] This special case is obtained when restricting to pure bloom causal models with $\mathsf{in}(\varphi) = \emptyset$ and single output boxes (i.e. the setting of Proposition 3.6), as well as restricting to the Markov category **FinStoch**. In this case, conditional independence reduces to Definition 1.15 by Remark 1.17.

Following conventions in the study of Bayesian networks, Corollary 3.11 states that every probability distribution compatible with a causal structure satisfies a certain type of *Markov property*. In the following, we introduce the *local* and *global Markov property* in the context of generalized causal models. These allow for a simple formulation of the d-separation criterion. The original definitions in the language of DAGs go back to [KSC84] and [Lau96].

Definition 3.12. *Let φ be a generalized causal model and f a morphism in a strict Markov category $\mathbf{C}$. Then we say that f satisfies:*

(*i*) *the* global Markov property *with respect to φ if for every three disjoint sets of outputs $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subseteq \mathsf{out}(\varphi)$ with $\mathsf{in}(\varphi) \subseteq \mathcal{Y} \cup \mathcal{Z}$:*

$$\mathcal{X} \text{ and } \mathcal{Y} \text{ are categorically } d\text{-separated by } \mathcal{Z} \text{ in } \varphi \quad \Longrightarrow \quad \mathcal{X} \perp \mathcal{Y} \mid \mathcal{Z} \text{ in } f.$$

(*ii*) *the* local Markov property *with respect to φ if for every box b in φ, we have*

$$\mathsf{out}(b) \perp \mathsf{Dec}(\mathsf{out}(b))^c \setminus \mathsf{in}(b) \mid \mathsf{in}(b) \text{ in } f.$$

The local Markov property states that every output of a box can be sampled independently from all non-descendants of the box only by accessing the information of the input wires of the box. In the following, we show that although the local Markov property is weaker than the global Markov property, it suffices for compatibility of a morphism with a generalized causal model.

Theorem 3.13 (The categorical d-separation criterion)**.** *Suppose we are given the following:*

[7] See [VP90] for the original proof and [Pea09, Theorem 1.2.5(i)] for a textbook account.

▷ *$\mathbf{C}$ is a strict Markov category with conditionals.*

▷ *φ is a pure bloom causal model over a hypergraph Σ such that each box in φ has a distinct type.*

▷ *$f : \bigotimes_{i=1}^{n} W'_i \to \bigotimes_{j=1}^{m} V'_j$ is a morphism in $\mathbf{C}$.*

Then the following statements are equivalent:

(*i*) *f is compatible with the causal model φ.*

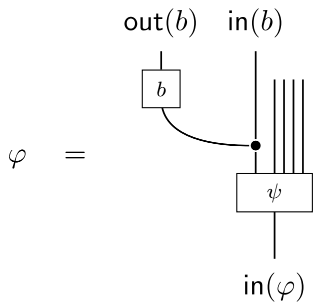
(*ii*) *f satisfies the global Markov property.*

(*iii*) *f satisfies the local Markov property.*

Proof. (*i*) $\Longrightarrow$ (*ii*): The global Markov property is precisely the statement of Corollary 3.11.

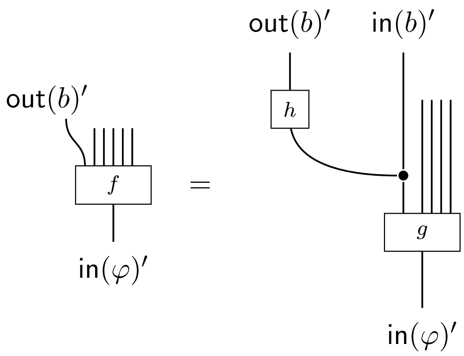
(*ii*) $\Longrightarrow$ (*iii*): This follows from the fact that $\mathsf{Dec}(\mathsf{out}(b))^c$ and $\mathsf{out}(b)$ are d-separated by $\mathsf{in}(b)$, and $\mathsf{in}(\varphi) \subseteq \mathsf{Dec}(\mathsf{out}(b))^c$, which makes the global Markov property specialize to the local one.

(*iii*) $\Longrightarrow$ (*i*): We prove this statement by induction over the number of boxes $k := |B(G)|$. The case $k = 1$ is trivial. For the step from k to $k+1$, let b be a final box in φ, which means that $\mathsf{Dec}(\mathsf{out}(b)) = \mathsf{out}(b)$. Then, φ factorizes as



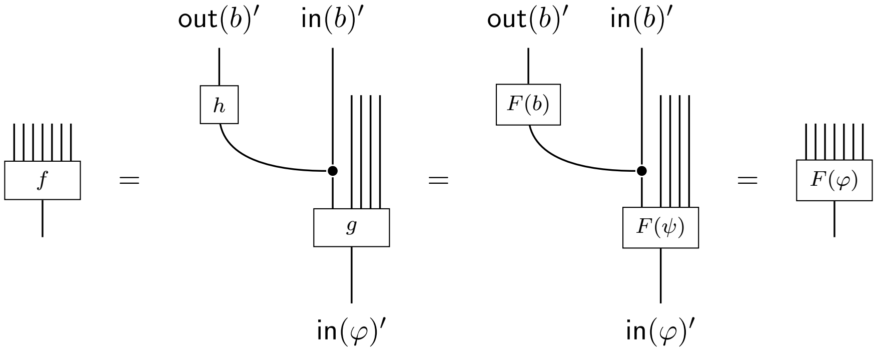
where ψ is another causal model satisfying all of our assumptions, and no box in ψ has the same type in Σ as b does.

