Generalized Measurement Models

Ricardo Silva and Richard Scheines
Center for Automated Learning and Discovery
rbas@cs.cmu.edu, scheines@andrew.cmu.edu
March 22, 2004
CMU-CALD-04-101

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Given a set of random variables, it is often the case that their associations can be explained by hidden common causes. We present a set of well-defined assumptions and a provably correct algorithm that allow us to identify some of such hidden common causes. The assumptions are fairly general and sometimes weaker than those used in practice by, for instance, econometricians, psychometricians, social scientists and in many other fields where latent variable models are important and tools such as factor analysis are applicable. The goal is automated knowledge discovery: identifying latent variables that can be used across diferent applications and causal models and throw new insights over a data generating process. Our approach is evaluated throught simulations and three real-world cases.

Keywords: Causality discovery, graphical models, latent variable models, structural equation models, data mining

1 Introduction

Latent variables are everywhere in science. Concepts such as gravitational fields, subatomic particles or various classes of antibodies are essential building blocks of models of great practical impact, and yet such entities are unobservable. Sometimes there is overwhelming evidence that hidden variables are actual physical entities, and sometimes they are useful abstractions to be added to the scientific vocabulary to make the description of Nature more tractable.

For instance, focusing in our particular interest of artificial intelligence (AI), it is hard to conceive a robot or any kind of intelligent agent seemly integrated to its environment if such agent is not able to reason with latent variables. Consider a futuristic version of Pearl, the nursing robot described in (Pineau et al., 2003). Imagine the task of autonomously attempting to diagnostic and reduce stress or depression levels of a patient, considering that someone suffering from depression will not in many cases ask for help. If the robot detects the patient is feeling too stressed, it could remotely contact healthcare professionals to come over and properly treat the patient. This would be especially useful if he or she is an elderly person living alone.

However, "stress" is not an easily describable concept: unlike "height" and "weight", there is no simple scale for it. Instead, one can measure stress through a varied set of *indicators* such as blood pressure, amount of hours slept by day, cold and sweaty hands, and so on. By using such indicators obtained from physical sensors, an agent is able to reason about a latent concept and do the proper intervention in the world. In either way, latent variables play a major role in the process of scientific modeling and discovery, and any tool that could aid the discovery of latent variables would be of great interest.

This is the goal of this paper. We introduce a machine learning algorithm to discover possible hidden common causes of a set of observed variables in a causal graphical model framework. Unlike factor analysis, there is no need to rely on arbitrary rotations of the latent space. Unlike general hill-climbing algorithms over directed acyclic graphs (DAGs) with latent variables, our approach provides an equivalence class of models that are empirically indistinguishable. Moreover, a proof of consistency of the algorithm is given on the limit of infinite data. That is, given the constraints that hold in the population over the measured variables, and a set of assumptions we make explicit in Section 3 below, the algorithm will output an equivalence class that includes the correct latent variable measurement model.

Our assumptions are described in detail. The most important assumption is that observed variables are *measures* of a set of unknown latent variables. In graphical model terminology, it means that no observed variable is an ancestor of a latent variable, but direct connections among observed variables are allowed. A stronger variation of this assumption is widely used in other latent variable discovery methods such as exploratory factor analysis. The graphical structure of the latent nodes is free to take any form: an arbitrary DAG, a DAG with other hidden common causes, cyclic graphs.

In this work, we will not discuss how to learn the structure among latent variables. Instead, we will provide an algorithm to learn a graphical structure describing which latent variables are parents of which observed variables, i.e., a *measurement model*. The procedure is an exploratory data analysis, or data mining, method to discover latent concepts that can

be useful for AI applications and in a variety of scientific models. The measurement model obtained can then be fixed such that another learning procedure is applied to search for a structure among latents, as done in Silva (2002), but such problem can be treated independently and will not be further discussed to provide a better focus on learning measurement models.

This paper is organized as follows:

- **Section 2: Related work** is a brief overview of other approaches directly or indirectly related to the task of building a measurement model from data;
- **Section 3: Problem statement and assumptions** formally defines the problem and introduces which assumptions are considered in order to provide a rigorous interpretation of our models. Such assumptions will be essential when proving the consistency of our procedure;
- **Section 4: Learning measurement models** is the main section, describing the standard algorithm for learning a representation of a set of measurement models consistent with the data. This section considers the learning problem assuming the population joint distribution is known. Later sections will treat the problem of learning with finite samples;
- **Section 5: Purification and identifiability** describes a specific class of measurement models that in practical applications will be the representation of choice due to theoretical and practical reasons;
- **Section 6: Statistical learning and practical implementations** details how to use the given algorithms when the population joint density is not known and which heuristics can be used to improve robustness to sample variability, and how how to deal with the computational complexity of this procedure;
- **Section 7: Empirical results** discusses series of experiments with simulated data and three real-world data sets, along with criteria of success;
- **Section 8: Conclusion** wraps up the contributions of this paper.

2 Related work

Arguably, the most traditional framework for discovering latent variables is through factor analysis (see, e.g., Johnson and Wichern, 2002). A number of factors is chosen based in some criterion such as the minimum number of factors that fit the data at a given level or the number that maximizes a score such as BIC. After fitting the data, usually assuming a Gaussian distribution, different transformations to the latent covariance matrix are applied in order to satisfy some criteria of simplicity. Latents are interpreted based on the magnitude of the loadings (the coefficients relating each observed variable to each latent).

This method can be quite unsatisfactory due to the underterminacy of the solution in the Gaussian case. Rotation methods used to transform the latent covariance matrix have

no formal justification. For non-Gaussian cases, variations such as independent component analysis and independent factor analysis (Attias, 1999) or tree-based component analysis (Bach and Jordan, 2003) do little to contribute to solve the problem of learning measurement models: by severely constraining latent relationships through marginal independencies or at most pairwise dependencies, the goal is to obtain good joint density estimation or to perform blind source separation, but not model interpretation or latent concept discovery.

In constrast, Zhang (2004) does provide a sound representation for measurement models for discrete observed and latent variables with a multinomial probabilistic model. The model is constrained to be a tree, and every observed variable has one and only (latent) parent and no child. Therefore no observed variable can be a child of another observed variable or a parent of a latent. To some extent, an equivalence class of graphs is described, which limits the number latents and the possible number of states each categorical latent variable can have without being empirically indistinguishable from another graph with less latents or less states per latent. Under these assumptions, the set of possible latent variable models is therefore finite. Besides being useful to model joint probability distributions, Zhang also points out that such model can be used to cluster analysis, generalizing standard one-latent approaches for clustering such as AutoClass (Cheeseman and Stutz, 1996). However, as pointed out by Zhang, this choice of representation does not guarantee that every joint distribution can be modeled well.

A related approach is given by Elidan et al. (2000) where latent variables are introduced into dense regions of a DAG learned through standard algorithms. Once one latent is introduced as the parent of a set of nodes originally strongly connected, the standard search is executed again and the process is iterated. They provide several results where this procedure is effectively able to increase the fit over a latent-free graphical model, but little is discussed about how to interpret the output. No equivalence classes are given, and all examples described in Elidan et al. (2000) and Elidan and Friedman (2001), comparing an estimated structure against a true model structure known by simulation, use as starting points graphs that are very close to the true graph. The main problem of using this approach for model interpretation and causal analysis is the lack of a description of which graphs are empirically indistinguishable.

Silva et al. (2003) provide the foundations of the work here described. In the next sections, we discuss how we generalize the previous approach and which new heuristics are applied. The present work itself is inspired by the approaches introduced in Glymour et al. (1987), where measurement models are modified based on an initial model where all latents are given, not discovered. More discussion about related work is also given in Silva et al. (2003).

3 Problem statement and assumptions

The goal of learning measurement models is identifying abstract or unmeasured concepts ("factors") that causally explain the associations measured over a set of observable random variables. The language of graphical models (Jordan, 1998), a graphical causality calculus and the concept of d-separation will be used as a formal language for our models. If the reader is not familiar with the concept of d-separation and causal models, books such as Pearl (1988,

2000) and Spirtes et al.(2000) present the definitions in full detail. The following definitions introduce the families of measurement models and graphical models of interest. We begin by our particular definition of latent variable graph.

Definition 1 (Latent variable graph) *A latent variable graph G is a graph with the following characteristics:*

1. *there are two types of nodes: observed and latent;*
2. *no observed node is an ancestor of any latent node;*
3. *each observed node is a child of (measures) at least one latent node;*
4. *there are no cycles involving an observed variable;*

The notation $G(\mathbf{O})$ will sometimes be used in this report to denote a latent variable graph G with a set of observed variables $\mathbf{O}$. The second assumption, which we call the *measurement assumption*, cannot in general be tested empirically. Nevertheless, its use is justified in several applications (e.g., Bartholomew et al., 2002). It is also the core assumption of all procedures with goals similar to factor analysis, even when it is not made explicit. See Silva et al. (2003) for more discussion on this topic. Also important, notice that it partitions the graph in two main parts, one of them composed of latent variables only. We can explore this modularization when defining a parameterization of the latent variable graph in order to avoid making unnecessary assumptions about the causal structure of the unobserved variables.

In the next section, we define which types of models our latent variable graphs can represent. We then introduce a particular useful equivalence class of models and formally state the problem of learning measurement models under our assumptions and equivalence class.

3.1 Interpretation and parameterization

We assume that a latent variable graph G is quantitatively instantiated as a semi-parametric model with the following properties:

1. G satisfies the Causal Markov condition (Spirtes et al., 2000); Pearl (2000))
2. each observed node is a linear function of its parents plus an additive error term of positive finite variance which is independent of every other error term;
3. the marginal distribution over latent variables has second moments, positive variances and all correlations less than 1.

We call such an object a *semilinear latent variable model*. If the relationships among the latent variables are also linear, that is, if each latent variable is a linear function of its parents plus additive noise, then we call it a *linear latent variable model*, an instance of a structural equation model (Bollen, 1989). For simplicity, we will assume that all variables

have zero mean. Unless otherwise specified, all latent variable models that we refer to in this report are semilinear models. Sometimes we will call the graph for a semilinear model as *semilinear latent variable graph*. A *linear latent variable graph* is defined analogously.

The linearity assumption linking the parents of an observed variable to itself is one way of constraining the classes of models represented by latent variable graphs. We call such assumption *linearity of measurement*. With arbitrary functional relationships among children and parents and arbitrary structure, any data can fit some latent variable model (Suppes and Zanotti, 1981). For instance, to be able to introduce an useful, constrained latent variable model, Zhang (2004) assumes that the latent variable graph has a tree structure, and variables are discrete. He does not assume linearity of measurement.

However, our work concerns graphical causal modeling: representing causal processes as directed graphs. Assuming the true (and unknown) processes in nature that generate our data to have the graphical structure of a tree is not very interesting in most cases, considering the bulk of applications of latent variable models in many sciences such as econometrics, social sciences and psychometrics (e.g., Bollen (1989)), all of which share many points in common with AI modeling. We prefer to allow the graphical structure over the latent nodes to be entirely unrestricted: an arbitrary DAG, a DAG with other hidden common causes, a cyclic graph, etc. Linearity of measurement might seem restrictive, but it is often explicitly designed into econometric, psychometric, and social scientific studies. Althought the linearity of measurement assumption is not sufficient to guarantee full identifiability of a graph as we will see later, it is still useful to distinguish a variety of features that only some graphs can share for a given distribution.

Notice that requiring linear direct effects from latents into observed variables can be interpreted just as a change of latent space. For instance, suppose we have the graphical model depicted in Figure 1(a), in which for simplicity we do not consider error terms. Variable η has a linear direct effect in three variables, and a nonlinear effect in the remaining three. The same model can be represented as in Figure 1(b), where the latent space is split into two latent variables with a linear measurement model. The process $\eta_1 \rightarrow \eta_2$ is a variable equivalent to η and the fact that we can break it down into two simpler hidden common causes might actually improve the interpretation of the model. The assumption about linear latent effects on observed variables is therefore weaker than it might seem in principle: it is basically a way of defining which latent variables can be considered direct causes of the observed variables.

Given the definition of latent variable model, we can now introduce a key definition:

Definition 2 (Measurement model) *Let $G(\mathbf{O})$ be a latent variable model. The submodel containing the complete set of nodes of G, and all and only those edges that point into $\mathbf{O}$, is called the measurement model of G.*

Under this context, observed variables are also called *indicators*. Therefore, the measurement model of a latent variable graph is just its subgraph when we remove all edges that might exist among latent variables. Notice that graphically the measurement model is a DAG. Also, a DAG submodel containing a subset of the observed nodes of G, their latent parents, and all and only those edges that point into these observed nodes, is called a *measurement submodel*.

