Abstraction between Structural Causal Models: A Review of Definitions and Properties

Abstract

Structural causal models (SCMs) are a widespread formalism to deal with causal systems. A recent direction of research has considered the problem of relating formally SCMs at different levels of abstraction, by defining maps between SCMs and imposing a requirement of interventional consistency. This paper offers a review of the solutions proposed so far, focusing on the *formal properties* of a map between SCMs, and highlighting the different *levels* (structural, distributional) at which these properties may be enforced. This allows us to distinguish families of abstractions that may or may not be permitted by choosing to guarantee certain properties instead of others. Such an understanding not only allows to distinguish among proposal for causal abstraction with more awareness, but it also allows to tailor the definition of abstraction with respect to the forms of abstraction relevant to specific applications.

1 INTRODUCTION

Modelling causal systems at different levels of abstraction is a central feature in our understanding of the world and in our scientific endeavours. Generating representations at the right level of granularity is often the implicit goal of unsupervised and representation learning; in the causal setup, learning high-level causal representations is the central task of causal representation learning (CRL) [Schölkopf et al., 2021].

A study of the formal properties of abstraction in the context of structural causal models (SCM) has been proposed in a few recent papers [Rubenstein et al., 2017, Beckers and Halpern, 2019, Beckers et al., 2020, Rischel, 2020, Rischel and Weichwald, 2021, Otsuka and Saigo, 2022]. All these works have in common the presentation of abstraction as a map between SCMs that would be consistent (or approximately consistent) with respect to interventions; that is, a map that would commute (or approximately commute) with respect to interventions, thus enforcing the intuition that working first at the microlevel and then abstracting to the macrolevel would produce the same result as immediately switching to the macrolevel and then working there.

Yet SCMs are complex structured objects and the abstraction maps proposed in the existing literature vary considerably between them. This is reflected in the different nomenclature (abstraction, transformation), theoretical background (measure theory, category theory) and concrete mappings that are allowed to be considered as abstractions. In this paper, we focus only on definitions of abstractions, without discussing explicitly the requirement of (interventional) consistency. The objective is to reconcile existing formalisms in terms of the *levels* at which an abstraction is defined and the *properties* that are enforced. This will bring some immediate benefits, among which: (i) a better understanding of the alternative formalisms being proposed; (ii) an insight into which maps we would want to be considered as abstractions; (iii) in a CRL framework, the ability to tailor the definition according to the properties we actually want to hold.

The paper is organized as follows: in Section 2 we review the definition of SCM and abstraction, and we summarize relevant work in the literature; in Section 3 we analyze how abstractions can be characterized at two different levels, and which properties can be required; finally, in Section 4, we discuss the implications of our analysis, summarizing how properties at different levels may be used to restrict the definition of an abstraction.

2 BACKGROUND

2.1 STRUCTURAL CAUSAL MODELS

A (semi-Markovian) SCM Pearl [2009], Peters et al. [2017] $\mathcal{M}$ is a tuple $\langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ where:

Submitted to the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022). **To be used for reviewing only**.

- $\mathcal{X}$ is a set of N endogenous variables, that is, observed variables of interest in the model; each variable $X_i \in \mathcal{X}$ is defined on a finite domain $\mathcal{M}[X_i]$;
- $\mathcal{U}$ is a set of N exogenous variables, that is, unobserved variables accounting for factors beyond the scope of the model; each variable $U_i \in \mathcal{U}$ is defined on a domain $\mathcal{M}[U_i]$;
- $\mathcal{F}$ is a set of N modular and measurable structural functions, one for each endogenous variable X_i; the value assumed by an endogenous node is deterministically computed as $X_i = f_i(pa(X_i), U_i)$, where $pa(X_i) \subseteq \mathcal{X} \setminus \{X_i\}$ is the set of endogenous variables directly affecting X_i;
- $\mathcal{P}$ is a set of N probability distributions p_i, one for each exogenous variable.

A SCM admits an underlying acyclic graph $\mathcal{G}_{\mathcal{M}} = \langle V, S \rangle$ whose set of vertices V is given by the endogenous variables in $\mathcal{X}$ and the set of edges E is given by the structural functions [Pearl, 2009].