In order to construct a functor F as in Definition 2.11, note first that it must satisfy (2.14), which already lets us write the domain of f as $\mathsf{in}(\varphi)'$, and similarly for the codomain. Since f satisfies the local Markov property with respect to b, we can decompose f by Definition 1.16 as



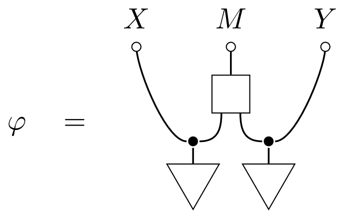
(3.6)

By the induction hypothesis, we have that g is compatible with ψ since g satisfies the local Markov properties specified by ψ. Since box b appears only once in φ, we can freely define the action of the functor F on b as $F(b) := h$. Then, we obtain

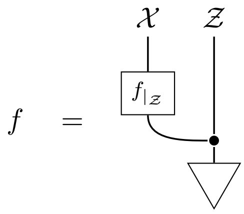


where we use in the first step Equation (3.6) and in the last the fact that F is a Markov functor. □

Remark 3.14. (i) Note that we have used the assumption that **C** has conditionals only for the implication *(i)* $\implies$ *(ii)*. Therefore, for an arbitrary strict Markov category, the global as well as the local Markov property is a sufficient condition for the compatibility of a morphism with a generalized causal model (satisfying our assumptions). However, these Markov properties implicitly require the existence of certain conditionals. Consider for example the generalized causal model



where all boxes are of distinct types. Choosing $\mathcal{X} = \{M\}, \mathcal{Y} = \emptyset$ and $\mathcal{Z} = \{X, Y\}$, a morphism f satisfying the global Markov property displays in particular the conditional independence $\{M\} \perp \emptyset \mid \mathcal{Z}$, pictorially:



This shows that the conditional $f_{|\mathcal{Z}}$ exists, and this recovers the box that outputs M (up to almost sure equality).

(ii) Theorem 3.13 shows that d-separation correctly detects causal compatibility for the Markov categories **FinStoch**, **Gauss**, or **BorelStoch**. For the Markov category **Stoch**, which does not have conditionals, the global and local Markov properties are at least sufficient for compatibility since our proof of these implications has not used conditionals.

(iii) Note that Theorem 3.13 only applies to causal models where each box appears at most once in the model (which implies that φ has no nontrivial symmetries). However, the implication *(i)* $\implies$ *(ii)* applies to arbitrary generalized causal models as proven in Corollary 3.11. △

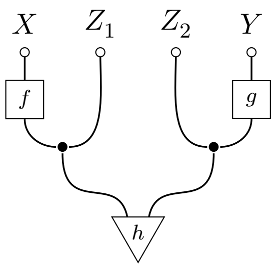
While the proof requires that φ is a pure bloom causal model, it is an open question whether the d-separation criterion extends to symmetric causal models (i.e. not every box has a distinct type in Σ)

by adding an appropriate symmetry constraint to the Markov properties.

Question 3.15. *Can Theorem 3.13 be extended to more general causal models? In particular, what about allowing the same box to appear several times in φ?*

Example 3.16. We now present two examples that go beyond the classical d-separation criterion. In (i) we will study a causal structure that does not arise from a DAG, while in (ii) we study a DAG causal structure with continuous variables.

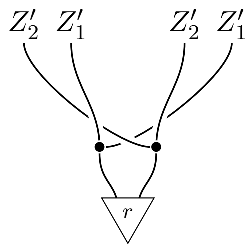
(i) Let φ be the causal structure



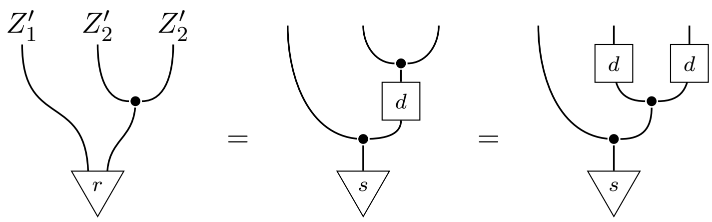
and let **C** be a strict Markov category with conditionals. By Theorem 3.13, a morphism $t : I \to X' \otimes Z'_1 \otimes Z'_2 \otimes Y'$ in **C** is compatible with this structure if and only if it satisfies

$$X \perp \{Y, Z_2\} \mid Z_1 \quad \text{and} \quad Y \perp \{X, Z_1\} \mid Z_2$$

For a general class of examples, consider $X' = Z'_2$ and $Y' = Z'_1$ and any morphism in **C** of the form



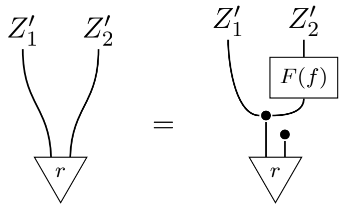
We claim that such a distribution is compatible with φ if and only if there exist morphisms d and d' such that



(3.7)

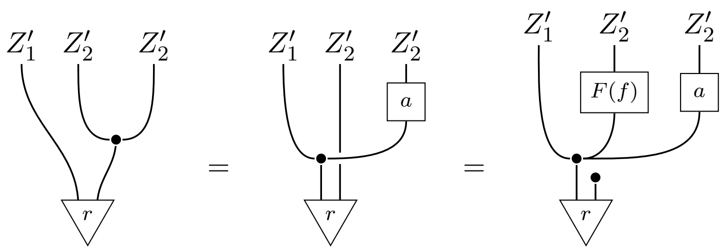
where s is the first marginal of r, and similarly $d' : Z'_2 \to Z'_1$ satisfies the same equations the other way around. Here, the second equation states that the morphism d is s-a.s. deterministic [Fri20, Definition 13.11], and similarly for d'.

Indeed, assuming compatibility we have that



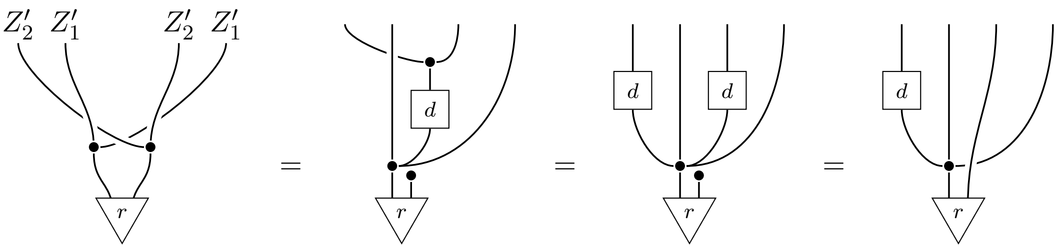
(3.8)

which shows the first equality in Equation (3.7). For the second equality, we have that



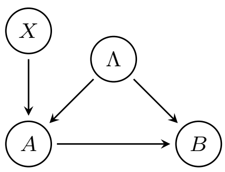
where we have used the conditional independence $Z_2 \perp X \mid Z_1$ in the first step and Equation (3.8) in the second step. Since the morphism is symmetric with respect to permutations of the output wires X and Z_2, we have $a = F(f)$ s-a.s. which shows the second equality in Equation (3.7). Proving the existence of d' works analogously by interchanging the roles of X and Y as well as Z_1 and Z_2.