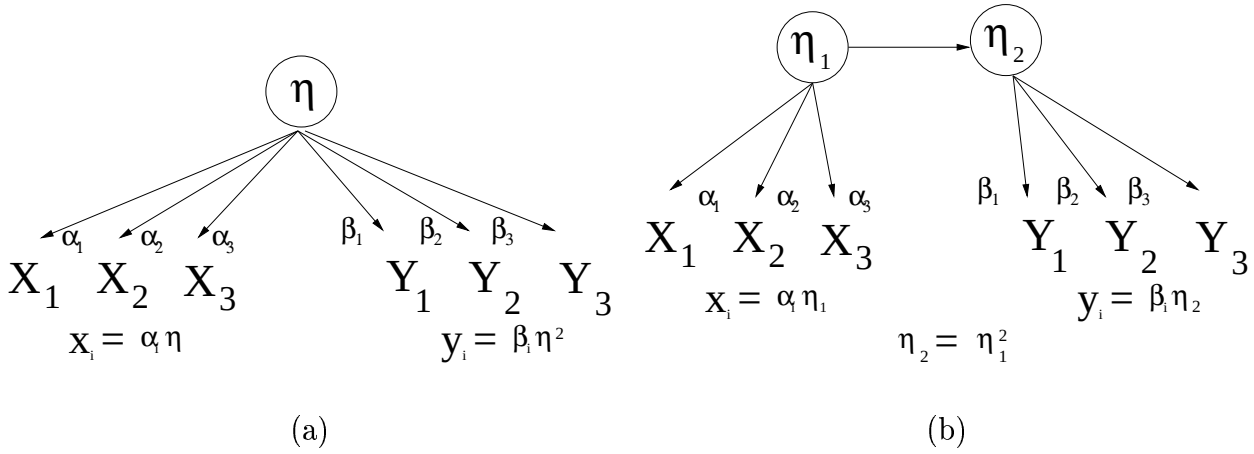

Figure 1: (a) A single latent explains the association of random variables $\{X_i, Y_i\}$ through non-linear direct effects (for simplicity, assuming no random noise is added to indicators). (b) This variable can be written as two different events associated by a directed edge, where now all direct effects on the indicators are linear.

The remaining edges of G, along with the respective latent nodes, form the complementary structure defined below:

Definition 3 (Structural model) *Let $G(\mathbf{O})$ be a latent variable model. The submodel containing only the latents of G, and all and only the edges between latents, is called the structural model of G.*

Therefore, the union of a measurement model and a structural model with the same set of latents forms a latent variable model. As hinted before, we will not discuss here how to learn a structural model. Still, these two tasks are related according to this loose formulation of our discovery problem: *assuming that the true model is a latent variable model, given a data set with variables* $\mathbf{O}$*, find the set of measurement models over* $\mathbf{O}$ *that are indistinguishable under a certain class of constraints on the observed marginal, and that will facilitate finding the Markov equivalence class, or the Partial Ancestral Graph of the structural model* (Spirtes et. al, 2000).

Later in this report, we briefly describe which other assumptions could be used to support discrete variables. However, we do need two extra assumptions for any result in this report that requires the true model G to be a linear measurement model instead of the more general semilinear one:

1. G satisfies the Faithfulness assumption (Spirtes et al., 2000), called *stability* in Pearl (2000). That is, a any conditional independence is entailed in G by the causal Markov condition if and only if it holds in the probability distribution over the variables represented in the graph

2. G is acyclic;

3.2 Tetrad classes

In order to be able to distinguish among different measurement models that might have generated our observed joint probability distribution, we need to report those models that are compatible with observed constraints of the joint. A measurement model is compatible if it entails only observed constraints:

Definition 4 (Constraint entailment) *A latent variable graph G entails a constraint if and only if the constraint holds in every distribution parameterized by the pair (P_G, Θ), where P_G is a probability distribution over the latent variables that satisfies the Markov condition for the structural model in G, and Θ the linear coefficients and error variances for the observed variables. The measurement model of G entails a constraint if and only if the constraint holds in every distribution parameterized by the pair (P_G^0, Θ), where P_G^0 is any probability distribution among the latents. (P_G and P_G^0 have also to satisfy the assumptions on latent variable models about first and second moments.)*

We are interested in a specific class of constraints. Given the covariance matrix of four random variables $\{A, B, C, D\}$, we have that zero, one or three of the following constraints may hold:

$$\begin{aligned} \sigma_{AB}\sigma_{CD} &= \sigma_{AC}\sigma_{BD} \\ \sigma_{AC}\sigma_{BD} &= \sigma_{AD}\sigma_{BC} \\ \sigma_{AB}\sigma_{CD} &= \sigma_{AD}\sigma_{BC} \end{aligned}$$

where σ_{XY} represents the covariance of X and Y.

Like conditional independence constraints, different latent variable graphs might entail different tetrad constraints. Therefore, a given set of observed tetrad constraints will restrict the set of possible latent variable graphs that are compatible with the data. We restrict our algorithm to search for measurement models that entail the observed tetrad constraints and vanishing partial correlations judged to hold in the population. Since these constraints ignore any information concerning the joint distribution besides its second moments, this might seem an unnecessary limitation. What can be learned from these constraints can be substantial, however, and attending to only the lower order moments makes the algorithm less prone to statistical errors. The empirical results discussed in Section 7 support this tradeoff. Assuming that the correct model entails all such constraints in the marginal probability distribution is a restricted version of the Faithfulness assumption discussed in (Spirtes et al., 2000).

In the particular case of linear models, tetrad constraints have a well-defined graphical implication. First, we need to introduce a few more definitions:

- in a graphical model, a *collider* on a path is a pair of consecutive directed edges on this path such that both edges point to the same node;
- a *trek* between a pair of nodes X and Y is an (undirected) path that does not contain any collider;

- a *choke point* for two sets of nodes $\mathbf{X}$ and $\mathbf{Y}$ is a node that lies on every trek between an element of $\mathbf{X}$ and an element of $\mathbf{Y}$[1].

A graphical characterization of tetrad constraints for linear graphs is known under the Faithfulness assumption:

Theorem 1 (The Tetrad Representation Theorem) *Let G be a linear graph, and let I_1, I_2, J_1, J_2 be four variables in G. Then $\sigma_{I_1 J_1}\sigma_{I_2 J_2} = \sigma_{I_1 J_2}\sigma_{I_2 J_1}$ if and only if there is a choke point between $\{I_1, I_2\}$ and $\{J_1, J_2\}$.*

Proof: See Shafer et al. (1993) and Spirtes et al. (2000). □

One can see how tetrad constraints are useful for learning the structure of latent variable graphs in the linear case: for instance, if one is given a linear latent variable graph as a starting point, this graph will entail several tetrad constraints that may hold or not among observed variables, and various modifications can be suggested to the current structure in order to make it entail more of the tetrad constraints that hold in the probability distribution and less of the constraints that do not hold. This is explored in Glymour et al. (1987) and Spirtes et al. (2000).

In this work, we explore principled approaches to reconstruct several features of the graphical structure of an unknown measurement model based on the covariance matrix of the observed variables, where no starting graph is required and the true model can be semilinear. It is an extension of the work of Silva et al. (2003) with relaxed assumptions. The principle continues to be matching entailed tetrad constraints to observed ones.

However, since there is no known graphical criterion of tetrad entailment for arbitrary semilinear latent variable graphs (or even for vanishing partial correlations, which will also be useful) such as the d-separation calculus for conditional independencies, we have to rely on the Definition 4, which is not purely graphical. It is basically a criterion of invariance with respect to the parameters of the measurement model. Invariance with respect to parameters is the key property of what is sometimes called a "structural" constraint (e.g., as in Shafer et al., 1993) and we claim nothing is lost in causal analysis by defining entailment in a causal graph where the causal features that are not of immediate interest are not parameterized (in our case, the structural model). We will show several results that hold only with probability 1 with respect to a Lebesgue measure taken over Θ, the linear coefficients and error variances in such graphs, but in practice this is no stronger than assuming the Faithfulness condition, which is known to fail for a set of parameters that has measure zero (Spirtes et al., 2000) for linear models.

In order to understand the difference between entailment by a latent variable graph and entailment by its respective measurement model, one can look at the example given in Figure 2. Latent L_2 is a choke point $(X_1, X_2) \times (Y_1, Y_2)$ and will imply the tetrad constraint $\sigma_{X_1 Y_1}\sigma_{X_2 Y_2} = \sigma_{X_1 Y_2}\sigma_{X_2 Y_1}$ independently of the model being linear or semilinear. However, the other possible tetrad $\sigma_{X_1 X_2}\sigma_{Y_1 Y_2} = \sigma_{X_1 Y_1}\sigma_{X_2 Y_2}$ will hold if and only if $\sigma^2_{L_1}\sigma_{L_2 L_3} = \sigma_{L_2}\sigma_{L_3}$,

[1] This is actually the definition of *weak choke point* as explained in Shafer, Kogan and Spirtes (1993), but it will suffice for our exposition. For the full definition, consult Shafer et al.

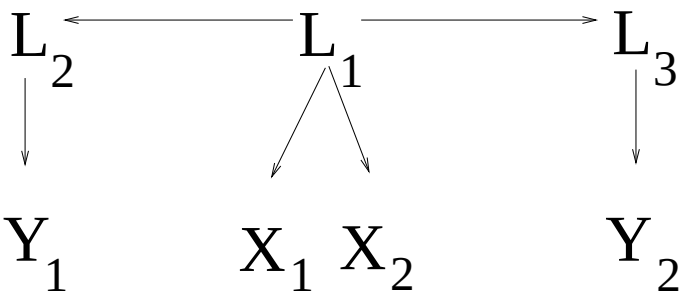

Figure 2: In this model, different choices of covariance matrix for latents $\{L_1, L_2, L_3\}$ will make three or only one tetrad constraint for variables $\{X_1, X_2, Y_1, Y_2\}$ hold.

i.e., the partial correlation of L_2 and L_3 conditioned on L_1 being zero, which is true for all probability distributions that are Markov relative to the latents in this graph, but not for an arbitrary latent covariance matrix. Therefore, this particular tetrad is not entailed by the measurement model. We need to distinguish between the two forms of entailment because we want to learn about measurement models independently of the possible structural model of the true latent graph. They will therefore form equivalence classes.

Definition 5 (Tetrad equivalence class) *A tetrad equivalence class $T(\mathcal{C})$ is a set of latent variable graphs T all of whose measurement models entail the same set of tetrad constraints and vanishing partial correlations $\mathcal{C}$ among the measured variables. An equivalence class of measurement models $M(\mathcal{C})$ for $\mathcal{C}$ is the union of the measurement models in $T(\mathcal{C})$.*

To summarize, we assume that the true model is a latent graphical model with the properties described in this Section. Under this condition, several results will be proved in the next sections. The goal is not identifying the exact true measurement model, because in general our assumptions are still strong for such task. The general problem can then be reformulated as follows: *assuming the true model is a latent variable model, given a data set with variables* $\mathbf{O}$*, return all possible measurement models over* $\mathbf{O}$ *that are indistinguishable under the class of tetrad constraints and vanishing partial correlations on the observed marginal.* We will show this is possible to some extent.

An interesting question is if it makes a difference assuming the true graph is linear instead of semilinear, i.e.: if, for some set $\mathcal{C}$ of tetrad and vanishing partial correlation constraints and a fixed latent probability distribution P_G faithful to a linear model, the set of possible linear latent models conditioned on P_G that entail $\mathcal{C}$ is strictly smaller than the set of semilinear models. The answer for this question and the reason we are interested in results for a fixed marginal distribution for the latents will be discussed in Section 4.2.

4 Learning measurement models

In this section, we introduce different criteria for learning features common to all possible latent variable graphs that generated the observed tetrad constraints and vanishing partial

correlations. Sections 4.1, 4.2 and 4.3 describe which constraints are used and which structural features can be discovered. Section 4.4 will introduce an algorithm that uses those constraints to output measurement models compatible with the observed covariance matrix.

4.1 Locally sound constraint sets

There is a specific class of sets of probabilistic constraints of practical interest which we will denominate *locally sound constraint sets*. A locally sound constraint set is a collection of constraints on the joint distribution of k observed variables, where k is a constant that does not grow with the total number of given variables. The variables used in the constraint set are called the *domain* of the constraint set. When such constraint set holds, then it should be sound (as defined in the next paragraph) to infer some particular feature of the unknown graph of interest. For instance, algorithms such as the PC SEARCH (Spirtes et al., 2000) and GES (Meek, 1997) test constraints that can refer to up to all variables in the domain, and therefore can not be considered "local" in the sense given here. However, anytime variations of the same idea such as the ANYTIME FCI algorithm of Spirtes (2000) fixes the size of the largest number of variables on which tests of conditional independence are evaluated, and therefore such constraints can be considered locally sound constraints under that context.