Our definition of SCM makes a few assumption about the causal model such as: (a) finite number of endogenous variables; (b) finite domains for the endogenous variables; (c) unique exogenous variable per endogenous variable; (d) exogenous variables not necessarily independent; (e) modularity and measurability of structural functions; (f) acyclic underlying graph. All these assumptions are discussed at length in the literature [Pearl, 2009, Peters et al., 2017], and they represent a minimum shared set of assumptions between the existing proposals for abstraction. In the following, we will therefore assume that our SCMs always comply with these assumptions, unless otherwise stated.

Thanks to the assumptions above, it is possible to pushforward probability distributions from the exogenous variables onto the endogenous variables. Furthermore, we can conveniently represent a SCM as a collection of sets and a collection of mechanisms $\mathcal{M}[\phi_{X_i}]$, one for each endogenous variable, encoded as a stochastic matrix (or Markov kernel) [Rischel, 2020].

SCMs can be modified via intervention through the $\iota : do(\mathbf{X} = \mathbf{x})$ operator which fixes a set of endogenous variables $\mathbf{X} \subseteq \mathcal{X}$ to the values $\mathbf{x}$ [Pearl, 2009]. An intervention ι on a SCM $\mathcal{M}$ produces a new post-interventional model $\mathcal{M}_\iota$.

2.2 ABSTRACTIONS

Given two SCMs $\mathcal{M}^m = \langle \mathcal{X}, \mathcal{U}, \mathcal{F}, \mathcal{P} \rangle$ and $\mathcal{M}^M = \langle \mathcal{X}', \mathcal{U}', \mathcal{F}', \mathcal{P}' \rangle$, an abstraction is a map $\boldsymbol{\alpha} : \mathcal{M}^m \to \mathcal{M}^M$ between the two SCMs. For simplicity, we will refer to the model $\mathcal{M}^m$ in the domain of $\boldsymbol{\alpha}$ as the micromodel, or the low-level model, and to the model $\mathcal{M}^M$ in the codomain of $\boldsymbol{\alpha}$ as the macromodel, or the high-level model.

Characterizing an abstraction requires a concrete specification of the map $\boldsymbol{\alpha} : \mathcal{M}^m \to \mathcal{M}^M$, together with the properties that this map must satisfy. We can distinguish two forms of properties that are usually required: (i) formal properties regarding the map itself, that statically constrain the map (e.g.: surjectivity); and (ii) consistency properties guaranteeing that working either with the micromodel or the macromodel would lead to consistent results. The requirement of consistency often takes the form of (perfect or approximate) commutativity with respect to interventions, which translates the understanding that the results observed from the micromodel and the macromodel should be consistent when we intervene on them. In the next sections we will focus on formal properties, leaving a detailed study of the property of consistency for future work.

2.3 RELATED WORK

A first definition of abstraction between SCMs (for which assumption (b) and (f) are not necessarily required) has been put forth by Rubenstein et al. [2017]: a (τ-ω)-*transformation* is a map $\tau : \mathcal{M}^m[\mathcal{X}] \to \mathcal{M}^M[\mathcal{X}']$ from the joint domain of the outcomes of all the microvariables to the joint domain of the outcomes of all the macrovariables, which is interventionally consistent with respect to a set of interventions of interest $\mathcal{I}$.

Dealing with deterministic SCMs, Beckers and Halpern [2019] build over this work by presenting the notion of *uniform* (τ-ω)-*transformation* as a map $\tau : \mathcal{M}^m[\mathcal{X}] \to \mathcal{M}^M[\mathcal{X}']$ that would satisfy interventional consistency for any choice of distribution over the exogenous variables in $\mathcal{M}^m$. Further, they also offer a stronger defintion of τ-*abstraction* as a surjective function $\tau : \mathcal{M}^m[\mathcal{X}] \to \mathcal{M}^M[\mathcal{X}']$ with an associated surjective function between exogenous nodes of the micromodel and the macromodel, and an associated function between set of interventions in the micromodel and the macromodel.

Relying on category theory, Rischel [2020], Rischel and Weichwald [2021] suggest a more detailed definition of an abstraction. An abstraction $\boldsymbol{\alpha}$ is a a tuple $\langle R, a, \alpha_{X'} \rangle$ given by a set of relevant nodes in the micromodel $R \subseteq \mathcal{X}$, a surjective map between variables in the micromodel and macromodel $a : R \to \mathcal{X}'$, and a collection of surjective maps $\alpha_{X'}$, one for each macrovariable: $\alpha_{X'} : \mathcal{M}^m[a^{-1}(X_i)] \to \mathcal{M}^M[X_i']$; consistency is evaluated with respect to interventions.