Conversely, we have

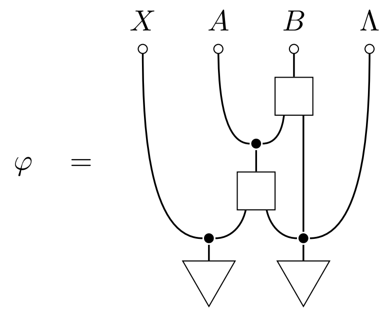


where we have used the assumption that d is s-a.s. deterministic in the second equation. Repeating this calculation, interchanging the roles of Z_1 and Z_2 as well as X and Y, shows the statement.

(ii) Consider the instrumental scenario given by the DAG



This has been previously studied mainly in the context of DAGs with latent variables [Pea95; Bon01]. For our analysis, we assume each variable to be observed, which means that the causal structure reads string-diagrammatically as



(3.9)

There are two non-trivial d-separations:

(a) Between X and B by $\{A, \Lambda\}$,

(b) Between X and Λ.

Therefore, Theorem 3.13 implies that a distribution P on a four-fold tensor product object in a Markov category with conditionals is compatible with φ if and only if $X \perp B \mid A, \Lambda$ and $X \perp \Lambda$. In **BorelStoch**, this means that P is compatible with φ if and only if

$$\begin{aligned} &P(X \in E_1, A \in E_2, B \in E_3, \Lambda \in E_4) \\ &= \int_{E_2} \int_{E_4} P_{X|A,\Lambda}(X \in E_1|a, \lambda)\, P_{B|A,\Lambda}(B \in E_3|a, \lambda)\, P_{A,\Lambda}(\mathrm{d}a, \mathrm{d}\lambda) \end{aligned}$$

and

$$P(X \in E_1, \Lambda \in E_4) = P(X \in E_1) \cdot P(\Lambda \in E_4)$$

where E_i are measurable sets in the Borel σ-algebras of the spaces X', A', B' and Λ'.

For simplicity, assume that all random variables take values in $\mathbb{R}$ and are absolutely continuous, i.e. there exists a density $f : X' \times A' \times B \times \Lambda' \to [0, \infty)$ such that

$$P(X \in E_1, A \in E_2, B \in E_3, \Lambda \in E_4) = \int_{E_1 \times E_2 \times E_3 \times E_4} f(x, a, b, \lambda)\, \mathrm{d}x\, \mathrm{d}a\, \mathrm{d}b\, \mathrm{d}\lambda$$

The causal compatibility now amounts to the following two conditions:

(a) $X \perp \Lambda$, i.e.

$$f_{X,\Lambda}(x, \lambda) = f_X(x) \cdot f_\Lambda(\lambda) \quad \text{a.e.} \tag{3.10}$$

where a.e. means almost everywhere with respect to the Lebesgue measure on $\mathbb{R}$.

(b) $X \perp B \mid A, \Lambda$, i.e.

$$f(x, a, b, \lambda) = f_{X|A,\Lambda}(x, a, \lambda) \cdot f_{A,\Lambda}(a, \lambda) \cdot f_{B|A,\Lambda}(b, a, \lambda) \quad \text{a.e.,} \tag{3.11}$$

where the conditional densities are defined implicitly by

$$f_{A|X\Lambda}(a|x, \lambda) \cdot f_{X,\Lambda}(x, \lambda) = f_{X,A,\Lambda}(x, a, \lambda) \quad \text{a.e.}$$

Combining Eq. (3.10) and Eq. (3.11) results in

$$f(x, a, b, \lambda) = f_\Lambda(\lambda) \cdot f_X(x) \cdot f_{A|X\Lambda}(a, x, \lambda) \cdot f_{B|A,\Lambda}(b, a, \lambda) \quad \text{a.e.}$$

which is the usual factorization condition for compatibility with the causal structure in (3.9). △

Chapter 4

Conclusion and Outlook

In this thesis, we have studied the causal compatibility problem in the language of categorical probability. We have reviewed the concept of Markov categories as an abstract model of probability independent of the usual measure-theoretic axiomatization. Further, we have used the framework of Markov categories as a toolbox to describe and study causal models using free Markov categories.

In Chapter 1, we have reviewed the key concepts of categorical probability, building on the framework of Markov categories (Definition 1.1). We have introduced Markov categories and studied the construction of several concrete examples like finite probability distributions (Example 1.3), Gaussian probabilities (Section 1.2.3), Borel probability distributions (Section 1.2.2), and general measure-theoretic probabilities (Section 1.2.1). Further, we have reviewed the existence of conditionals (Section 1.3) and conditional independence (Section 1.4) in Markov kernels which are central in deciding compatibility via the d-separation criterion.

In Chapter 2, we introduce the categorical framework to study causal models as string diagrams. We start by reviewing the category of hypergraphs (Section 2.2) which is the basic toolbox for building up string diagrams. Later, we review the construction of free gs-monoidal and free Markov categories (Section 2.3 and Section 2.4), which arise as hypergraphs with certain properties. Finally, we define generalized causal models (Section 2.5) as morphisms in free Markov categories, which are gs-monoidal string diagrams.

In Chapter 3, we study the causal compatibility problem for generalized causal models by investigating the d-separation criterion. We show that the d-separation criterion fully characterizes causal compatibility for Markov categories with conditionals (Theorem 3.13) and pure bloom causal models (i.e. causal models with access to all random variables). We introduce a categorical version of d-separation (Definition 3.2), which arises from topological disconnectedness in string diagrams. This notion constitutes a meaningful generalization of classical d-separation on DAGs (Definition 3.4). Moreover, the correspondence to disconnectedness makes categorical d-separation more intuitive than its classical counterpart.