Let the true latent variable model G be parameterized by a pair (P_G, Θ), where P_G is a joint distribution over its latents that is Markov relative to G and Θ is the set of coefficients and error variances for the respective measurement model. We define *soundness of a constraint set* in the context of latent variable models as follows: if a constraint set establishes that certain feature should hold in G, then the probability of failure is zero with respect to a Lebesgue measure over Θ. That is, for some set of values of Θ, an inference rule using the constraint set is allowed to fail, as long as this set has measure zero. However, this constraint should hold for *every* P_G. The reason is we do not know how to quantify if the set of distributions P_G^s in which the constraint set rule erroneously applies is a "small" set in some measurable sense since P_G^s might be a result of the Markov condition applied to the unknown functional relationships among latents in G, and we do not make assumptions on how the parameterization of such functions is done. As discussed before, allowing a chance of error with probability zero is not stronger than assuming the Faithfulness condition in, say, linear DAGs.

The computational cost is not the only attractive feature of locally sound constraint sets: it is a reasonable idea not to rely on constraints with a large number of variables because statistical decisions are less reliable. The theoretical results should then be constructed with this self-imposed limitation in mind, making the theory more relevant for practical applications.

There are two main structural features of measurement models that can be discovered by our method:

- instances where two given observed variables cannot have a common parent in any latent variable graph entailing the observed tetrad constraints and vanishing partial correlations (Section 4.2);

- instances where a given observed variable cannot be an ancestor of another given

observed variable in any latent variable graph entailing the observed tetrad constraints and vanishing partial correlations (Section 4.3);

For the first situation, we will make use of constraint sets of $k = 6$. The reasons are simple: first, because of its practical use, as illustrated in the empirical examples described later in this report. Second, because they are the simplest constraint sets that can be used, as given by the following result:

Theorem 3 *There is no locally sound tetrad constraint set of domain size less than 6 for deciding if two nodes A and B do not have a common parent in a latent variable graph G, if* $\rho_{X_1 X_2 . X_3} \neq 0$ *and* $\rho_{X_1 X_2} \neq 0$ *for all* $\{X_1, X_2\}$ *in the domain of the constraint set and observed variable* X_3.

All of our non-trivial constraint sets require partial correlations to be nonzero, and it can be argued that there might be combinations of vanishing partial correlations and vanishing tetrads that could be used instead. We claim this combination is not likely to be useful. We are mostly interested in tetrad constraints that arise because of some latent choke point, and if such node exists, then no correlations and partial correlations over those variables will vanish. On the other hand, if the choke point is an observed variable, the we can use directly the observed vanishing partial correlations to infer that some nodes cannot share a parent without using tetrad constraints.

It is certainly possible to use vanishing partial correlations only in order to detect some instances where two nodes cannot have a latent common parent: the FCI algorithm described in Spirtes et al. (2000) does it even for some situations where pairs of variables are dependent conditioned in any subset of the others. In a more restricted sense, conditional independencies can be used to rule out hidden common causes among pairs of variables as suggested in Heckerman (1998) (and tested empirically in a few cases by Elidan et al., 2000). However, in this work we try to avoid conditional independencies as much as possible: their identification in finite samples becomes unreliable based on the size of the conditioning set and there are other theoretical issues on the reliability of conditional independence constraints in causal analysis even when variables are strongly independent (Robins et al., 2003). This becomes especially relevant in our case, which is biased toward models where all variables have hidden common causes. In Section 4.4 we discuss the use of partial correlations in the context of the full algorithm.

4.2 Constraints for non-overlapping parent sets

In this section, we describe a series of constraints for deciding when two nodes cannot have a common (latent) parent in a latent variable graph $G(\mathbf{O})$. We start by a constrain set rule (CS1) given as follows:

The correctness of such rule is given by the following lemma:

Lemma 3 *Let* $G(\mathbf{O})$ *be a semilinear latent variable graph. Assume* $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subseteq \mathbf{O}$ *and* $\sigma_{X_1 Y_1}\sigma_{X_2 X_3} = \sigma_{X_1 X_2}\sigma_{X_3 Y_1} = \sigma_{X_1 X_3}\sigma_{X_2 Y_1}$, $\sigma_{X_1 Y_1}\sigma_{Y_2 Y_3} = \sigma_{X_1 Y_2}\sigma_{Y_1 Y_3} = \sigma_{X_1 Y_3}\sigma_{Y_1 Y_2}$, $\sigma_{X_1 X_2}\sigma_{Y_1 Y_2} \neq \sigma_{X_1 Y_2}\sigma_{X_2 Y_1}$ *and that for all triplets* $\{A, B, C\}, \{A, B\} \subset \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}$,

For variables $\mathbf{S} = \{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subset \mathbf{O}$, if

$\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$, for all $\{A, B\} \subset \mathbf{S}, C \in \mathbf{O}$

$$\sigma_{X_1Y_1}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{X_3Y_1} = \sigma_{X_1X_3}\sigma_{X_2Y_1}$$
$$\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$$
$$\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$$

then X_1 and Y_1 cannot have a common parent in a latent variable graph

$C \in \mathbf{O}$, we have $\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$. Then X_1 and Y_1 do not have a common parent in G with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.

The proofs for this lemma and for many other results in this report are given in the Appendix. A second constraint set rule, CS2, is as follows:

For variables $\mathbf{S} = \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}$, if

$\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$, for all $\{A, B\} \subset \mathbf{S}, C \in \mathbf{O}$

$$\sigma_{X_1Y_1}\sigma_{X_2Y_2} = \sigma_{X_1Y_2}\sigma_{X_2Y_1}$$
$$\sigma_{X_2Y_1}\sigma_{Y_2Y_3} = \sigma_{X_2Y_3}\sigma_{Y_2Y_1}$$
$$\sigma_{X_1X_2}\sigma_{X_3Y_2} = \sigma_{X_1Y_2}\sigma_{X_3X_2}$$
$$\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$$

then X_1 and Y_1 cannot have a common latent parent in a linear latent variable graph

The correctness of such rule is given by the following lemma:

Lemma 4 *Let $G(\mathbf{O})$ be a linear latent variable graph. Assume $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subseteq \mathbf{O}$ and $\sigma_{X_1Y_1}\sigma_{X_2Y_2} = \sigma_{X_1Y_2}\sigma_{X_2Y_1}$, $\sigma_{X_2Y_1}\sigma_{Y_2Y_3} = \sigma_{X_2Y_3}\sigma_{Y_2Y_1}$, $\sigma_{X_1X_2}\sigma_{X_3Y_2} = \sigma_{X_1Y_2}\sigma_{X_3X_2}$, $\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$ and that for all triplets $\{A, B, C\}, \{A, B\} \subset \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}, C \in \mathbf{O}$, we have $\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$. Then X_1 and Y_1 do not have a common latent parent in G.*

A third constraint set rule, CS3, is as follows:

For variables $\mathbf{S} = \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}$, if

$\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$, for all $\{A, B\} \subset \mathbf{S}, C \in \mathbf{O}$

$$\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$$
$$\sigma_{X_1Y_2}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_2X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_2}$$
$$\sigma_{X_1Y_3}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_3X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_3}$$
$$\sigma_{X_1X_2}\sigma_{Y_2Y_3} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_3}$$

then X_1 and Y_1 cannot have a common latent parent in a linear latent variable graph

The correctness of such rule is given by the following lemma:

Lemma 5 *Let $G(\mathbf{O})$ be a linear latent variable graph. Assume $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subseteq \mathbf{O}$ and $\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$, $\sigma_{X_1Y_2}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_2X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_2}$, $\sigma_{X_1Y_3}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_3X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_3}$, $\sigma_{X_1X_2}\sigma_{Y_2Y_3} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_3}$ and that for all triplets $\{A, B, C\}, \{A, B\} \subset \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}, C \in \mathbf{O}$, we have $\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$. Then X_1 and Y_1 do not have a common latent parent in G.*

We conjecture that CS2 might also be sound for semilinear latent variable graphs. CS3, however, has an important difference with respect to the others: one can show it requires the assumption that the true graph is linear instead of semilinear, as established by the next lemma.

Lemma 6 *CS3 is not sound for semilinear latent variable graphs.*

We are now able to give an to the question presented at the end of Section 3.2. Let Σ be an observable covariance matrix, and $LT(\Sigma)$ the set of all linear latent variable graphs that entail all and only the tetrad and vanishing partial correlation constraints in Σ, and let $ST(\Sigma, \Sigma_L)$ the set of all semilinear latent variable graphs with latent covariance matrix Σ_L that entail all and only the tetrad and vanishing partial correlation constraints in Σ. We say that $G \in LT_M(\Sigma)$ if the measurement model of G is the measurement model of some graph in $LT(\Sigma)$, and a similar definition describes $ST_M(\Sigma, \Sigma_L)$. We have the following theorem as a direct result from the previous two lemmas:

Theorem 2 There is some Σ_L such that $LT_M(\Sigma)$ and $ST_M(\Sigma, \Sigma_L)$ are not equal.

Therefore, we can gain more discriminative power if we assume that the true graph is a linear latent variable graph in the class of tetrad constraints. However, we only know one rule that is provably not valid for semilinear graphs, and it is the most constrained of all, which makes the extra assumption of full linearity not particularly attractive. Still, it holds for multivariate normal distributions, a very important pratical case. More importantly from the point of view of causality discovery, the known methods for learning a structural model (Silva, 2002) require full linearity.

Before we move to the next section, it is interesting to state the following:

Proposition 1 *CS1, CS2 and CS3 are logically independent.*

In other words, the rules presented in this section are not redundant. Figure 3 depicts three situations where only one of each rule can be applied.

4.3 Discovering other features of latent variable graphs

It is possible in many cases to tell if an observed node is not an ancestor of another.

Lemma 1 *Let $G(\mathbf{O})$ be a semilinear latent variable graph. For some set $\mathbf{O}' = \{A, B, C, D\} \subseteq \mathbf{O}$, if $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$ and for all triplets $\{X, Y, Z\}$, $\{X, Y\} \subset \mathbf{O}', Z \in \mathbf{O}$, we have $\rho_{XY.Z} \neq 0$ and $\rho_{XY} \neq 0$, then no element in $X \in \mathbf{O}'$ is an ancestor of any element*

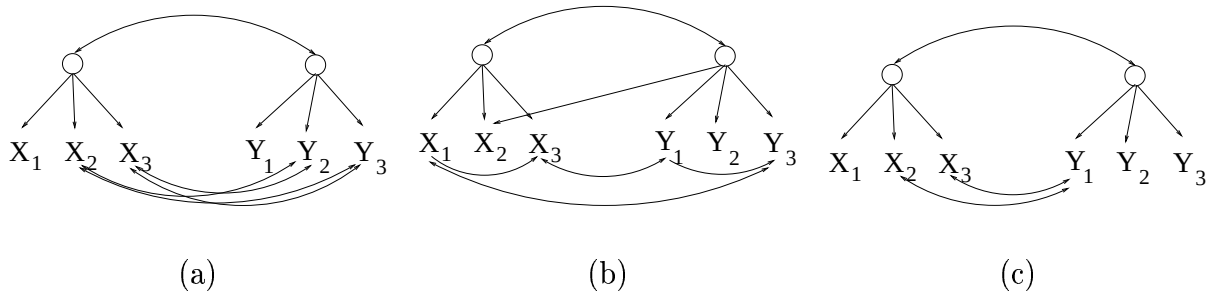

Figure 3: Three examples with two main latents and several independent latent common causes of two indicators (represented by double-directed edges). In (a), CS1 applies, but not CS2 nor CS3 (even when exchanging labels of the variables); In (b), CS2 applies, but not CS1 nor CS3. In (c), CS3 applies, but not CS1 nor CS2.

in $\mathbf{O}' \backslash X$ *in* G *with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

There are certainly other features of interest in a measurement model, such as which nodes do have a common parent, how many parents are common, and if a node is a parent of another. However, tetrad constraints are quite limited with respect to these other features: going back to the linear case and the Tetrad Representation Theorem, one can see that the lack of a choke point can be explained in many different ways, from the existence of multiple common parents to even the fact that one node is a parent of another observed node. There is very little that can be done for these other features within a tetrad equivalence class, but there are two alternatives.

The first one is to use tetrad constraints only to initialize a model by excluding common parents and possible observed ancestors where we know they should not exist. Then, proceed with a standard algorithm for learning Bayesian network structures. There are many heuristic search algorithms that can work reasonably well in practice when the starting point is close to the true graph (e.g., Elidan et al., 2000). However, no theoretical guarantees of consistency are known.

A second alternative is to select a subset of variables where there are no other major features to be discovered, i.e.: for every pair of nodes we know if they share exactly one parent or none, and observed nodes cannot be parents of another observed nodes. Under the given constraints, we can cluster all variables into groups that share a single or no common parent. This process is called *purification* and can be done entirely under a tetrad equivalence class with theoretical guarantees. This alternative will be explored in detail in Section 5.