Finally, Otsuka and Saigo [2022] offers an alternative category-theoretical definition, by requiring an abstraction $\boldsymbol{\alpha}$ to be defined by first finding a graph homomorphism from $\mathcal{G}_{\mathcal{M}^m}$ to $\mathcal{G}_{\mathcal{M}^M}$, expressing the micromodel $\mathcal{M}^m$ and the macromodel $\mathcal{M}^M$ as functors, and finally by seeing the abstraction as a natural transformation between the two functors. Interventions, once again used to enforce interventional

consistency, take the form of endofuctors.

3 CHARACTERIZING ABSTRACTIONS

In the previous section we have briefly reviewed different specifications of abstractions. Here, we want to bring all these definitions together by reviewing how they satisfy different types of properties at different levels.

3.1 LEVELS OF AN ABSTRACTION

A SCM is a mathematical object that brings together two types of information: the statistical data-driven behaviour of a set of variables in a given model (which could be either the pre-interventional $\mathcal{M}$ or a post-interventional $\mathcal{M}_\iota$), and the causal assumption-driven structure connecting variables (through causal links) and models (through interventions).

Although these two types of information are not independent, it can be useful, when specifying an abstraction, to define explicitly how the abstraction behaves with respect to them. We can therefore distinguish these two levels:

1. A *structural* level which deals with a map $\mathcal{G}_{\mathcal{M}^m} \to \mathcal{G}_{\mathcal{M}^M}$, that is, how the underlying graph $\mathcal{G}_{\mathcal{M}^m}$ of a micromodel is transformed through the abstraction $\boldsymbol{\alpha}$. This level is concerned with maps between nodes and, possibly, edges. The structural level accounts for how an abstraction can transfrom the identity of the causes, and how it can affect flow and the directionality of causes and effects.
2. A *distributional* level which deals with maps $\mathcal{M}^m[\mathbf{X}] \to \mathcal{M}^M[\mathbf{X}']$, $\mathbf{X} \subseteq \mathcal{X}, \mathbf{X}' \subseteq \mathcal{X}'$, that is, how outcomes and distributions implied by a micromodel are transformed through the abstraction $\boldsymbol{\alpha}$. This level may define a single map $\mathcal{M}^m[\mathcal{X}] \to \mathcal{M}^M[\mathcal{X}']$ relating the whole joint outcome spaces of the models [Rubenstein et al., 2017, Beckers and Halpern, 2019], or a collection of maps, such as $\mathcal{M}^m[X_i] \to \mathcal{M}^M[X'_i]$, relating the outcome space of single or subsets of variables [Rischel, 2020]. These maps implicitly define pushforward maps from the probability measure over the set $\mathcal{M}^m[\mathbf{X}]$ onto the probability measure over the set $\mathcal{M}^M[\mathbf{X}']$. The distributional level thus accounts for how an abstraction can affect the representation and the strengths of relationships of cause and effect.

This separation of concerns is already implicitly present in Rischel [2020], Rischel and Weichwald [2021] where an abstraction is defined on two levels, as a mapping a between variables and a collection of mappings $\alpha_{X'}$ between outcomes. This distinction is given stronger emphasis in Otsuka and Saigo [2022] where an explicit mapping between graphs (via a graph homomorphism) and a mapping between outcomes (via a natural transformation) are required; this setup follows from the category-theoretical approach of representing a model (e.g., a casual model in Jacobs et al. [2019] or a database in Spivak [2014]) at a syntactic level capturing the underlying structure (in a free category generated from a graph) and at a semantic level (as a functor to a category that instantiates specific values).

3.2 PROPERTIES OF AN ABSTRACTION

Relying on the distinction between the two levels above, we now analyze what specific properties may be required at each level, and what forms of abstraction they entail.