This abstract treatment of causal compatibility with the d-separation leads to several open questions:

- ▷ The framework of generalized causal models allows for symmetric versions of causal models (i.e. models that contain multiple identical processes, see Example 2.8 (iii)). However, the d-separation criterion only decides compatibility for causal models without explicit symmetry. It is an open question whether this can be generalized to symmetric models (see also Question 3.15).
- ▷ In contrast to DAGs, generalized causal models also manifestly include causal models with latent variables. While the soundness of the d-separation criterion holds for arbitrary causal

models (including models with latent variables), having access to all variables (i.e. restricting to pure bloom causal models) is essential for proving the completeness of the d-separation criterion. The inflation technique [WSF19; NW20] constitutes a powerful method to fully decide causal compatibility with a given DAG model, including even latent variables. More specifically, [WSF19] shows the soundness and [NW20] the completeness of the criterion. However, the inflation technique is so far only developed for discrete random variables. We expect that it naturally extends to generalized causal models and morphisms in certain Markov categories.

▷ Since we have a purely abstract result of causal compatibility on Markov categories, a natural question is whether this result leads to interesting and meaningful statements for other concrete Markov categories going beyond probability theory, for example, certain hypergraph categories (see [Fri20, Section 8] and references therein).

Concluding, string diagrams constitute a new approach for causal models which generalizes DAGs in several directions. In addition, d-separation attains a much simpler description in the language of string diagrams than in the traditional DAG setting. This is not completely surprising; string diagrams in Markov categories model information flow, which motivates the link between disconnectedness of string diagrams and conditional independence of compatible stochastic maps.

These observations suggest that string diagrams might be a more natural framework for causal models than DAGs.

Bibliography

[Bog07] V. I. Bogachev. *Measure Theory: Volume I*. Springer (2007) (pp. 10, 62).

[Bon+16] F. Bonchi, F. Gadducci, A. Kissinger, P. Sobociński, and F. Zanasi. *Rewriting modulo symmetric monoidal structure*. Proc. LICS Symposium (2016). DOI: 10.1145/2933575.2935316 (p. 23).

[Bon01] B. Bonet. *Instrumentality Tests Revisited*. Proc. Conf. on Uncertainty Artif. Intell. (2001). DOI: 10.48550/arXiv.1301.2258 (p. 52).

[CJ19] K. Cho and B. Jacobs. *Disintegration and Bayesian inversion via string diagrams*. Math. Struct. Comput. Sci. **29** (2019). DOI: 10.1017/S0960129518000488 (pp. 3, 5, 15, 16).

[CS11] B. Coecke and R. W. Spekkens. *Picturing classical and quantum Bayesian inference*. Synth. **186** (2011). DOI: 10.1007/s11229-011-9917-5 (p. 5).

[CG99] A. Corradini and F. Gadducci. *An Algebraic Presentation of Term Graphs, via GS-Monoidal Categories*. Appl. Categ. Struct. **7** (1999). DOI: 10.1023/A:1008647417502 (p. 5).

[Fon13] B. Fong. *Causal Theories: A Categorical Perspective on Bayesian Networks*. Master's thesis, University of Oxford (2013). arXiv: 1301.6201 (pp. 5, 20, 21, 33).

[FS18a] B. Fong and D. I. Spivak. *Hypergraph Categories*. J. Pure Appl. Algebra **223** (2018). DOI: 10.1016/j.jpaa.2019.02.014 (p. 25).

[FS18b] B. Fong and D. I. Spivak. *Seven Sketches in Compositionality: An Invitation to Applied Category Theory*. Cambridge University Press (2018). DOI: 10.1017/9781108668804 (p. 3).

[For21] P. Forré. *Transitional Conditional Independence* (2021). arXiv: 2104.11547 (p. 16).

[Fri20] T. Fritz. *A synthetic approach to Markov kernels, conditional independence and theorems on sufficient statistics*. Adv. Math. **370** (2020). DOI: 10.1016/j.aim.2020.107239 (pp. 1, 3, 5, 9, 13, 15, 16, 32, 51, 56).

[FGP21] T. Fritz, T. Gonda, and P. Perrone. *De Finetti's Theorem in Categorical Probability*. J. Stoch. Anal. **2** (2021). DOI: 10.31390/josa.2.4.06 (p. 1).

[Fri+20] T. Fritz, T. Gonda, P. Perrone, and E. F. Rischel. *Representable Markov Categories and Comparison of Statistical Experiments in Categorical Probability* (2020). arXiv: 2010.07416 (p. 1).

[FK22] T. Fritz and A. Klingler. *The d-separation criterion in Categorical Probability* (2022). arXiv: 2207.05740 (pp. iii, 3, 19, 37).

[FL22] T. Fritz and W. Liang. *Free gs-monoidal categories and free Markov categories* (2022). arXiv: 2204.02284 (pp. 7, 23, 27, 29, 30, 38, 42, 47).

[FP20] T. Fritz and P. Perrone. *Monads, partial evaluations, and rewriting*. Electron. Notes Theor. Comput. Sci. **352** (2020). DOI: 10.1016/j.entcs.2020.09.007 (p. 9).

[FR19] T. Fritz and E. F. Rischel. *Infinite products and zero-one laws in categorical probability*. Compositionality **2** (2019). DOI: 10.32408/compositionality-2-3 (p. 1).

[Gad96] F. Gadducci. *On The Algebraic Approach To Concurrent Term Rewriting*. PhD thesis, University of Pisa (1996). URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.57.5651 (p. 5).

[Gao22] D. Gao. *Functorial Causal Models: Towards an Algebraic Approach to Causal Inference*. Bachelor's thesis, Brown University (2022). URL: https://www.brown.edu/academics/math/sites/math/files/Gao%2C%20Dichuan_0.pdf (p. 20).

[Gir82] M. Giry. *A categorical approach to probability theory*. Categorical aspects of topology and analysis. Springer, Berlin, Heidelberg (1982). DOI: 10.1007/BFb0092872 (p. 9).

[Gol99] P. V. Golubtsov. *Axiomatic description of categories of information transformers*. Prob. Inf. Transm. **35** (1999) (p. 5).