4.4 Algorithm

We now use the information that can be obtained by tetrad constraints and vanishing partial correlations in a learning algorithm. First, one has to notice that it is difficult to design a principled score-based algorithm for learning measurement models because in general there is

Algorithm FINDPATTERN
Input: a covariance matrix Σ

1. Start with a complete graph C over the observed variables.
2. Remove edges of pairs that are marginally uncorrelated or uncorrelated conditioned on a third variable.
3. For every pair of nodes linked by an edge in C, apply successively rules CS1, and CS2/CS3, if wanted. Remove an edge between every pair corresponding to a rule that holds. Stop when it is not possible to apply any rule.
4. Let G be a graph with no edges and with nodes corresponding to observed variables.
5. For each maximal clique on C, add a new latent to G and make it a parent to all corresponding nodes in the clique.
6. For each pair of nodes (A, B), if there is no other pair (C, D) such that $\sigma_{AB}\sigma_{BD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$, add an undirected edge between A and B.
7. Return G.

Table 1: Returns the generalized measurement pattern of a latent variable graph.

no known notion of score equivalence, i.e., how to describe which structures will correspond to the same score. So far, we do not have a characterization of which measurement models will be score equivalent for any kind of score function based on the likelihood or posterior distribution of latent graphs. In this work, we will focus mainly in constraint-based search algorithms that has the property of Fisher consistency: given infinity data, the output is guaranteed to have specific properties.

Assume for now that the population covariance matrix Σ is known. Let $\mathcal{C}$ be the set of tetrad and vanishing partial correlation constraints in Σ, and $M(\mathcal{C})$ the measurement model equivalence class for $\mathcal{C}$. We define a *generalized measurement pattern*, or $GMP(\mathcal{C})$, to be a graphical object representing features of the equivalence class $M(\mathcal{C})$. The only edges allowed in a GMP are directed edges from latents to observed nodes, and undirected edges between observed nodes. Every observed node in a GMP has at least one latent parent. If two observed nodes X and Y in a $GMP(\mathcal{C})$ do not share a common latent parent, then X and Y do not share a common latent parent in any member of $M(\mathcal{C})$. If X and Y are not linked by an undirected edge in $GMP(\mathcal{C})$, then X is not an ancestor of Y in any member of $M(\mathcal{C})$.

Let FINDPATTERN be the algorithm described in Table 1. Then:

Theorem 4: *The output of* FINDPATTERN *is a generalized measurement pattern* $GMP(C)$ *with respect to the tetrad and vanishing partial correlation constraints of* Σ.

A measurement pattern also provides lower bounds on the number of underlying latent

variables: a bound can be obtained from the size of any clique in the complement of graph C as defined in Table 1.

Proposition 2 *Let C' be the complement of graph C obtained at the end of Step 3 of algorithm* FindPattern*, and let d be the size of any clique in C'. Then, there are at least d latents in the unknown latent variable graph.*

Proof: Follows directly from the fact that two neighbors in C' correspond to two observed variables that do not share a common parent, by the soundness of CS1, CS2 and CS3. Since no two elements have a common parent in the clique, there is at least one latent for each element in the clique. □

Notice we only use partial correlations with up to 1 variable in the conditioning set. In principle, the algorithm can start with a DAG obtained from a standard structure learning algorithm (again, this is how the heuristic given in Heckerman (1998) works), but we choose to ignore this extra information to avoid extra statistical decisions. Since we are assuming that observed variables are heavily connected by hidden common causes, there is little to be gained from conditional independence constraints. Also, since a DAG over the observed variables should be very dense under such assumptions, the computational cost of testing all necessary partial correlations might be prohibitive.

Even though the measurement pattern is limited in information, it is still useful for data mining purposes: it provides an indication of possible underlying latent concepts. However, a more informative graph can be obtained if we are willing to select only a subset of the variables given as input. Next section discuss what *purified patterns* are, and which desirable properties they have.

5 Purification and identifiability

In Spirtes et al. (2000) and Silva et al. (2003) we discuss a special class of measurement models called *pure measurement models.*

Definition 6 (Pure measurement model) *Let G be a latent variable graph. A pure measurement model for G is a measurement submodel of G in which each observed variable is d-separated from every other variable conditional on one of its latent parents, that is, it is a tree beneath the latents.*

Therefore, in pure measurement models, observed variables should have one and only (latent) parent. Pure measurement models are shown to be useful in Silva (2002) as a principled way of testing conditional independence among latents. Also, Silva et al. (2003) designed an algorithm for learning measurement models from data that allows one to identify every latent in the true unknown latent graph that generated the data, as well as at least three of the indicators of each latent, as long as the measurement model is pure. This is done by selecting a subset of the given observed variables. Also important, as observed by Silva et al. (2003), learning a pure measurement model of the latents is a task much more robust to

sample variability then attempting to learn the less constrained measurement pattern. We concluded that is better to learn a submodel (i.e., using only a subset of the given variables) that is more reliable than trying to learn a more complete model that is more prone to be the result of several statistical mistakes.

However, as discussed in Section 2, we do not want to make the same assumptions[2] as in Silva et al. (2003). Because of that, we will lose the ability of identifying each latent in the true unknown graph, and the latents appearing in the final output of our algorithm may also correspond to more than one latent in the true graph. As important advantages, this approach not only relies on less untestable assumptions, but also has desirable properties of anytime computation, i.e., it gives you results even when computation is interrupted before the end. The anytime properties of our algorithm will be discussed in the next section.

Consider the following algorithm for creating a pure model from a GMP found by FindPattern: make it pure by removing all nodes that have more than one latent parent or are adjacent to another observed variable. This improves what we know about the measurement model in the true graph G among the variables now remaining. For example, we know that each remaining measured variable is d-separated from all other remaining measured variables given its latent parents in the true graph G, which is crucial for discovering features of the structural model in G (Spirtes et. al, 2000, Chapter 12). Even a purified GMP is not, however, necessarily complete with respect to features of the measurement model equivalence class. Two observed variables that share a parent in the purified GMP might not share a single latent parent in the true latent variable graph. Therefore, this GMP cannot parameterize a measurement model where observed variables are linear functions of their parents.

We have not defined, however, how a GMP does or does not entail a constraint. Instead of doing so directly, we introduce the concept of an *l-interpretation* ("latent interpretation"), in order to parameterize the measurement model given in a purified GMP. The constraints entailed by the l-interpretation are a subset of the constraints entailed by the measurement model of the true latent variable graph G, a variant of I-maps (Pearl, 2000) for tetrad constraints:

Definition 7 *Given a latent variable graph $G(\mathbf{O})$ whose measurement model entails a set of constraints $\mathcal{C}$, an l-interpretation $\mathcal{I}(\mathbf{O}')$ of G for $\mathbf{O}' \subseteq \mathbf{O}$ is a latent variable graph such that the measurement model of $\mathcal{I}$ entails only constraints in $\mathcal{C}$.*

BuildPureClusters, an algorithm to create a l-interpretation for the unknown true graph, is given in Table 2. The output is a pure generalized measurement pattern, or simply a pure measurement model. It does not specify how choices in specific steps are made (e.g., which latents should be chosen in Step 2), and implementation details will be postponed to Section 6.4. It is clear that a generalized measurement pattern becomes a pure measurement model when we remove all nodes that have more than one parent and some observed neighbor. And of course there are trivial l-interpretations, such as complete graphs. However, not all l-interpretations are pure generalized measurement patterns. The following theorem states that both properties hold for the output of BuildPureClusters:

[2]Silva et al. assume that the true model has a pure submodel with at least three indicators for each latent, a much stronger assumptions

Algorithm BUILDPURECLUSTERS
Input: a covariance matrix Σ

1. $G \leftarrow$FINDPATTERN(Σ).
2. Choose a set of latents in G. Remove all other latents and all observed nodes that are not children of the remaining latents.
3. Remove all nodes that have more than one latent parent in G.
4. For all pairs of nodes linked by an undirected edge, remove one element of each pair.
5. If for some set of nodes $\{A, B, C\}$, all children of the same latent, there is a fourth node D in G such that $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$ is *not* true, remove one of these four nodes.
6. If for some pair of nodes $\{A, B\}$, both children of the same latent, and another pair of nodes $\{C, D\}$ we have $\sigma_{AC}\sigma_{BD} \neq \sigma_{AD}\sigma_{BC}$, remove one of these four nodes.
7. Remove all latents with no children.
8. Return G.

Table 2: An algorithm for obtaining a pure l-interpretation.

Theorem 5 *Let $G(\mathbf{O})$ be a latent variable graph. Then the output of* BUILDPURECLUSTERS *is a valid l-interpretation for G in the family of tetrad and vanishing partial correlation constraints and a pure generalized measurement pattern.*

One can also show that:

Lemma 10 *Let $G(\mathbf{O})$ be a latent variable graph with latent covariance matrix Σ_L. For any set $\{A, B, C, D\} = \mathbf{O}' \subseteq \mathbf{O}$, if $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BD}$ and for every set $\{X, Y\} \subset \mathbf{O}', Z \in \mathbf{O}$ we have $\rho_{XY.Z} \neq 0$ and $\rho_{XY} \neq 0$, then if A and B have a common latent parent L_1 in G, B and C have a common latent parent L_2 in G, we have $L_1 = L_2$ with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

The l-interpretation output by BUILDPURECLUSTERS can, in some circumstances, tell us a lot about the true latent variable graph. Let $\mathbf{O}'$ be the set of observed nodes in the pure measurement pattern P obtained by applying BUILDPURECLUSTERS to the covariance matrix generated by a true latent variable graph G. Let a *cluster* be a set of nodes that are children of the same latent parent in P. We can infer the following graphical features of G from P:

- Nodes in different clusters in P do not have a common parent in G
- For all pairs $\{X, Y\} \in \mathbf{O}'$, X cannot be an ancestor of Y in G;

- Let $\mathbf{C_0}$ be a cluster of P with at least 3 elements, and assume P has at least four observed variables. Then if any subset of $\mathbf{C_0}$ share a common parent in G, then it is a unique parent in G.

Thus, P forms a clustering that may be coarser than the one in G. That is, when a set of variables has a single common cause in P, then G may partition the variables in the cluster having separate latent common causes. How far could we refine the clustering in P is a topic for future research. Silva et al. (2003) describe a set of assumptions sufficient to obtain a 1-to-1 correspondence between each latent in P and each latent in G; the assumptions include the requirement that a sub-model of G has a pure measurement model with at least 3 indicators per latent.

As a final note, notice it is possible that some tetrad constraints exist in the population but are not represented in the purified output. For instance, if there is a triplet of fully connected latents $\{L_1, L_2, L_3\}$ such that $\rho_{L_2 L_3 . L_1} = 0$, then there will be one tetrad contraint with one indicator of L_2, one of L_3 and two from L_1 that, by the definition of entailment in measurement models, will not be entailed by the output graph (since the definition requires that any entailed constraint should hold for any choice of latent covariance matrix). However, this is of no importance as far as learning l-interpretations goes.

6 Statistical learning and practical implementations

There are computational and statistical issues with the theoretical specification of BuildPureClusters that have to be approached in a practical implementation. The computational cost of the procedure is apparently excessive, there are steps that are not fully specified (such as Step 2 of BuildPureClusters) and one has to define how to deal with statistical issues since only a sample covariance matrix will be available.

In the next section, we will first describe the anytime properties of the general algorithms described in Sections 4.4 and 5. We then brief explain how to adapt our method to model discrete distributions. This is followed by a discussion on statistical learning of graphical models using constraint-satisfaction and model scoring and how it is related to the problem of learning within the tetrad equivalence class. We conclude our discussion about practical implementations by describing in full detail an algorithm that can be readily implemented with heuristics that we believe to be useful in real-world applications.

6.1 Anytime properties

The algorithm in Silva et al. (2003) had the property of being able to identify all and only the latents in the true unknown measurement model, given the assumptions and the true covariance matrix. This is a stronger claim than the one given in Theorem 5, which concerns l-interpretations and and generalized measurement patterns, and might not only collapse different latents into one, but also throw away some of the latents found in the true graph.

However, in order to learn a measurement model with such guarantees, besides the stronger assumptions the algorithm of Silva et al. (2003) also required the enumeration of all maximal cliques of graph C (as described in Table 1). The number of maximal cliques

can be quite large, especially if data are noisy and many edges of C are erroneously removed or kept. Moreover, an auxiliary graph has to be built, where each node corresponds to a clique in C. A *maximum* clique has to be found in this new graph, which is a well-known NP-hard problem without any efficient approximation algorithm. In constrast, the weaker features of a l-interpretation allow a formal description on how to interpret the output when only partial information is provided.