Properties at the structural level. Let us first consider a map $\mathcal{G}_{\mathcal{M}^m} \to \mathcal{G}_{\mathcal{M}^M}$. First of all, notice that between two graphs we can typically establish a *function on nodes* $\mathcal{X}^m \to \mathcal{X}^M$, or a stricter *structure-preserving functor* $\mathcal{C}_{\mathcal{M}^m} \to \mathcal{C}_{\mathcal{M}^M}$, where $\mathcal{C}_{\mathcal{M}^m}, \mathcal{C}_{\mathcal{M}^M}$ are the free categories generated from the DAGs of the micromodel and the macromodel [Otsuka and Saigo, 2022]. Other solutions, such as a *function on edges*, are not considered in the literature. Let us now analyze these two approaches and the connected properties, starting from the functional map:

- *Functionality:* a function on the nodes requires a mapping of all the nodes of $\mathcal{G}_{\mathcal{M}^m}$ onto the nodes of $\mathcal{G}_{\mathcal{M}^M}$. This is a property expressing the requirement that each node in the micromodel is abstracted, and no node can be ignored. This translates the idea of abstraction as a strict coarsening of the variables in a micromodel; structural functionality is implied by functoriality in Otsuka and Saigo [2022] .
  On the other hand, dropping this requirement means that microvariables may just be ignored; this may happen when we want to consider abstractions that capture or synthesize only a sub-part of a micromodel, discarding irrelevant information or indirectly relegating it over the exogenous variables; this is the case in Rischel [2020] where the mapping a between variables has, as its domain, the restriction of $\mathcal{X}^m$ to relevant variables $R \subseteq \mathcal{X}^m$.
- *Functional surjectivity:* functional surjectivity requires a mapping such that all the nodes of $\mathcal{G}_{\mathcal{M}^M}$ are mapped from the nodes of $\mathcal{G}_{\mathcal{M}^m}$. This reflects the understanding that all the variables in a macromodel are explained by one or more variables in the micromodel. Functional surjectivity is explicitly required by Rischel [2020].
  Dropping this condition is equivalent to accepting that a macromodel may include variables that have no explanation in the micromodel; this allows for forms of abstractions akin to compression and embedding, or cases in which exogenous factors of variance in the micromodel has implicitly become endogenous in the macromodel; this choice is made in Rubenstein et al. [2017] and Otsuka and Saigo [2022].

- *Functional injectivity:* functional injectivity requires a mapping such that all the nodes of $\mathcal{M}^m$ are mapped to distinct nodes of $\mathcal{M}^M$. This would encode the desideratum that our abstraction guarantees that no variables are collapsed together. Functional injectivity allows for forms of abstractions such as embeddings, or abstractions in which variables are identical, but their domains and dynamics may be simplified. No work in the literature enforces this property. Without functional injectivity, coarsening and collapsing of variables from the micromodel to the macromodel is allowed.
- *Functional bijectivity:* a trivial property of functional bijection follows by enforcing surjectivity and injectivity. This would allow only for abstractions where there is an isomorphism (an identity or a permutation) between the nodes. No work in the literature enforces this property.

A functorial map implies a more structured map, and it may allow for analogous properties:

- *Functoriality:* a structure-preserving functor requires an explicit mapping of nodes and edges in $\mathcal{M}^m$, such that composition is preserved[1]. This is a strong requirement implying that we want to deal only with abstractions that preserve the directionality of causes and that do not arbitrarily drop any causal link; this is enforced in Otsuka and Saigo [2022] by requiring a graph homomorphism between $\mathcal{G}_{\mathcal{M}^m}$ and $\mathcal{G}_{\mathcal{M}^M}$. Violating this property means that we accept abstractions that may arbitrarily drop some causal connections or even reverse them; this may be acceptable for some types of abstractions, such as macromodels that may ignore causal connections with a strength under a certain threshold [Janzing et al., 2013]. Non-functoriality is accepted in Rischel [2020], where the mapping a between variables does not express any constraints between edges in the micromodel and macromodel.
- *Functorial fullness:* functorial fullness requires surjectivity between the hom-sets of the micromodel and the macromodel; that is, if X'_i and X'_j are two nodes in the macromodel mapped respectively from X_i and X_j, then we want a surjective map between edges going from X_i to X_j in the micromodel and edges going from X'_i to X'_j in the macromodel. This reflects the understanding that every causal link, either direct or mediated (given by composition of edges), in the macromodel must have a corresponding causal link in the micromodel. No work in the literature enforces this property. No fullness implies the possible presence of additional relationships of cause and effects in the macromodel which are absent in the micromodel.
- *Funtorial faithfulness:* functorial faithfullness requires injectivity between the hom-sets of the micromodel and the macromodel; that is, if X'_i and X'_j are two nodes in the macromodel mapped respectively from X_i and X_j, then we want an injective map between edges going from X_i to X_j in the micromodel and edges going from X'_i to X'_j in the macromodel. This implies that every causal link, either direct or mediated (given by composition of edges), in the micromodel has a corresponding causal link in the macromodel. No work in the literature enforces this property. Without functorial faithfulness, collapsing of edges is possible.
- *Functorial fully faithfulness:* a trivial property of fully faithfulness follows by enforcing fullness and faithfulness. Together with bijectivity on nodes, this would allow only for strictly structure-preserving abstractions expressing an isomorphism (an identity) between nodes and edges. No work in the literature enforces this property.