[JKZ19] B. Jacobs, A. Kissinger, and F. Zanasi. *Causal Inference by String Diagram Surgery*. Lect. Notes Comput. Sci. **11425** (2019). DOI: 10.1007/978-3-030-17127-8_18 (p. 20).

[KSC84] H. Kiiveri, T. P. Speed, and J. B. Carlin. *Recursive Causal Models*. J. Aust. Math. Soc. **36** (1984). DOI: 10.1017/S1446788700027312 (p. 48).

[KB18] A. Kolmogorov and A. Bharucha-Reid. *Foundations of the theory of probability: Second English Edition*. Courier Dover Publications (2018) (p. 3).

[Lau96] S. L. Lauritzen. *Graphical Models*. Oxford University Press (1996) (p. 48).

[Lei14] T. Leinster. *Basic category theory*. Cambridge University Press (2014) (pp. 3, 24).

[Mac78] S. Mac Lane. *Categories for the Working Mathematician*. Springer Science & Business Media (1978). DOI: 10.1007/978-1-4757-4721-8 (p. 3).

[MZ22] A. Milosavljevic and F. Zanasi. *String Diagram Rewriting Modulo Commutative Monoid Structure* (2022). arXiv: 2204.04274 (p. 23).

[NW20] M. Navascués and E. Wolfe. *The Inflation Technique Completely Solves the Causal Compatibility Problem*. J. Causal Inference (2020). DOI: 10.1515/jci-2017-0020 (p. 56).

[Par67] K. Parthasarathy. *Probability measures on metric spaces*. Academic Press, New York and London (1967) (p. 13).

[Pea09] J. Pearl. *Causality: Models, reasoning, and inference*. Cambridge University Press (2009). Second edition. DOI: 10.1017/CBO9780511803161 (pp. 35, 41, 48).

[Pea95] J. Pearl. *On the Testability of Causal Models with Latent and Instrumental Variables*. Proc. Conf. on Uncertainty Artif. Intell. (1995). URL: https://dl.acm.org/doi/10.5555/2074158.2074208 (p. 52).

[Per19] P. Perrone. *Notes on Category Theory with examples from basic mathematics*. Lecture notes (2019). arXiv: 1912.10642 (pp. 3, 7, 8, 24).

[Ris20] E. F. Rischel. *The Category Theory of Causal Models*. Master's thesis, University of Copenhagen (2020). URL: https://www.erischel.com/documents/mscthesis.pdf (p. 20).

[RW21] E. F. Rischel and S. Weichwald. *Compositional abstraction error and a category of causal models*. Proc. Conf. on Uncertainty Artif. Intell. **161** (2021). arXiv: 2103.15758 (p. 20).

[Sea12] G. J. Seal. *Tensors, monads and actions*. Theory Appl. Categories **28** (2012). arXiv: 1205.0101 (p. 8).

[Stu06] M. Studený. *Probabilistic Conditional Independence Structures*. Springer Science & Business Media (2006). DOI: 10.1007/b138557 (p. 16).

[TM11] K. Thulasiraman and N. S. Madisetti. *Graphs: Theory and Algorithms*. John Wiley & Sons, 2011 (p. 45).

[VP90] T. Verma and J. Pearl. *Causal Networks: Semantics and Expressiveness*. Machine intelligence and pattern recognition **9** (1990). DOI: 10.1016/B978-0-444-88650-7.50011-1 (p. 48).

[WSF19] E. Wolfe, R. W. Spekkens, and T. Fritz. *The Inflation Technique for Causal Inference with Latent Variables*. J. Causal Inference **5** (2019). DOI: 10.1515/jci-2017-0020 (p. 56).

Appendix A

Construction of the Giry monad

In this appendix, we give a detailed construction of the Giry monad, which has been used to introduce the Markov categories **Stoch** and **BorelStoch** in Section 1.2.

Let **Meas** be the category of measurable spaces, i.e. objects being measurable spaces (X, Σ_X) (Definition 1.10) and morphisms $f : X \to Y$ being measurable maps (Definition 1.11).

To construct the Giry monad we start by defining the functor[1]

$$\mathcal{P} : \mathbf{Meas} \to \mathbf{Meas} \qquad \text{with} \qquad \mathcal{P}(X, \Sigma_X) = \Big(P(X, \Sigma_X), \Sigma_{\mathcal{P}X}\Big)$$

where

$$P(X, \Sigma_X) := \{p : \Sigma_X \to [0, 1] : p \text{ probability measure}\}$$

and $\Sigma_{\mathcal{P}X}$ is the smallest σ-algebra on $P(X, \Sigma_X)$ that makes the evaluation map

$$\mathsf{eval}_A : P(X, \Sigma_X) \to [0, 1] : p \mapsto p(A)$$

measurable[2] for every measurable set $A \in \Sigma_X$.

In other words, $\Sigma_{\mathcal{P}X}$ can be written as

$$\Sigma_{\mathcal{P}X} = \sigma\Big(\big\{\mathsf{eval}_A^{-1}([0, r]) : 0 \leq r \leq 1, A \in \Sigma_X\big\}\Big),$$

see Equation (1.24), where we have additionally used that sets of the form $[0, r]$ with $0 \leq r \leq 1$ generate the Borel σ-algebra $\mathcal{B}([0, 1])$.

A morphism $f : X \to Y$ in **Meas**, i.e. a measurable function, is mapped to the pushforward

$$\mathcal{P}f := f_\star : P(X, \Sigma_X) \to P(Y, \Sigma_Y) : p \mapsto f_\star\, p := p \circ f^{-1}.$$

This definition coincides with the finite case which shows that the Giry monad generalizes the discrete probability monad (Example 1.9).

However, in contrast to the discrete probability functor, it is not apparent whether this functor is indeed an endofunctor (i.e. mapping into **Meas** again). For this reason, we have to show that $f_\star : \mathcal{P}X \to \mathcal{P}Y$ is a morphism in **Meas**.

---

[1] In the following, we will also use the notation $\mathcal{P}X := P(X, \Sigma_X)$, $\mathcal{P}^2 := \mathcal{P}\mathcal{P}X, \ldots$ to address the set of probability distributions on sets of probability distributions

[2] The set $[0, 1]$ is equipped with the Borel σ-algebra $\mathcal{B}([0, 1])$.