There is a stage in FindPattern where finding all maximal cliques of a graph seems to be necessary. In fact, it is not. Identifying more cliques will only increase the chance of having a larger output by the end of the algorithm (which is good). As hinted by the freedom of choice in Step 2 of BuildPureClusters, stopping Step 5 of FindPattern after a given amount of time will not affect the result estabilished by Theorem 5. Another concern are the $O(N^6)$ loops on Step 3 of FindPattern, N being the number of variables. Still, computing this set of loops is not a fundamental limitation if there is not enough computational resources to accomplish it. One can stop Step 3 at any time at the price of losing information, but not the theoretical guarantees of Theorem 5. This is summarized by the following corollary:

Corollary 1 *Let $G(\mathbf{O})$ be a latent variable graph. Then the output of* BuildPureClusters *is a l-interpretation for G in the family of tetrad and vanishing partial correlation constraints even when rules CS1, CS2 and CS3 are applied an arbitrary number of times in* FindPattern *for any arbitrary subset of nodes and an arbitrary number of maximal cliques is found.*

In other words, one can stop the loop at Step 3 of FindPattern at any moment, as well as the one at Step 5, and still get a theoretical guarantee of consistency. There is a clear trade-off in this procedure: the longer one keeps such loops running, the more likely there will be more nodes in the final purified pattern, and the more informative it will be since nodes of different latents that in principle can be separated might not be if the proper test was not applied.

6.2 Discrete models

Although we do not perform any experiments with discrete models in this report, it is worthy mentioning that there are relatively straightforward ways of adapting the algorithms here discussed to discrete distributions. We can build on the same ideas used in discrete factor analysis. The closest framework is the *underlying variable* approach, where observed variables are assumed to be discretizations of some unobserved underlying continuous variable. The underlying variables are then indicators of another set of latents, in the same way our observed continuous variables are associated by a layer of hidden common causes. Tetrad constraints will hold for some sets of underlying variables, basically carrying on the same algorithm for another level of unobserved variables.

In order to test tetrad constraints among underlying variables, one needs to assume a probabilistic model for latent variables, where the probability mass of an underlying variable in a given range will correspond to the observed probability of a discrete variable assuming

some value. To test a set of tetrad constraints, one will need to fit a particular submodel that entails those tetrads. This is computationally expensive, since it will require numerical integration over the respective ranges that each underlying variable spam for each combination of values of the observed discrete variables. Bartholomew and Knott (1999) describe discrete factor analysis in detail.

As an alternative, one could assume that latent and observed variables are binary. In this situation, tetrad constraints will still hold (Pearl, 1988). However, in our preliminary experiments with simulated models, statistical tests of binary tetrad constraints failed to be reliable.

6.3 Statistical learning

Silva et al. (2003) argue that estimating measurement patterns from data can be a very difficult task: in simulations, the outcome was that the empirical patterns had considerably more induced latents than the synthetic models from which we sampled. The purified measurement models obtained from such patterns were quite close to the true ones, even in cases where the statistical model was wrong (i.e., assuming Gaussian distributions where data were not Gaussian). Since in this work we are allowing even less constrained measurement models, we will still focus on the estimation of pure measurement models only. Patterns will be estimated as an intermediate step, but our goal in the algorithms here described is to reliably learn pure measurement models from data.

Given a sample covariance matrix, one cannot expect that any tetrad constraints will hold exactly, but they will hold approximately. In order to test the statistical significance of such constraints, Spirtes et al. (2000) use a normal approximation for each sample tetrad difference $r_{IJ}r_{KL} - r_{IL}r_{JK}$, where r_{XY} is the sample correlation coefficient of X and Y. Mean and variances for such statistics are described in Wishart (1928). Bollen (1990) describes an asymptotically distribution free test of vanishing tetrads. The computational cost of the later test may slow down the procedure considerably, since Bollen's procedure requires the fourth moments of the data set. Concerning vanishing partial correlations, Spirtes et al. (2000) also discuss possible tests.

Therefore, in FindPattern and BuildPureClusters one could plug-in those tests in order to verify which constraints are significant. This would be a typical "constraint-satisfaction" approach for causality discovery in graphical models, in constrast to "score-based" approaches that defines a score function for a model and a set of operators that generates new candidates from the current one. There is a clear advantage in using the score-based method, in a sense that each model is scored as a whole and, therefore, uses a more robust account of the quality of a candidate graph. In constrast, a constraint-satisfaction method scores parts of a model independently.

However, it is often much easier in latent variable models to define a consistent search space for constraint-satisfaction approaches, since it is possible to control which particular constraints are going to be used. While a typical score function used in score-based search is a function of the posterior distribution of the graph given the data, for general latent variable models it is not obvious how to characterize score-equivalent models, a necessary first step to even start considering the design of consistent algorithms. Even if such equivalence is

proven, there is still a major problem of designing a computational practical algorithm for consistent estimation of the true graph. Zhang (2004) does describe score-equivalent groups of latent variable models, but does not give a prove of consistency for his hill-climbing search procedure.

In our preliminary experiments in learning pure measurement models, it is often the case that finding out which indicators should not be clustered together is a quite robust step (under the implementation we describe in the next section). However, purification is a more sensitive step: at least for a fixed p-value and using false discovery rates to control for multiplicity of tests, purification by constraint-satisfaction often throws away many more indicators than necessary when the number of variables is relative small, and does not eliminate many impurities when the number of variables is too large.

Instead, we will adopt a hybrid constraint-satisfaction/score-based approach. The first stage consists of an algorithm to cluster variables based on a modification of FindPattern. An implementation of a modified purification (Steps 5 and 6 of BuildPureClustes) is also described, which will be based on a greedy hill-climbing score-based search that first heuristically identify extra paths among indicators that are not intermediated by latents. Details of such algorithms will be covered in the next section. In the rest of this subsection, we discuss how to score a measurement model and fit its parameters to a given data set.

For our algorithms we use the Bayesian Information Criterion (BIC) as a score function under a multivariate Gaussian distribution. Althought one can claim that such representation requires strong assumptions about the joint distribution of the data, it is still largely used as the parametric family of choice for measurement models (Bollen, 1989). Such assumptions might not too harmful considering that only the second moments of the distribution are important for our algorithms. Section 7 shows a few simulation results when the true distribution is far from normal. Also, the essence of the main algorithm as discussed in Section 6.4 is not affected by the choice of probability model, althought the estimation procedures as discussed next will need to be modified if one wants to adopt a different model.

Another concern could be the choice of BIC as score function: it is known that BIC is not a consistent approximation of the posterior of a latent variable model (Rusakov and Geiger, 2004). BIC is used in our framework for its many computational advantages, especially when used with Structural EM (Friedman, 1998). More important, we will show through simulations in Section 7 that BIC is still a useful approximation to use in our problem.

6.3.1 Parameterization and scoring

For our Gaussian probability models, first we will assume that variables are centered on their means, and therefore we only have to define the implied covariance matrix of the distribution as a function of the model parameters. Let η be the vector representing the latent variables in the model and $\mathbf{y}$ the vector of observed variables. All relationships will be linear under this distribution, with additive error terms. Let ϵ represent the error terms associated with observed variables, and ζ the error terms of latent variables. We parameterize the direct effect of parents on the respective children as follows:

$$\begin{aligned} \mathbf{y} &= \Lambda_y \mathbf{y} + \Lambda_\eta \eta + \epsilon \\ \eta &= \mathbf{B}\eta + \zeta \end{aligned}$$

Matrices Λ_y and Λ_η can be very sparse: for instance, there will be a non-zero entry for $\Lambda_y^{(ij)}$ only if y_j is a parent of y_i in our model. On the other hand, matrix $\mathbf{B}$ will be a bottom triangular matrix with zeroes along the diagonal and above it, and no other zero entries. This is equivalent to a fully connected subgraph of latent variables, representing the irrelevance of the actual latent structure for our task. Matrix ζ is diagonal. Notice this is just one way of enconding an arbitrary positive semidefinite matrix.

Let $\Theta = \{\Lambda_\mathbf{y}, \Lambda_\eta, \mathbf{B}, \Phi, \Psi\}$ be the parameter set of our model, where $\Phi = E[\epsilon\epsilon^\mathbf{T}]$, the covariance matrix of the error terms of observed variables, and $\Psi = E[\zeta\zeta^\mathbf{T}]$. We will denote by $\Sigma_{\eta\eta}(\Theta)$ the implied covariance matrix of η, which can be shown to be as follows:

$$\Sigma_{\eta\eta}(\Theta) = (\mathbf{I} - \mathbf{B})^{-1}\Psi(\mathbf{I} - \mathbf{B})^{-T}$$

where $\mathbf{I}$ is the identity matrix.

Analogously, the implied covariance matrix of $\mathbf{y}$ will be given by

$$\Sigma_{yy}(\Theta) = (\mathbf{I} - \Lambda_\mathbf{y})^{-1}[\Lambda_\eta\Sigma_{\eta\eta}(\Theta){\Lambda_\eta}^T + \Phi](\mathbf{I} - \Lambda_\mathbf{y})^{-T}$$

Let $\hat{\Theta}$ be the maximum likelihood estimator of Θ. Let d be the number of parameters in Θ and let $\mathbf{S}$ be sample covariance matrix of the observed variables and N the sample size. Then the BIC score of a measurement model, up to additive constants, will be given by

$$BIC = -log|\Sigma_{yy}(\hat{\Theta})| - tr(\mathbf{S}\Sigma_{yy}^{-1}(\Theta)) - \frac{d}{2}log(N) \tag{1}$$

where $tr(\mathbf{M})$ denotes the trace of matrix $\mathbf{M}$ and $|\mathbf{M}|$ its determinant.

6.3.2 Estimation

In order to score a model, one has to find the maximum likelihood estimator of the parameters. There are a variety of methods for accomplishing this, including gradient based methods and expectation-maximization variations. However, when choosing a method one has to consider it will be used inside a computationally expensive search method to find a good fitting model. Since Structural EM (Friedman, 1998) is a natural choice for efficient hill-climbing search among latent variable models which we adopt in the algorithm described in Section 6.4, an EM estimator will be used.

We generalize the results of Rubin and Thayer (1982) in order to allow direct effects of observed variables on other observed variables. Also, we will allow correlated error terms of observed variables, i.e., Φ will be allowed to be an arbitrary symmetric positive definite matrix.

Since given Θ the distribution is jointly normal, from standard results in linear regression, the conditional distribution η given $\mathbf{y}$ can be obtained from $\Sigma_{yy}(\Theta), \Sigma_{yz}(\Theta)$ and $\Sigma_{zz}(\Theta)$, where it can be shown that

$$\Sigma_{yz}(\Theta) = (\mathbf{I} - \Lambda_y)^{-1}\Lambda_\eta\Sigma_{\eta\eta}(\Theta)$$

and the conditional distribution of η given $\mathbf{y}$ is a multivariate normal with mean $\delta\mathbf{y}$ and covariance Δ given by:

$$\begin{aligned} \delta &= \Sigma_{yy}^{-1}(\Theta)\Sigma_{yz}(\Theta) \\ \Delta &= \Sigma_{\eta\eta}(\Theta) - \Sigma_{yz}^{T}(\Theta)\delta \end{aligned}$$

Therefore, the expectation step of the algorithm is reduced to

$$\begin{aligned} E[\Sigma_{yy}|\mathbf{S},\Theta] &= \mathbf{S} \\ E[\Sigma_{yz}|\mathbf{S},\Theta] &= \mathbf{S}\delta \\ E[\Sigma_{zz}|\mathbf{S},\Theta] &= \delta^{T}\mathbf{S}\delta + \Delta \end{aligned}$$

where $\mathbf{S}$ is the sample covariance matrix.

Once a full correlation matrix of observed and latent variables is obtained, we need to estimate the parameters of the model. Non-zero non-diagonal entries in the error covariance matrix are represented by bidirected edges in the graph to indicate extra hidden common causes of a pair of variables that are independent of the other latents. We apply the algorithm of Drton and Richardson (2003) using the joint expected covariance matrix of latents and observed variables. We do not use straightforward maximum likelihood estimation, e.g., gradient-based methods or closed-formula regressions, because of the bidirected edges: unconstrained maximization might result in non-positive definite implied covariance matrices, since no constraints are enforced in the parameterization of bidirected edges. Drton and Richardson's algorithm explicitly takes into account bidirected edges, and it is guaranteed to converge to a local maximum.