A derived property that is sometimes discussed is the *invertibility* of the map $\mathcal{G}_{\mathcal{M}^m} \to \mathcal{G}_{\mathcal{M}^M}$; it immediately follows that functional bijectivity allows perfect invertibility on nodes; functional surjectivity allows set-invertibility on nodes; functorial fully faithfulness allows perfect invertibility on edges; and, functorial fullness allows set-invertibility on edges[2].

Finally, there are two important properties that are commonly taken for granted, but which could be changed or relaxed, thus providing abstractions with different degrees of freedom:

- *Micro-to-macro:* it seems a reasonable assumption that the directionality of an abstraction should be $\mathcal{G}_{\mathcal{M}^m} \to \mathcal{G}_{\mathcal{M}^M}$, as we normally move from micromodels to macromodels by reducing their complexity. In some contexts, it may be of interest to consider possibly stochastic maps $\mathcal{G}_{\mathcal{M}^M} \to \mathcal{G}_{\mathcal{M}^m}$ going from a macromodel to a micromodel.
- *Determinism:* the mapping from the microlevel to the macrolevel is often interpreted as a deterministic supervenient map. However, in case of limited knowledge and uncertainty, the overall requirement of functoriality or functionality may be dropped by requiring the map $\mathcal{G}_{\mathcal{M}^M} \to \mathcal{G}_{\mathcal{M}^m}$ to be a stochastic map. This would allow for forms of abstraction in which the contribution of a microvariable may be split among a collection of macrovariables. This actually happens in Rubenstein et al. [2017], where outcomes of a microlevel variable may be mapped to outcomes of diffferent macrolevel variables, thus implying a splitting of the contribution of a microvariable across many macrovariables.

[1] This is formally a functor between $\mathcal{C}_{\mathcal{M}^m} \to \mathcal{C}_{\mathcal{M}^M}$ that maps objects and morphisms, while preserving identity and composition.

[2] Invertibility on edges is always with reference to edges between nodes that are mapped under $\mathcal{C}_{\mathcal{M}^m} \to \mathcal{C}_{\mathcal{M}^M}$.

Properties at the distributional level. Let us move on to consider the map $\mathcal{M}^m[\mathbf{X}] \to \mathcal{M}^M[\mathbf{X}']$. Differently from the previous map between graphs, we are now dealing with a map between sets that takes the form of a *function*. As before, let us investigate the properties that can be enforced on this map:

- *Functionality:* this property requires all the outcomes of the micromodel $\mathcal{M}^m$ to be mapped onto outcomes of the macromodel $\mathcal{M}^M$; there is no microlevel outcome, no matter how unlikely, that can be dropped and ignored in the macrolevel outcomes. Beyond a conceptual justification, a mathematical reason underpins this setup: a (measurable) function is necessary to guarantee a proper pushforward of the probability distribution over the outcomes of the micromodel $\mathcal{M}^m$ onto the outcomes of the macromodel $\mathcal{M}^M$; Functionality is a common assumption shared so far by all works on abstraction.
  Dropping this assumption would possibly require us to define a set of relevant outcomes to be mapped (in analogy with the definition of a set of relevant variable R as a domain for the structural-level mapping a between variables in Rischel [2020]), together with a renormalization before or after the pushforward. This form of abstraction may reflect a mapping in which we capture the behaviour of a micromodel over a specific domain of outcomes, ignoring perhaps limit cases.
- *(Functional) surjectivity:* surjectivity requires that every outcome of the macromodel $\mathcal{M}^M$ is mapped from the domain of outcomes of the micromodel $\mathcal{M}^m$. This expresses the understanding that all possible macrolevel outcomes are explained at the microlevel. This property is usually considered a staple of abstraction, and for this reason it is introduced by Beckers and Halpern [2019] in their definition of τ-abstraction, and it is enforced from the beginning in Rischel [2020].
  Dropping surjectivity allows for some macrolevel outcomes without an explanation in the micromodel. This is accepted in Rubenstein et al. [2017] and, implicitly, in Otsuka and Saigo [2022].
- *(Functional) injectivity:* injectivity requires all the outcomes of the micromodel $\mathcal{M}^m$ to be mapped to different outcomes in the macromodel $\mathcal{M}^M$. This leads to a form of abstraction where the outcomes at microlevel are embedded into the set of outcomes at the macrolevel. No work in the literature enforces this property.
- *(Functional) bijectivity:* a trivial property of bijectivity follows from surjectivity and injectivity. This would allow for abstractions where there is an isomorphism (an identity or a permutation) between micromodel and macromodel outcomes. Every outcome in the micromodel would have a unique corresponding outcome in the macromodel with the exact same probability. No work in the literature enforces this property.