Lemma A.1. *Let $f : X \to Y$ be Σ_X - Σ_Y measurable. Then, the pushforward*

$$f_\star : P(X, \Sigma_X) \to P(Y, \Sigma_Y) : p \mapsto f_\star\, p$$

is $\Sigma_{\mathcal{P}X}$ - $\Sigma_{\mathcal{P}Y}$ measurable.

Proof. Since pre-images of the evaluation maps are the generating sets of $\Sigma_{\mathcal{P}Y}$, it remains to show that for every $A \in \Sigma_Y$ and every $0 \le r \le 1$ the following holds:

$$(f_\star)^{-1}\big(\mathsf{eval}_A^{-1}([0,r])\big) \in \Sigma_{\mathcal{P}X}.$$

For every probability measure $\mu \in P(X, \Sigma_X)$ and every measurable set $A \in \Sigma_Y$ we obtain

$$(\mathsf{eval}_A \circ f_\star\,)(\mu) = \mathsf{eval}_A(f_\star\, \mu) = \mu(f^{-1}(A)) = \mathsf{eval}_{f^{-1}(A)}(\mu).$$

This implies that,

$$(f_\star)^{-1}\big(\mathsf{eval}_A^{-1}([0,r])\big) = (\mathsf{eval}_A \circ f_\star)^{-1}([0,r]) = \mathsf{eval}_{f^{-1}(A)}^{-1}([0,r]) \in \Sigma_{\mathcal{P}X}$$

which shows the statement. □

Since $\mathcal{P}\mathrm{id}_X = \mathrm{id}_{\mathcal{P}X}$ and

$$\mathcal{P}(f \circ g) = (f \circ g)_\star = - \circ (f \circ g)^{-1} = \left(- \circ g^{-1}\right) \circ f^{-1} = \mathcal{P}(f) \circ \mathcal{P}(g)$$

$\mathcal{P}$ is indeed a functor.

In the following, we will equip $\mathcal{P}$ with a monad structure. In order to prove the commutativity of the diagrams in Definition 1.4, we will use the following lemma, which relates integrals over measures that are connected via a pushforward f. A proof of this statement can be found for example in [Bog07, Theorem 3.6.1].

Lemma A.2 (Change-of-variables formula)**.** *Let $f : X \to Y$ be a measurable function and $g : Y \to \mathbb{R}$ integrable with respect to $\mu \in P(Y, \Sigma_Y)$. Then, $g \circ f$ is integrable with respect to $f_\star\, \mu \in P(X, \Sigma_X)$ and*

$$\int_X (g \circ f)(x)\; \mu(\mathrm{d}x) = \int_Y g(y)\; (f_\star\, \mu)(\mathrm{d}y).$$

The basic idea behind the unit in the monad is similar to the discrete distribution monad. More specifically, the unit η_X assigns the Dirac δ-distribution at x to every element $x \in X$. Hence, we define the unit $\eta : \mathrm{id} \Rightarrow P$, by setting

$$\eta_X : (X, \Sigma_X) \to P(X, \Sigma_X) : x \mapsto \delta_x.$$

We start by showing that η_X is well-defined, i.e. it is measurable. To this end, let $0 \le r \le 1$ and $A \in \Sigma_X$. We have

$$\eta_X^{-1}\big((\mathsf{eval}_A)^{-1}([0,r])\big) = \begin{cases} A^c & r < 1 \\ X & r = 1 \end{cases} \in \Sigma_X$$

since $\delta_x(A) \in \{0, 1\}$.

Moreover, η is a natural transformation since the naturality square

$$\begin{array}{ccc} X & \xrightarrow{\quad f \quad} & Y \\ {\scriptstyle \eta_X}\big\downarrow & & \big\downarrow{\scriptstyle \eta_Y} \\ P(X,\Sigma_X) & \xrightarrow[f_\star]{} & P(Y,\Sigma_Y) \end{array}$$

commutes, which is obtained by

$$\delta_{f(x)}(A) = \delta_x(f^{-1}(A)) = \big(f_\star\, \delta_x\big)(A)$$

for every $x \in X$, morphism $f : X \to Y$ and $A \in \Sigma_Y$.

Defining the multiplication also extends the ideas from the discrete distribution monad. It maps any distribution on the space of distributions $P(X, \Sigma_X)$ to the mean distribution respective to the given distribution. More specifically, we define a multiplication $\mu : PP \Rightarrow P$ via

$$\mu_X : \mathcal{P}^2 X \to \mathcal{P}X : \mathcal{M} \mapsto \left(A \mapsto \int\limits_{P(X,\Sigma_X)} p(A)\ \mathcal{M}(\mathrm{d}p) \right)$$

Equivalently, for fixed $A \in \Sigma_X$ and fixed $\mathcal{M} \in \mathcal{P}^2 X$, the right-hand side can be understood as an expectation value of the evaluation map with respect to $\mathcal{M}$, i.e.

$$\mu_X(\mathcal{M})(A) = \mathbf{E}_{\mathcal{M}}[\mathsf{eval}_A(p)].$$

Showing that μ is a natural transformation is divided into three steps:

(i) For every distribution $\mathcal{M} \in \mathcal{P}^2 X$, we show that $\mu_X(\mathcal{M})$ is a probability measure.

(ii) The components μ_X are morphisms, i.e. they are measurable maps $\mathcal{P}^2 X \to \mathcal{P}X$.

(iii) The naturality square commutes for μ.

The first two parts show that every component μ_X is indeed a well-defined morphism in the category **Meas** while the last part shows the naturality of μ.