6.4 Actual implementation

The main problem of applying FindPattern directly by using statistical tests of tetrad constraints is the number of false positives: accepting a rule (CS1, CS2, or CS3) as true when it does not hold in the population. One can see that might happen relatively often when there are large groups of observed variables that are pure indicators of some latent: for instance, assume there is a latent L_0 with 10 pure indicators. Consider applying CS1 to a group of six pure indicators of L_0. The first two constraints hold in the population, and so assume they are correctly identified by the statistical test. The last constraint, $\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$, should not hold in the population, but will not be rejected by the test with some probability. Since there are $10!/(6!4!) = 210$ ways of CS1 being wrongly applied due to a statistical mistake, we might get many false positives. The problem gets worse if there is a pure submodel of the true graph with many latents and many indicators per latents since the same situation can happen using indicators of not only one latent, but multiple ones, and this can be observed in simulations.

We propose here a modification to increase the robustness of FindPattern, described in detail in Table 3 – the RobustBuildPureClusters algorithm: add a first step, FindInitialSelection (Table 4), where we decide that two variables X_1 and Y_1 do not have common parents only when there are sets $\mathbf{X}$ and $\mathbf{Y}$, $X_1 \in \mathbf{X}, Y_1 \in \mathbf{Y}$ where the same holds for every pair in $\mathbf{X} \times \mathbf{Y}$. In this case, we use sets of size three, since we have to have at least six variables in order to make a local decision of nodes not sharing a same parent. Since the

number of constraints tested in this situation is much higher, there is a considerably smaller chance of an acceptance happening by statistical coincidence.

Once we generate maximal cliques from C (as defined in Table 4) in FINDINITIALSELECTION using this restricted condition, we generate an intermediate graph H where each clique from C is represented by a node in H. For each pair $\{M_i, M_j\}$ of nodes in H, we check again if there is a group $\mathbf{X}$ of nodes in the clique represented by M_i and a group $\mathbf{Y}$ of nodes in the clique represented by M_j such that every pair in $\mathbf{X} \times \mathbf{Y}$ satisfies the condition of disjoint parents. An edge between a pair M_i and M_j will be added only if such condition is satisfied. Finally, a maximal clique of nodes in H is selected and purified. The final purified model is used as a seed for the next step.

The actual test DISJOINTGROUP$(X_1, X_2, X_3, Y_1, Y_2, Y_3; \Sigma)$ used in FINDINITIALSELECTION is an application of CS1 for all pairs in $\{X_1, X_2, X_3\} \times \{Y_1, Y_2, Y_3\}$ using only nodes from this set. Also, we add an extra constraint: for every pair $\{X_i, X_j\} \subset \{X_1, X_2, X_3\}$ and every pair $\{Y_p, Y_q\} \subset \{Y_1, Y_2, Y_3\}$ we also require that $\sigma_{X_i Y_p}\sigma_{X_j Y_q} = \sigma_{X_i Y_q}\sigma_{X_j Y_p}$. The motivation is that we are looking for two sets of three indicators each from two different latent variables, where these constraints will hold. The extra redundancy will then help to reduce the number of false positives. Notice also that in DISJOINTGROUP we do not test for vanishing correlations: it is verified as part of the graphical structure of C_0. In the experiments in the next section, we actually do not make use of the vanishing partial correlations of first order, reducing the set of statistical decisions. We are implicitly assuming that no observable conditional d-separations exist in the true model.

Looking for triplets of indicators of two distinct latents is also a motivation for defining yellow edges in FINDINITIALSELECTION. If two nodes cannot be separated but also cannot be in the same cluster in a purified l-interpretation with two latents and three indicators each (which would entail the constraints in DISJOINTGROUP), then it is of no use to add both to our initial selection.

FINDMAXIMALCLIQUES can be any algorithm for finding maximal cliques. We used the one described in Bron and Kerbosch (1973). A different matter is CHOOSECLUSTERINGCLIQUE which we will describe as follows: since the number of cliques (maximal or not) in H can be large, we will be interested only in the clustering that satisfies a given optimality condition (where a *clustering* is a set of clusters, i.e., a set of sets of indicators which in the end will correspond to a pure model). Such condition should be associated with the number of indicators that remain in the model after purification. We will search for a good clustering greedily without enumerating all cliques. First, we define the *size* of a clustering $H_{candidate}$ (a set of nodes from H, which means a set of sets of nodes in $\mathbf{O}$, where each node in H is a cluster) as the number of indicators that remain according to the following elimination criteria: 1. eliminate all indicators that appear in more than one cluster inside $H_{candidate}$; 2. for each pair of indicators $\{I_1, I_2\}$ such that I_1 and I_2 belong to different clusters in $H_{candidate}$, if there is an edge $I_1 - I_2$ in C, then we remove one element $\{I_1, I_2\}$ from $H_{candidate}$ (i.e., guarantee that no pair of indicators from different clusters which were not shown to have any common latent parent will exist in $H_{candidate}$). We eliminate the one that belongs to the largest cluster, unless the smallest cluster has less than three elements to avoid extra fragmentation; 3. eliminate clusters that have only one indicator.

The optimality condition will be finding a clustering of largest size. The assumption is

Algorithm RobustBuildPureClusters
Input: Σ, a sample covariance matrix of a set of variables **O**

1. $(Selection, C, C_0) \leftarrow$FindInitialSelection$(\Sigma)$.
2. For every pair of nonadjacent nodes $\{N_1, N_2\}$ in C where at least one of them is not in *Selection* and an edge $N_1 - N_2$ exists in C_0, add a RED edge $N_1 - N_2$ to C.
3. For every pair of nodes linked by a RED edge in C, apply successively rules CS1, CS2 (and CS3, if wanted). Remove an edge between every pair corresponding to a rule that holds. Stop when it is not possible to apply any rule or till we run out of time.
4. Let H be a graph where each node corresponds to a maximal clique in C. Make H a complete graph.
5. $FinalClustering \leftarrow$ ChooseClustering(H).
6. Return RobustPurify$(FinalClustering, C, \Sigma)$.

Table 3: A modified BuildPureClusters algorithm that starts from an initial pure model and ends with another purification. See the text for the definition of ChooseClustering and the next tables for the definition of the other functions.

that a model with a large size will have a large number of indicators after purification. Our suggested heuristic to be implemented as ChooseClusteringClique is trying to find a good model using a very simple hill-climbing algorithm that starts from an arbitrary node in H and add new clusters to the current candidate according to the one that will increase its size mostly while still forming a maximal clique in H. We stop when we cannot increase the size of the candidate. This is calculated using each node in H as a starting point, and the largest candidate is returned by ChooseClusteringClique.

The next steps in RobustBuildPureClusters are basically the FindPattern of Table 1 with a final purification. The main difference is that we do not check anymore if pairs of nodes in the initial clustering given by *Selection* should be separated. The intuition explaining the usefulness of this implementation is as follows: if there is a group of latents forming a pure subgraph of the true graph with a large number of pure indicators for each latent, then the initial step should identify such group. The consecutive steps will refine this solution without the risk of splitting the large clusters of variables, which are exactly the ones most likely to produce false positive decisions with constraint sets {CS1, CS2, CS3}. RobustBuildPureClusters has the power of identifying the latents with large sets of pure indicators and refining this solution with more flexible rules, therefore generating the smaller clusters. The function ChooseClustering is identical to ChooseClusteringClique, but now we do not worry about which pairs of nodes from our new H are linked.

Notice that FindInitialSelection is very similar to the FindMeasurementPattern algorithm of Silva et al. (2003). An essential difference is that we are not concerned about finding a pure model with three indicators per latent: for instance, it might be the

Algorithm FINDINITIALSELECTION
Input: Σ, a sample covariance matrix of a set of variables $\mathbf{O}$

1. Start with a complete graph C over $\mathbf{O}$.

2. Remove edges of pairs that are marginally uncorrelated or uncorrelated conditioned on a third variable.

3. $C_0 \leftarrow C$.

4. Color every edge of C as BLUE.

5. For all edges $N_1 - N_2$ in C, if there is no other pair $\{N_3, N_4\}$ such that all three tetrads constraints hold in the covariance matrix of $\{N_1, N_2, N_3, N_4\}$, change the color of the edge $N_1 - N_2$ to GRAY.

6. For all pairs of variables $\{N_1, N_2\}$ linked by a BLUE edge in C

   If there exists a pair $\{N_3, N_4\}$ that forms a BLUE clique with N_1 in C, and a pair $\{N_5, N_6\}$ that forms a BLUE clique with N_2 in C, all six nodes form a clique in C_0 and DISJOINTGROUP$(N_1, N_3, N_4, N_2, N_5, N_6; \Sigma) = true$, then remove all edges linking elements in $\{N_1, N_3, N_4\}$ to $\{N_2, N_5, N_6\}$.

   Otherwise, if there is no node N_3 that forms a BLUE clique with $\{N_1, N_2\}$ in C, and no BLUE clique in $\{N_4, N_5, N_6\}$ such that all six nodes form a clique in C_0 and DISJOINTGROUP$(N_1, N_2, N_3, N_4, N_5, N_6; \Sigma) = true$, then change the color of the edge $N_1 - N_2$ to YELLOW.

7. Remove all GRAY and YELLOW edges from C.

8. $List_C \leftarrow$FINDMAXIMALCLIQUES(C).

9. Let H be a graph where each node corresponds to an element of $List_C$ and with no edges. Let M_i denote both a node in H and the respective set of nodes in $List_C$.

10. Add an edge $M_1 - M_2$ to H only if there exists $\{N_1, N_2, N_3\} \subseteq M_1$ and $\{N_4, N_5, N_6\} \subseteq M_2$ such that DISJOINTGROUP$(N_1, N_2, N_3, N_4, N_5, N_6; \Sigma) = true$.

11. $H_{choice} \leftarrow$CHOOSECLUSTERINGCLIQUE(H).

12. Let $H_{clusters}$ be the corresponding set of clusters, i.e., the set of sets of observed variables, where each set in $H_{clusters}$ correspond to some M_i in H_{choice}.

13. $Selection \leftarrow$ROBUSTPURIFY$(H_{clusters}, C, \Sigma)$.

14. Return $(Selection, C, C_0)$.

Table 4: Selects an initial pure model.

case that one of the latents chosen before purification will be discarded if all of its measures were removed by the RobustPurify algorithm.

To give an idea of how the later steps of refinement are essential for the sucess of RobustBuildPureClusters, we ran some simulations with models that according to the experiments analyzed in Silva et al. (2003) were the most challenging for FindMeasurementPattern: models where the largest pure subgraph of the true graph has exactly three pure indicators per latent. We generated 20 different data sets with 1,000 instances, each one sampled from a different random parameterization[3] of a pure measurement model with a fully connected latent structure, 5 latents and 3 indicators per latent. We got an average number of 1.89 latents missing with FindMeasurementPattern (standard deviation of 0.87), where "missing latents" are counted as follows: there are none of its indicators (known from the simulated graph) in the outcome; or there is one indicator, but it is clustered with indicators of other latents. In contrast, we got an average of 0.4 missing latents with RobustBuildPureClusters (standard deviation of 0.6).

FindMeasurementPattern got an average number of 0.37 indicators misplaced in a wrong cluster, where "misplaced indicators" are counted as follows: for a given cluster in the outcome of the algorithm, the misplaced indicator is the only one from a different true cluster[4]. RobustBuildPureClusters got an average of 0.1. Finally, FindMeasurementPattern got an average number of 6 missing indicators with respect to the maximum possible pure graph (standard deviation of 3), which has all 15 indicators. RobustBuildPureClusters got a much smaller average of 2.85 (standard deviation of 2.41)[5].

In contrast, given data generated from pure models with 5 indicators per latent, FindMeasurementPattern almost always get the correct number of clusters (see experiments in Silva et al., 2003). However, running RobustBuildPureClusters without FindInitialSelection resulted in an average of 1.3 clusters that were split (in half) with a high standard deviation of 1.03, indicating that it was not unlikely that in some runs of this experiment we got 3 true clusters that were split in half. Only in 25% of these 20 trials we got a perfect number of clusters. Therefore, FindInitialSelection can be of great value.

Notice that the order by which tests are applied might influence the outcome of FindInitialSelection, since if we remove an edge $X - Y$ in C at some point, then we are excluding the possibility of using some tests where X and Y are required (e.g., when searching to separate P and Q, we will not consider DisjointGroup($P, X, Y, Q, \_, \_$), for instance). Imposing such restriction reduces the overall computational cost and also reduces the number of statistical tests that are performed. Consequently, the number of statistical mistakes is also reduced. To minimize the ordering effect, an option is to run the algorithm multiple times and select the output with the highest number of nodes. The more different is the true model from a pure model, the more variety will be observed among different runs. Purification also introduces variability: if two variables are linked to the same number of impurities, we remove the first one according to the ordering given. In our experiments, we actually do not avoid tests if the required BLUE cliques do not exist as proposed by

[3]In the next section, we explain how we generate parameters for our simulated models.

[4]In only one case, we got two indicators from one cluster grouped together with two indicators from another true cluster in the same outcome cluster. This happened in both algorithms.