A property of invertibility would follow from surjectivity and injectivity; furthermore, as before, two additional properties, normally taken for granted, are:

- *Micro-to-macro:* instead of assuming the directionality of an abstraction being $\mathcal{M}^m[\mathbf{X}] \to \mathcal{M}^M[\mathbf{X}']$, we could consider possibly stochastic maps $\mathcal{M}^M[\mathbf{X}'] \to \mathcal{M}^m[\mathbf{X}]$ going from the outcomes of the macromodels to the (sets of) outcomes of the micromodel.
- *Determinism:* the assumption of determinism reflects the understanding that the mapping from outcomes in the micromodel to outcomes in the macromodel expresses an exact deterministic relationship; if we were to drop this assumption, such as in the case in which we were to have limited knowledge, we could express the map $\mathcal{M}^m[\mathbf{X}] \to \mathcal{M}^M[\mathbf{X}']$ as a stochastic map. This would allow for forms of abstraction in which the contribution of an outcome in the micromodel would be split among a collection of outcomes in the macromodel; this could capture, for instance, our uncertainty on how an outcome at the microlevel would correspond to an outcome at the macrolevel.

4 DISCUSSION

Abstractions, like SCMs, are complex mathematical objects encoding not just statistical information but also causal assumptions normally expressed in the form of a DAG. Relying on a definition of an abstraction on the structural and distributional level (as proposed by Otsuka and Saigo [2022], Rischel [2020], Rischel and Weichwald [2021]) guarantees a more fine-grained control on the definition of the form of an abstraction compared to approaches that focus only on the distributional level (as done by Rubenstein et al. [2017], Beckers and Halpern [2019], Beckers et al. [2020]).

Although some definitions of abstraction do not explicitly characterize the abstraction on the structural level, information about the structure of the graphs underlying the SCMs is not generally discarded but mediated by consistency. In Rubenstein et al. [2017], although knowledge of the structure of the SCM is not necessary, the distributions over the intervened models generated by the set $\mathcal{I}$ of interventions of interest force the structure of the SCM $\mathcal{M}^m$ in the $\mathcal{I}$-Markov equivalence class Hauser and Bühlmann [2015]; moreover, correspondence between distributions as required by consistency impose a further constraint on the form of the micrograph and the macrograph. However, the structural relationship is left implicit, and no formal properties may be requested for a mapping at the structural level. The progressive strengthening in Beckers and Halpern [2019] of the original definition of $(\tau\text{-}\omega)$-abstraction may be seen as a way to sufficiently constrain the distributional-level

definition so that undesired mappings between incompatible models whose distributions had been artificially set to satisfy the requirement of consistency are ruled out.

By explicitly considering the structural and the distributional level, we have seen how enforcing different properties on the two levels of an abstraction may allow us to enlarge or restrict the family of transformations that can be considered as legitimate abstractions. Table 1 and 2 in the Additional Material summarize the properties that we defined and the types of abstractions that would be admitted by adding or dropping specific requirements. For instance, requesting the property of functoriality at the structural level rules out the possibility that an abstraction could simplify a model by dropping causal links with limited strength. If such a simplification does not fit our understanding of abstraction, the property of functoriality could be part of the definition of abstraction. The present treatment has highlighted which types of transformations would be preserved or rejected through the enforcement of specific properties, but it does not argue which ones should be selected.