(i) Given $\mathcal{M} \in \mathcal{P}^2 X$, $\mu_X(\mathcal{M})$ is a measure. Obviously, $\mu_X(\mathcal{M})$ is indeed a function with co-domain $[0, 1]$ since

$$\mu_X(\mathcal{M})(X) = \int\limits_{P(X,\Sigma_X)} p(X)\ \mathcal{M}(\mathrm{d}p) = \int\limits_{P(X,\Sigma_X)} 1\ \mathcal{M}(\mathrm{d}p) = 1$$

where we used the fact that $\mathcal{M}$ is a probability measure. Moreover, for a countable family of disjoint sets $\{A_i\}_{i\in\mathbb{N}}$ in Σ_X we have that

$$\begin{aligned}\mu_X(\mathcal{M})\left(\bigsqcup_{i=1}^{\infty} A_i\right) &= \int\limits_{P(X,\Sigma_X)} p\left(\bigsqcup_{i=1}^{\infty} A_i\right) \mathcal{M}(\mathrm{d}p) \\ &= \int\limits_{P(X,\Sigma_X)} \sum_{i=1}^{\infty} p(A_i)\, \mathcal{M}(\mathrm{d}p) \\ &= \sum_{i=1}^{\infty} \int\limits_{P(X,\Sigma_X)} p(A_i)\, \mathcal{M}(\mathrm{d}p) = \sum_{i=1}^{n} \mu_X(\mathcal{M})(A_i).\end{aligned} \tag{A.1}$$

The third equality applies the dominated convergence theorem, which holds true since $p \mapsto \sum_{i=1}^{n} p(A_i)$ is measurable by definition of $\Sigma_{\mathcal{P}X}$ and bounded from above by 1 for every $n \in \mathbb{N}$.

(ii) $\mu_X : \mathcal{P}^2X \to \mathcal{P}X$ is a morphism, i.e. it is $\Sigma_{\mathcal{P}^2X}$ - $\Sigma_{\mathcal{P}X}$ - measurable. By definition of $\Sigma_{\mathcal{P}X}$ we have to show that for every $A \in \Sigma_X$ and $0 \leq r \leq 1$

$$\mu_X^{-1}\big(\mathsf{eval}_A^{-1}([0,r]\big) \in \Sigma_{\mathcal{P}^2X}$$

Put differently, we have to verify that the map

$$\mathcal{P}^2X \to [0,1] : \mathcal{M} \mapsto \int\limits_{P(X,\Sigma_X)} p(A)\, \mathcal{M}(\mathrm{d}p)$$

is measurable.

Instead of showing the statement for the function $p \mapsto p(A)$, we will show it for arbitrary measurable function $F : \mathcal{P}X \to [0,1]$. First, we show it for characteristic functions. For every measurable set $\mathcal{A} \in \Sigma_{\mathcal{P}X}$ we obtain

$$\mathcal{M} \mapsto \mathsf{eval}_{\mathcal{A}}(\mathcal{M}) = \mathcal{M}(\mathcal{A}) = \int\limits_{P(X,\Sigma_X)} \mathbb{1}_{\mathcal{A}}(p)\, \mathcal{M}(\mathrm{d}p).$$

This map is measurable by the definition of the σ-algebra of $\mathcal{P}^2X$. Hence, extending $\mathbb{1}_{\mathcal{A}}$ to any simple function $F : P(X,\Sigma_X) \to [0,1]$ keeps the mapping

$$\mathcal{M} \mapsto \int\limits_{P(X,\Sigma_X)} F(p)\, \mathcal{M}(\mathrm{d}p)$$

measurable. Since μ_X arises from setting $F := p \mapsto p(A)$ we obtain that μ_X is measurable by using the fact that the limit of measurable functions is still measurable and by applying the dominated convergence theorem.

(iii) μ is natural. Every morphism $f : X \to Y$ induces the morphism $f_{\star\star} := \mathcal{P}^2 f$ which reads

$$f_{\star\star} : \mathcal{P}^2X \to \mathcal{P}^2Y : \mathcal{M} \mapsto \Big(\mathcal{P}_Y \mapsto \mathcal{M}\big(f_\star^{-1}(\mathcal{P}_Y)\big)\Big).$$

This implies for every morphism $f : X \to Y$, every measurable set $A \in \Sigma_Y$ and every probability measure $\mathcal{M} \in \mathcal{P}^2 X$ that

$$\begin{aligned}\big(f_\star \mu_X(\mathcal{M})\big)(A) &= \int\limits_{P(X,\Sigma_X)} p(f^{-1}(A))\, \mathcal{M}(\mathrm{d}p) \\ &= \int\limits_{P(X,\Sigma_X)} \mathsf{eval}_A \circ f_\star \; \mathrm{d}\mathcal{M} \\ &= \int\limits_{P(Y,\Sigma_Y)} \mathsf{eval}_A \, \mathrm{d}(f_{\star\star}\mathcal{M}) \\ &= \int\limits_{P(Y,\Sigma_Y)} q(A)\, (f_{\star\star}\mathcal{M})(\mathrm{d}q) = \big(\mu_X(f_{\star\star}\mathcal{M})\big)(A).\end{aligned}$$

where we have applied Lemma A.2 in the third step. But this is precisely the statement that the naturality square

$$\begin{array}{ccc} \mathcal{P}^2 X & \xrightarrow{f_{\star\star}} & \mathcal{P}^2 Y \\ {\scriptstyle \mu_X}\downarrow & & \downarrow{\scriptstyle \mu_Y} \\ P(X,\Sigma_X) & \xrightarrow[f_\star]{} & P(Y,\Sigma_Y) \end{array}$$

commutes for every X, Y and $f : X \to Y$, which shows the naturality of μ.

To show that $(\mathcal{P}, \eta, \mu)$ is a monad, it remains to show that the three diagrams in Equation (1.12) commute. For this reason, we need one technical lemma.