[5]Notice that such deviations are high because of the small number of trials and variables

Step 6 of FindInitialSelection (with the exception of those that resulted from vanishing correlations, since they introduce undesirable vanishing tetrads). This reduces the effect of variability, but different choices of ordering of variables will in many cases still result in different clusterings if the number of variables is high. That happens because the greedy ChooseClustering/ChooseClusteringClique algorithms visit many states of equal value during search, and in our implementation a choice is made based on which maximal clique was generated first. Since the order of cliques that is generated is a function of a random order of nodes in each run, we get variations of the result among runs.

For instance, in our simulation studies reported in the next section, where synthetic models have relatively large pure submodels, there is virtually no ordering effect in the output. On the other hand, with the real-world cases, there is a clear variation of output with respect to the chosen order of variables. However, multiple runs can actually *increase* the insight given by pure models, as illustrated in Section 7.2. We will hardly ever have a pure model with all variables, but by showing multiple pure models over different sets of variables, one can still have a clear picture of the generative process. Also, in the future we might want to explore the effect of avoiding tests as defined by Step 6 of FindInitialSelection.

Finally, we define RobustPurify as in Table 5. After the first two steps, clusters do not overlap and according to our constraint rules no two elements in different sets can share a common parent in the true latent variable graph (BLUE or RED edge in C) or they cannot be in a pure subgraph (GRAY edge in C).

Structural EM is applied as an heuristic for identifying impurities. Notice the use of bidirected edges, which corresponds to freeing the correlation of the error terms of two observed nodes, as an alternative to add new independent latents. Adding a new latent would require recomputing the required expected values and therefore wasting computational time. We stress that in general the BIC score it is not going to give the same result for different graphs in the same tetrad equivalence class. The goal is to throw away indicators in the purification, a much more modest goal than claiming that extra edges among indicators can be identified. Therefore, we claim that heuristics for purification have a particular pratical use in this context. Althought there is no theoretical guarantee that Structural EM will converge to the global optimum of all DAGs, nor that greedy heuristic search with a BIC score provides a consistent penalization for complexity, we find this heuristic to be very useful in practice and consistent with some of the results in Elidan and Friedman (2001) which illustrate that, given a starting point close to the true graph, heuristic hill-climbing will provide an estimate graph reasonably close to the true graph. One can therefore also interpret the tetrad constraint search that initializes the Structural EM module as a principled approach to find a good starting point that is able to converge to a pure subgraph of the original network. In the next section we evaluate how good this procedure is.

Finally, we apply the heuristic of removing nodes iteratively according to the number of impurities related to each of them. Trying to achieve some kind of optimality such as maximizing the number of pure nodes or requiring at least k indicators per latent would result in a very expensive combinatorial optimization problem. For instance, even the problem of finding a purification of a given graph that includes the maximum number of latents can be shown to be hard.

Algorithm	RobustPurify
Inputs:	$Clusters$, a set of subsets of some set $\mathbf{O}$;
	C, an undirect graph over $\mathbf{O}$;
	Σ, a sample covariance matrix of $\mathbf{O}$.

1. Remove all nodes that have appear in more than one set in $Clusters$.

2. For all pairs of nodes that belong to two different sets in $Clusters$ and are adjacent in C, remove the one from $Clusters$ that belong to the largest set unless the smallest one has less than three elements.

3. Let G a graph with a latent corresponding to each nonempty set in $Clusters$. Add all nodes in $Clusters$ as observed nodes in G. For each set $S \in Clusters$, add a new latent as the only common parent of all nodes in S. Choose an arbitrary ordering of latents and according to that ordering create a fully connected DAG over the latents.

4. Apply Structural EM to (G, Σ) using the Gaussian BIC as a score function, and some hill-climbing algorithm with operators as follows: adding a directed edge from an observed node to another in the same cluster as long as it does not create cycles; adding a bidirected edge between two observed nodes (in the same cluster or not) as long as there is no directed path between these nodes; removing edges between observed nodes.

5. Let Ord be a list of the observed nodes in G in a decreasing order of the number of non-latent adjacencies they have.

6. Sequentially remove elements from G according to the order given by Ord till no observed node has an adjacency besides its unique latent parent.

7. Remove any latents without observed children.

8. Return G.

Table 5: A score-based purification.

Proposition 3 *Let G be a latent variable graph. Then, finding a purified subgraph of G with the maximum number of latents where each latent has at least one indicator is NP-hard.*

Proof: Reduction to MAX CLIQUE. Let G' be equal to G but where observed nodes that have more than one latent parent are removed. Create a graph H with the observed nodes of G'. Add an edge for every pair of nodes that do not share a common latent parent and are d-separated in G' given the latents. Then finding a maximum clique in H is equivalent to find a pure subgraph of G with the maximum possible number of latents, each latent with at least one indicator. □

7 Empirical results

Evaluating automated knowledge discovery algorithms is often a difficult task because of the lack of a readily available gold standard by which comparisons could be made. This is especially true for unsupervised learning techniques such as clustering and causality discovery. However, we can still compare the outcome of our algorithm to theoretical models designed by experts in a field of interest, althought the models themselves might not be perfect.

Another approach we take to evaluate our algorithm is by sampling synthetic data from simulated models. By knowing the true underlying structure, and we can come up with objective measures of success. Also, it is possible to perform sensitivity analysis of our model with respect to distributional assumptions: in the next subsection, we will also evaluate how the score-based purification is sensitive to non-gaussian distributions. The second part of our empirical evaluation concerns rebuilding the measurement model of three real-world data sets according to theoretical models.

7.1 Synthetic data

The data sets we use in this section are synthetic data sets. The importance of synthetic data is the fact that we know which is the true model that generated the given samples, and therefore we can calculate precisely some measures of distance from our induced models to the true structure. We will evaluate the following features for each pure model we get with respect to a purified true graph:

- **proportion of missing latents (ML)**, the number of latents in the true graph that do not appear in the estimated pure graph, divided by the number of latents in the true graph;
- **proportion of missing indicators (MI)**, the number of indicators in the true purified graph that do not appear in the estimated pure graph, divided by the number of indicators in the true purified graph;
- **proportion of misplaced indicators (MpI)**, the number of indicators in the estimated pure graph that end up in the the wrong cluster, divided by the number of indicators in the estimated pure graph;

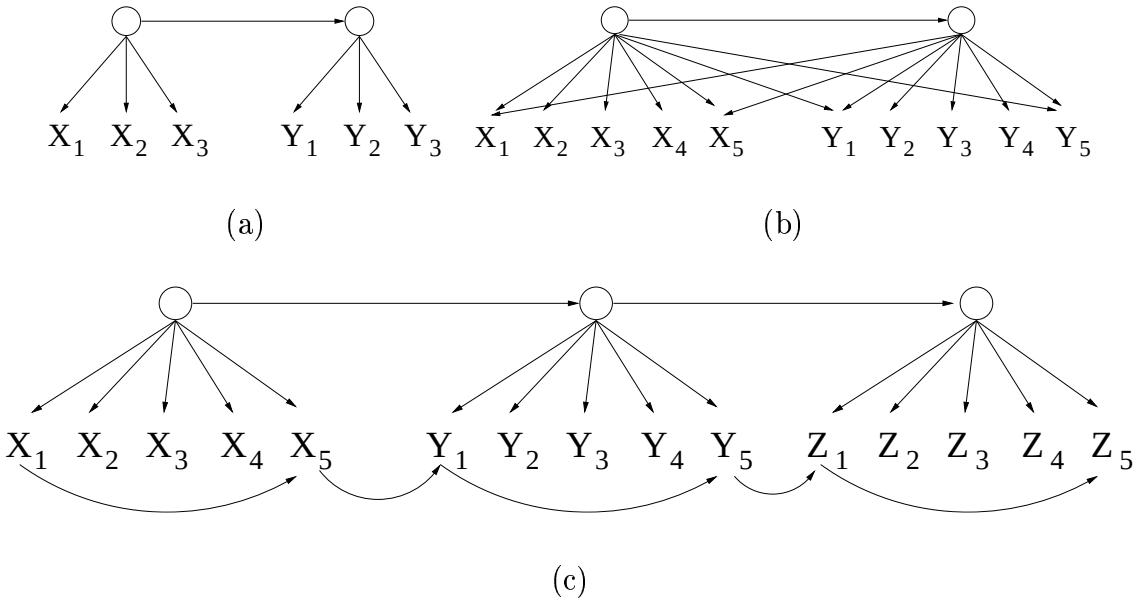

Figure 4: In (a), a pure model with 2 latents and three indicators per latent. In (b), a type of impurity model of 2 latents where 2 observed variables per each latent are children of multiple latents. The model in (c) is an example of model with three latents with a chain that turns the first and last indicators of each latent impure.

- **proportion of impurities (Im)**, the number of impurities in the estimated pure graph divided by the number of impurities in the true (non-purified) graph;
- **proportion of splits (Sp)**, the number of clusters in estimated pure graph that were split in more than one cluster, divided by the total number of clusters in the true graph.

To perform the comparison, we should indicate which latent found in the estimation corresponds to which of the original latents. The straightforward way is making the match according to the original parent of the majority of the indicators in a given estimated cluster: for example, suppose we have an estimated latent L_E. If, for instance, 70% of the measures in L_E are measures of the true latent L_2, we label L_E as L_2 in the estimated graph and calculate the statistics of comparison as described above. Ties are broken arbitrarily.

For the following results, we generated only multivariate normal indicators, with requires a linear latent structure. Samples were generated using the Tetrad IV program [6]. Values for the cofficients are then uniformly sampled from the interval $[-1.5, -0.5] \cup [0.5, 1.5]$. Variances for the exogenous nodes (i.e., latents without parents and error nodes) are uniformly sampled from the interval $[1, 3]$. The motivation for choosing such intervals is generating artificial models where the causal effects are not too big or too small. After the full parameterized

[6] Available at `http://www.phil.cmu.edu/tetrad`.

model is set, independent samples are pseudorandomly generated. The pseudorandom number generator used in the following experiments was the one used in the Java 1.4 virtual machine. The p-value used in all tests for all experiments was 0.05 for FindInitialSelection and reduced to 0.02 elsewhere, since in our simulations rules CS1, CS2, CS3 have a tendency to fire erroneously because parts of the rules concerning tetrads constraints that should not hold (e.g., $\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$ in CS1) are accepted as such when the null hypothesis is actually true (i.e., $\sigma_{X_1X_2}\sigma_{Y_1Y_2} = \sigma_{X_1Y_2}\sigma_{X_2Y_1}$).

We generate four types of models: pure models with three indicators per latent (Pure-3) as illustrated by Figure 4(a); pure models with five indicators per latent (Pure-5); models with three pure indicators per latent plus two observed variables per latent that are shared indicators (SI) of every latent (Pure-3 + SI) as illustrated by Figure 4(b); models with five indicators per latent, three of which are pure, and the other two are linked by a directed edge (Pure-3 + Chain). Also, the last indicator of each cluster is a parent of the first indicator of the consecutive cluster, as illustrated by Figure 4(c). In this way, every latent will have only three pure indicators, except the first latent in the chain, which will have four pure indicators.

Simulation results are given in Table 6. Each result is an average over 20 experiments with different parameter values randomly selected for each instance and three different sample sizes. There was a sensible improvement from trials based on samples of size 1000 compared to those with samples of size 200, but little difference was observed when comparing trials of sample size 1000 to those with sample size 10000. There was a tendency to remove more indicators than necessary in the purification procedure (i.e., high MI index). We conjecture that it can be a result of using BIC as a score function: notice that this phenomenon was less extreme with pure models. One can verify, at least empirically, that the Jacobian matrix of the parameters of the network with respect to the joint parameters (i.e., the matrix of derivatives of the entries of the covariance matrix with respect to coefficients and error variances) has full rank when the variances of the latents are scaled to a fixed value. According to Geiger et al. (1996), under this condition the BIC score might work well. A way to improve our results might be through adjusting the BIC score by using this rank instead of the number of parameters, but that might imply extra computational cost if it is not possible to find an analytical way of computing such rank. An alternative is running an iterated fit-and-purify procedure: after hill-climbing is done, remove only one variable. Repeat the process from scratch. In this way, the purification is less sensitive to the numerous edges that might have been added without necessity. However, the computational cost is also largely increased. In a future, we may try to adopt similar strategies.

We also ran experiments to detect how sensible RobustBuildPureClusters might be when the normality assumption is violated. Using the same causal structure from the Pure-3 + SI graph and multivariate Gaussian parameterization, we generated Gaussian data with additional random noise sampled from a mixture of two betas independently added to each observed variable. The mixture was defined randomly for each data set by sampling the four betas parameters from a uniform $[0, 10]$ distribution, and the mixture proportion from a uniform $[0, 1]$. We also multiplied the noise by 3, and generated 15 data sets where the average proportion of variance for each variable increased by at least 30% after adding noise.