Furthermore, the distinction between the structural and distributional levels also brings clarity on how an abstraction could independently act on the two levels to produce different results. *Coarsening* is a term often used as a generic synonym for abstraction. Yet, a coarsening may act on the structural or distributional level. A coarsening at the structural level implies a reduction in the number of observables. A coarsening at the distributional level implies a reduction in the resolution of the observables. Similarly, a coarsening on the structural level may be paired with an identity on the distributional level; this would mean that the number of observables is reduced, but the resolution or the dimensionality of the new macrolevel observables is large enough that we can have a one-to-one map of microlevel outcomes to macrolevel outcomes.

Interestingly, the discussion on abstraction has focused on graphs and sets, and never explicitly on distributions and mechanisms. Distributions are complex objects to map, and focusing on the underlying set simplifies the problem; a map between microlevel and macrolevel sets entails an automatic pushforward of the microlevel distributions. Mechanisms are instead completely underdetermined; in its definition, an abstraction does not express any constraint on the macrolevel mechanisms; possible constraints are instead introduced via the requirement of consistency.

Indeed, a complete operative specification of abstraction can not preclude a definition of the consistency properties that an abstraction is supposed to guarantee. However, this paper has highlighted that even restricting our attention to the formal properties of an abstraction, there is a wide degree of freedom in defining what class of transformations should be considered an abstraction. This understanding has practical consequences: selecting the right properties would then be relevant in any learning effort, as this may reduce the space of functions (or functors) over which we want to search for an abstraction.

References

Sander Beckers and Joseph Y Halpern. Abstracting causal models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2678–2685, 2019.

Sander Beckers, Frederick Eberhardt, and Joseph Y Halpern. Approximate causal abstractions. In *Uncertainty in Artificial Intelligence*, pages 606–615. PMLR, 2020.

Alain Hauser and Peter Bühlmann. Jointly interventional and observational data: estimation of interventional markov equivalence classes of directed acyclic graphs. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(1):291–318, 2015.

Bart Jacobs, Aleks Kissinger, and Fabio Zanasi. Causal inference by string diagram surgery. In *International conference on foundations of software science and computation structures*, pages 313–329. Springer, 2019.

Dominik Janzing, David Balduzzi, Moritz Grosse-Wentrup, and Bernhard Schölkopf. Quantifying causal influences. *The Annals of Statistics*, 41(5):2324–2358, 2013.

Jun Otsuka and Hayato Saigo. On the equivalence of causal models: A category-theoretic approach. *arXiv preprint arXiv:2201.06981*, 2022.

Judea Pearl. *Causality*. Cambridge University Press, 2009.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: Foundations and learning algorithms*. MIT Press, 2017.

Eigil F Rischel and Sebastian Weichwald. Compositional abstraction error and a category of causal models. *arXiv preprint arXiv:2103.15758*, 2021.

Eigil Fjeldgren Rischel. The category theory of causal models. 2020.

Paul K Rubenstein, Sebastian Weichwald, Stephan Bongers, Joris M Mooij, Dominik Janzing, Moritz Grosse-Wentrup, and Bernhard Schölkopf. Causal consistency of structural equation models. In *33rd Conference on Uncertainty in Artificial Intelligence (UAI 2017)*, pages 808–817. Curran Associates, Inc., 2017.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

David I Spivak. *Category theory for the sciences*. MIT Press, 2014.

A ADDITIONAL MATERIAL

	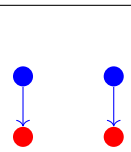	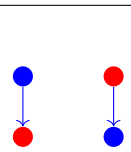	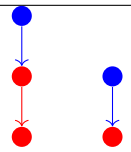	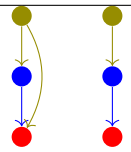	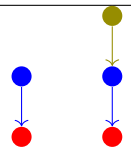	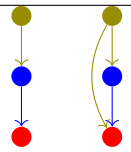	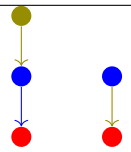	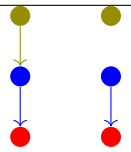	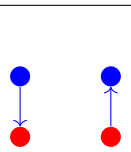	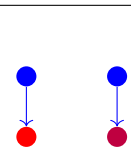	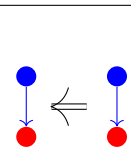
	Identity	*Node permutation*	*Node coarsening*	*Edge coarsening*	*Node embedding*	*Edge embedding*	*Node dropping*	*Edge dropping*	*Causal reversal*	*Causal splitting*	*Abs. Reversal*
Functionality	✓	✓	✓	✓	✓	✓	×	✓	✓	×	-
Surjectivity	✓	✓	✓	✓	×	✓	×	✓	✓	×	-
Injectivity	✓	✓	×	✓	✓	✓	×	✓	✓	×	-
Bijectivity	✓	✓	×	✓	×	✓	×	✓	✓	×	-
Functoriality	✓	×	✓	✓	✓	✓	×	×	×	×	-
Fullness	✓	×	✓	✓	✓	×	×	×	×	×	-
Faithfulness	✓	×	✓	×	✓	✓	×	×	×	×	-
Fully Faithfulness	✓	×	✓	×	✓	×	×	×	×	×	-
Non-Determinism	-	-	-	-	-	-	-	-	-	✓	-
Macro-to-micro	-	-	-	-	-	-	-	-	-	-	✓