Lemma A.3. *Let $f : \mathcal{P}X \to Y$ be a measurable function and $\mathfrak{M} \in \mathcal{P}^3 X$. Then, the following formula holds*

$$\int\limits_{\mathcal{P}X} f(p)\, \big(\mu_{\mathcal{P}X}(\mathfrak{M})\big)(\mathrm{d}p) = \int\limits_{\mathcal{P}^2X}\int\limits_{\mathcal{P}X} f(p)\, \mathcal{M}(\mathrm{d}p)\, \mathfrak{M}(\mathrm{d}\mathcal{M})$$

Proof. We first show the statement for simple functions f. For this reason, let $f = \mathbb{1}_{\mathcal{A}}$ be the characteristic function on a measurable subset $\mathcal{A} \subseteq \mathcal{P}X$. We then obtain

$$\begin{aligned}\int\limits_{\mathcal{P}X} \mathbb{1}_{\mathcal{A}}(p)\, \big(\mu_{\mathcal{P}X}(\mathfrak{M})\big)(\mathrm{d}p) &= \mu_{\mathcal{P}X}(\mathfrak{M})(\mathcal{A}) \\ &= \int\limits_{\mathcal{P}^2X} \mathcal{M}(\mathcal{A})\, \mathfrak{M}(\mathrm{d}\mathcal{M}) = \int\limits_{\mathcal{P}^2X}\int\limits_{\mathcal{P}X} \mathbb{1}_{\mathcal{A}}(p)\, \mathcal{M}(\mathrm{d}p)\, \mathfrak{M}(\mathrm{d}\mathcal{M})\end{aligned}$$

which shows the statement for characteristic functions. By linearity, this immediately proves the statement even for any simple function. Now, let f be an arbitrary measurable function and $s_i \nearrow f$ a monotone increasing sequence converging to f. Then, by the monotone convergence theorem, we have

$$S_i(\mathcal{M}) := \int\limits_{\mathcal{P}X} s_i(p)\, \mathcal{M}(\mathrm{d}p) \nearrow S(\mathcal{M}) = \int\limits_{\mathcal{P}X} f(p)\, \mathcal{M}(\mathrm{d}p).$$

Moreover, we have again by monotone convergence

$$\lim_{i\to\infty}\int_{\mathcal{P}X} s_i(p)\ (\mu_{\mathcal{P}X}(\mathfrak{M}))(\mathrm{d}p) = \int_{\mathcal{P}X} f(p)\ (\mu_{\mathcal{P}X}(\mathfrak{M}))(\mathrm{d}p).$$

Therefore, we have

$$\begin{aligned}\int_{\mathcal{P}X} f(p)\ \big(\mu_{\mathcal{P}X}(\mathfrak{M})\big)(\mathrm{d}p) &= \lim_{i\to\infty}\int_{\mathcal{P}X} s_i(p)\ \big(\mu_{\mathcal{P}X}(\mathfrak{M})\big)(\mathrm{d}p)\\ &= \lim_{i\to\infty}\int_{\mathcal{P}^2X} S_i(\mathcal{M})\ \mathfrak{M}(\mathrm{d}\mathcal{M})\\ &= \int_{\mathcal{P}^2X} S(\mathcal{M})\ \mathfrak{M}(\mathrm{d}\mathcal{M}) = \int_{\mathcal{P}^2X}\int_{\mathcal{P}X} f(p)\ \mathcal{M}(\mathrm{d}p)\ \mathfrak{M}(\mathrm{d}\mathcal{M})\end{aligned}$$

where we applied the statement for simple functions in the second step and the monotone convergence theorem in the third step. □

We now state and prove the main statement of this appendix.

Theorem A.4. $(\mathcal{P},\eta,\mu)$ *forms a monad.*

Proof. We have to show that the diagrams in Equation (1.12) commute.

(i) We have to verify that $\mu_X \circ \eta_{\mathcal{P}X} = \mathrm{id}_{\mathcal{P}X}$.

Let $p \in P(X,\Sigma_X)$. For every $A \in \Sigma_X$ we obtain

$$(\mu_X \circ \eta_{\mathcal{P}X}(p))(A) = \mu_X(\delta_p)(A) = \int_{P(X,\Sigma_X)} q(A)\ \delta_p(\mathrm{d}q) = p(A)$$

which shows the statement.

(ii) We have to show that $\mu_X \circ (\mathcal{P}\eta_X) = \mathrm{id}_{\mathcal{P}X}$.

Let again $A \in \Sigma_X$ and $p \in \mathcal{P}X$. Then, we have

$$\begin{aligned}\big(\mu_X \circ (\mathcal{P}\eta_X)(p)\big)(A) &= \int_{P(X,\Sigma_X)} \mathsf{eval}_A\, \mathrm{d}\big((\eta_X)_\star\, p\big)\\ &= \int_X \mathrm{ev}_A \circ \eta_X\, \mathrm{d}p\\ &= \int_X \delta_x(A)\ p(\mathrm{d}x) = p(A)\end{aligned}$$

where we have used Lemma A.2 in the second equation.

(iii) We finally verify that $\mu_X \circ \mathcal{P}\mu_X = \mu_X \circ \mu_{\mathcal{P}X}$. We show this by manipulating both sides of the equation. Let $\mathfrak{M} \in \mathcal{P}^3 X$ and $A \in \Sigma_X$. Then, the left-hand side leads to

$$\begin{aligned}(\mu_X \circ \mathcal{P}\mu_X)(\mathfrak{M})(A) &= \int\limits_{P(X,\Sigma_X)} \mathsf{eval}_A \, \mathrm{d}((\mu_X)_\star \, \mathfrak{M}) \\ &= \int\limits_{\mathcal{P}^2 X} \mu_X(\mathcal{M})(A) \, \mathfrak{M}(\mathrm{d}\mathcal{M}) \\ &= \int\limits_{\mathcal{P}^2 X} \int\limits_{P(X,\Sigma_X)} p(A) \, \mathcal{M}(\mathrm{d}p) \, \mathfrak{M}(\mathrm{d}\mathcal{M})\end{aligned}$$

where we have used Lemma A.2 in the second equation. Manipulating the right-hand side leads to

$$\begin{aligned}(\mu_X \circ \mu_{\mathcal{P}X})(\mathfrak{M})(A) &= \int\limits_{P(X,\Sigma_X)} \mathsf{eval}_A \, \mathrm{d}(\mu_{\mathcal{P}X}(\mathfrak{M})) \\ &= \int\limits_{\mathcal{P}^2 X} \int\limits_{P(X,\Sigma_X)} \mathsf{eval}_A \, \mathrm{d}\mathcal{M} \, \mathfrak{M}(\mathrm{d}\mathcal{M}) \\ &= \int\limits_{\mathcal{P}^2 X} \int\limits_{P(X,\Sigma_X)} p(A) \, \mathcal{M}(\mathrm{d}p) \, \mathfrak{M}(\mathrm{d}\mathcal{M})\end{aligned}$$

where we have used Lemma A.3 in the second equation. Since both manipulations lead to the same expression this shows the statement. □