Table 1: Summary of structural-level properties and types of abstraction. The table expresses whether enforcing a specific structural-level property allows for the definition of certain types of abstraction.
In the first row, a color-coded example of each type of abstraction is shown; the graphs of two simple chain-like models are presented: a micromodel on the left, and the corresponding macromodel on the right. The color of the nodes (and the edges, where relevant) in the micromodel illustrate to which corresponding nodes (and edges) of the macromodel they are mapped onto.
Observe that an *identity abstraction* is allowed under the enforcement of any of the properties. A *node permutation abstraction* is, in general, not allowed when enforcing functoriality, as it breaks the compositionality of edges. A *node coarsening abstraction* is possible if we do not require injectivity, while a *edge coarsening abstraction* is allowed if we do not require faithfulness. A *node embedding abstraction* requires surjectivity, while an *edge embedding abstraction* necessitates no fullness. *Node dropping abstraction* is not allowed under any property. *Edge dropping abstraction* and *causal reversal abstraction* are allowed if we do not require functoriality.
Notice that the bijectivity row and the fully fatihfulness row are just derived from surjectivity and injectivity or from fullness and faithfulness.
Finally the last two rows and columns consider less common properties and scenarios. A *causal splitting abstraction*, depicted by having a macrolevel node colored purple because it is mapped by both the blue and red microlevel nodes, requires a non-deterministic map. An *abstraction reversal*, represented by a double arrow going from the macrolevel to the microlevel, requires a structural-level map going from the graph of the macromodel to the graph of the micromodel.

	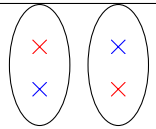	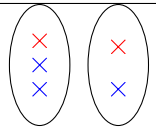	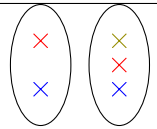	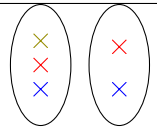	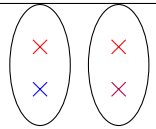	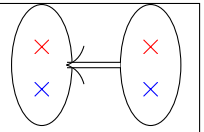
	Identity / Permutation	*Coarsening*	*Embedding*	*Outcome dropping*	*Outcome splitting*	*Abstraction reversal*
Functionality	✓	✓	✓	×	×	×
Surjectivity	✓	✓	×	×	×	×
Injectivity	✓	×	✓	×	×	×
Bijectivity	✓	×	×	×	×	×
Non-Determinism	-	-	-	-	✓	-
Macro-to-micro	-	-	-	-	-	✓

Table 2: Summary of distributional-level properties and types of abstraction. The table expresses whether enforcing a specific distributional-level property allows for the definition of certain types of abstraction.
In the first row, a color-coded example of each type of abstraction is shown; two simple sets are presented: the set of micromodel outcomes on the left, and the set of macromodel outcomes on the right. The color of the elements (crosses) in the micromodel set encodes the mapping onto elements (crosses) of the macromodel with the corresponding color.
Observe that an *identity/permutation abstraction* is allowed under the enforcement of any of the properties. A *coarsening abstraction* is possible if we do not require injectivity. An *embedding abstraction* if we do not require surjectivity. An *outcome dropping abstraction* is not allowed under any property.
Notice that the bijectivity row is just derived from surjectivity and injectivity.
Finally the last two rows and columns consider less common properties and scenarios. A *outcome splitting abstraction*, depicted by having a macrolevel cross colored purple because it is mapped by both the blue and red microlevel elements, requires a non-deterministic map. An *abstraction reversal*, represented by a double arrow going from the macrolevel to the microlevel, requires a distributional-level map going from the set of outcomes of the macromodel to the set of outcomes of the micromodel.