Evaluation of estimated purified models					
	ML	**MI**	**MpI**	**Im**	**Sp**
Pure-3					
sample size 200	0.16 ± 0.14	0.22 ± 0.08	0.05 ± 0.07	$- - -$	0.0 ± 0.0
sample size 1000	0.04 ± 0.07	0.10 ± 0.07	0.0 ± 0.0	$- - -$	0.0 ± 0.0
sample size 10000	0.04 ± 0.06	0.08 ± 0.08	0.0 ± 0.0	$- - -$	0.0 ± 0.0
Pure-5					
sample size 200	0.02 ± 0.05	0.25 ± 0.08	0.0 ± 0.0	$- - -$	0.06 ± 0.07
sample size 1000	0.02 ± 0.07	0.15 ± 0.09	0.01 ± 0.02	$- - -$	0.02 ± 0.05
sample size 10000	0.0 ± 0.0	0.03 ± 0.03	0.0 ± 0.0	$- - -$	0.0 ± 0.0
Pure-3 + SI					
sample size 200	0.09 ± 0.14	0.25 ± 0.11	0.02 ± 0.04	0.15 ± 0.10	0.01 ± 0.03
sample size 1000	0.05 ± 0.07	0.14 ± 0.11	0.0 ± 0.0	0.12 ± 0.07	0.0 ± 0.0
sample size 10000	0.07 ± 0.09	0.13 ± 0.10	0.0 ± 0.0	0.08 ± 0.08	0.0 ± 0.0
Pure-3 + chain					
sample size 200	0.14 ± 0.13	0.28 ± 0.11	0.02 ± 0.04	0.21 ± 0.12	0.02 ± 0.05
sample size 1000	0.02 ± 0.05	0.11 ± 0.06	0.0 ± 0.0	0.04 ± 0.06	0.02 ± 0.05
sample size 10000	0.04 ± 0.08	0.12 ± 0.12	0.0 ± 0.0	0.05 ± 0.05	0.02 ± 0.05

Table 6: Results obtained for estimated purified graphs. Each number is an average over 20 trials, with an indication of the standard deviation over these trials.

The results were as follows for a sample size of 1000: an average of 0.07 missing latents (standard deviation: 0.09), 0.24 missing indicators (deviation of 0.10), 0.02 misplaced indicators (0.03), 0.07 impurities (.07) and 0.009 clusters that were split (for only one cluster in one of the 15 trials). There was a significant increase of missing indicators compared to the case with no non-Gaussian noise, but the algorithm still demonstrated a robust behavior against deviations from normality according to the other criteria. This is not surprising, considering the relative robustness of linear models against wrong distributional assumptions. However, a more extensive sensitive analysis still needs to be done in the future.

7.2 Real-world applications

We now discuss results obtained in three different data sets in social sciences. Even though data collected from social questionnaires may pose significant problems for exploratory data analysis since sample sizes are usually small and noisy, nevertheless they have a very useful property for our empirical evaluation purposes: questionnaires are designed to target specific latent factors (such as "stress", "job satisfaction", and so on) and a theoretical measurement model is developed by experts in the area to measure the desired latent variables, thus providing a basis for comparison with the output of our algorithm. Such variables usually include dozens of different indicators, although the chance that various observed variables are not pure measures of their theoretical latents is high. Indicators are usually discrete, but ordered in a Likert scale (Bollen, 1989) such as {"strongly disagree", "relatively disagree",

"indifferent", "relatively agree", "strongly agree"}. We will treat them as continuous variables.

Since there are theoretical models, it is easier to evaluate how our algorithm performs. The evaluation performed in the following three data sets will basically contrast the qualitative models obtained from our tetrad analysis against the theoretical models specified by previous empirical research. As an additional comment, since sample sizes are small, such data sets could hardly be reliably analysed by full score-based hill-climbing algorithms, since the number of parameters would by far exceed the number of data points. When our procedure invokes the score-based purification, the number of parameters is already dramatically reduced.

Student anxiety factors. A survey of test anxiety indicators was administered to 335 grade 12 male students in British Columbia (Bartholomew et al., 2002). The survey consisted in 20 measures on symptoms of anxiety under test conditions. A brief description of the 20 indicators is shown in Table 7.

Using factor analysis, Bartholomew et al. concluded that two factors would be the best choice for this data set throught a scree plot. If we perform a chi-square test of statistical fitness using the given covariance matrix, the factor analysis implementation in SAS reveals that just one factor is enough with a p-value of 0.09. This is also the result that minimizes BIC. Bartholomew et al. favor a better account of the variation in this data by using a more complex model.

According to Bartholomew et al., this inventory has been used in many countries with similar results. The original study identified items $\{x_2, x_8, x_9, x_{10}, x_{15}, x_{16}, x_{18}\}$ as indicators of an "emotionality" latent factor (this includes physiological symptoms such as jittery and faster heart beatting), and items $\{x_3, x_4, x_5, x_6, x_7, x_{14}, x_{17}, x_{20}\}$ as indicators of a more psychological type of anxiety labeled "worry" by Bartholomew et al. No further description is given about the remaining five variables. Bartholomew et al.'s factor analysis with oblique rotation roughly matches this model.

We ran our algorithm 10 times with different random orderings of variables and we got always the same following measurement model (x_i represents the ith item in Table 7):

1. $x_2, x_8, x_9, x_{10}, x_{11}, x_{16}, x_{18}$
2. x_3, x_5, x_7
3. x_6, x_{14}

Interestingly, the largest cluster closely corresponds to the "emotionality" factor as described by previous studies. The remaining two clusters are a split of "worry" into two subclusters with some of the original variables eliminated. Variables in the second cluster are only questions that explicitly describe "thinking" about sucess/failure (the only other question in the survey with the same characteristic was x_{17} which was eliminated). Variables x_6 and x_{14} are can be interpreted as indicating self-defeat.

To evaluate how the model given by Bartholomew et al. compares to the outcome of our algorithm, we will compare their fits according to the usual chi-square test, and also

1. Lack of confidence during tests
2. Uneasy, upset feeling
3. Thinking about grades
4. Freeze up
5. Thinking about getting through school
6. The harder I work, the more confused I get
7. Thought interfere with concentration
8. Jittery when taking tests
9. Even when prepared, get nervous
10. Uneasy before getting the test back
11. Tense during test
12. Exams bother me
13. Tense/stomach upset
14. Defeat myself during tests
15. Panicky during tests
16. Worry before important tests
17. Think about failing
18. Heart beating fast during tests
19. Can't stop worrying
20. Nervous during test, forget facts

Table 7: Indicators of test anxiety described in Bartholomew et al. (2002).

evaluate intermediate models. The two-factor model given by all theoretical indicators of "emotionality" and "worry" does not fit as a pure model (p-value of zero): the full factor analysis solution will require that some of the indicators have significant loadings in both latents, but there is no simple principled way to explain why such loadings are necessary. They may be due to direct effects of one variable on another, or due to other latent factors independent of the two conjectured. Besides that, the significance of such coefficients is tied to whatever ad-hoc rotation method is employed order to obtain "simple structure".

If we remove variables x_4, x_{17} and x_{20} from Bartholomew et al.'s model because they are not in our purified model and fit a 2-factor purified model (i.e., equivalent to our model after merging clusters 2 and 3 and latents are always fully connected), we get a p-value of 0.11, corresponding to a chi-square statistic of 65.8 (53 degrees of freedom). This model itself might be significant, but comparing to our proposed model of p-value 0.47 (chi-square of 51.2, 51 degrees of freedom), the difference of chi-squares is large enough such that the p-value of the pure two-factor model, using as alternative hypothesis our model, drops to 0.0007. This strongly suggests that our model adds a significant improvement in fit to the pure two-factor model by splitting the group $\{x3, x_5, x_6, x_7, x_{14}\}$ into two. In contrast, by randomly partitioning the first cluster into two, we did not get any significant improvement (p-value < 0.05) in 5 trials. To summarize, by dropping only 3 out of 15 previously classified variables (among a total of 20 variables), our algorithm built a measurement model not only much simpler to understand, but also giving a better fit. All without using any domain-specific prior knowledge and without relying on ad-hoc definitions of "simplicity" such as the ones used to justify factor rotation.

Well-being and spiritual coping Bongjae Lee from the University of Pittsburgh (TECHREPORT REFERENCE) organized a study to investigate religious/spiritual coping and stress in graduate students. In December of 2003, 127 Masters in Social Works students answered a questionnaire intendent to measure three main factors:

- *stress*, measured with 21 items, each using a 7-point scale (from "not all stressful" to "extremely stressful") according to situations such as: "fulfilling responsabilities both at home and at school"; "meeting with faculty"; "writing papers"; "paying monthly expenses"; "fear of failing"; "arranging childcare";
- *well-being*, measured with 20 items, each using a 4-point scale (from "rarely or none" to "most or all the time") according to indicators as: "my appetite was poor"; "I felt fearful"; "I enjoyed life" "I felt that people disliked me"; "my sleep was restless";
- *religious/spiritual coping*, measured with 20 items, each using a 4-point scale (from "not at all" to "a great deal") according to indicators such as: "I think about how my life is part of a larger spiritual force"; "I look to God (high power) for strength in crises"; "I wonder wheter God (high power) really exists"; "I pray to get my mind off of my problems";

The full questionnaire is given in the Appendix. Theoretical latents are not necessarily unidimensional, i.e., they might be partioned into an unknown set of sublatents and their

indicators might be impure, but there was no prior knowledge about which impurities might exist.

The goal of the original study was to use graphical models to quantify how spiritual coping moderates the association of stress and well-being. Our goal in this analysis is to verify if we get a clustering consistent with the theoretical measurement model (i.e., questions related to different topics will not end up in a same cluster), and analyse how questions are partioned within each theoretical cluster (i.e., how a group of questions related to the same theoretical latent ended up divided in different subclusters) using no prior knowledge.

The algorithm was applied 10 times with a different random choice of variable ordering each time. On average we got 18.2 indicators (standard deviation of 1.8). Clusters with only one variable were excluded. On average, 5.5 latents were discovered (standard deviation of 0.85). Counting only latents with at least three indicators, we had on average 4 latents (standard deviation of 0.67). In comparison, using the theoretical model as an initial model and by applying purification directly [7], i.e. without automated clustering, we obtained 15 variables (8 indicators of stress, 4 indicators of coping and 3 indicators of depression). We should not expect to do much better with an automated clustering method. This clustering is given below:

1. **Clustering C0 (p-value: 0.28):**
   STR03, STR04, STR16, STR18, STR20
   DEP09, DEP13, DEP19
   COP09, COP12, COP14, COP15

By comparing each result to the theoretical model and taking the proportion of indicators that were clustered differently from the theoretical model, we had an average percentage of 0.05 (standard deviation of 0.05). The proportionally high standard deviation is a consequence of the small percentages: in 4 out of 10 cases there was no indicator mistakenly clustered with respect to the questionnaire, in 5 out of 10 we had only one mistake, and in only one case there were two mistakes.

The three outputs with the highest number of indicators (respectively, 21, 20, 20) were also the ones with the highest number of latents:

1. **Clustering C1 (p-value: 0.31)**
   STR05, STR06, STR08, STR09
   STR12, STR15, STR21
   DEP06, DEP08, DEP17, DEP18, DEP20
   DEP15, DEP19
   COP03, COP04, COP05, COP11, COP16
   COP10, COP13

[7] In order to save time, we first applied a constraint-based purification method described in Spirtes et al. (2000) as a first step, using false discovery rates as a method for controlling to multiple hypothesis tests. Due to relatively large number of variables, this method is quite conservative and will tend to underprune the model, and therefore should not compromise the subsequent score-based purification that was applied. For instance, after the first step the model still had a p-value of zero according to a chi-square test.

2. **Clustering C2 (p-value: 0.80)**
   STR06, STR09, STR10
   STR07, STR15, STR21
   DEP08, DEP12
   DEP01, DEP07, COP06
   COP02, COP03, COP04, COP11
   COP15, COP16, COP18
   STR17, DEP36
3. **Clustering C3 (p-value: 0.52)**
   STR05, STR08, STR09, STR10
   STR12, STR21
   DEP06, DEP10, DEP17, DEP18, DEP20
   DEP08, DEP12, DEP16
   COP03, COP05, COP11, COP18
   COP10, COP13

P-values are obtained from a chi-square test assuming a multivariate Gaussian distribution. Notice that variables COP11 and COP16 are clustered together in C1, while they are separated in C2. The reason for that was due to the first stage of clustering used in our implementation, where we look for clusters of size at least three based on a more stringent version of CS1. In the case of C1, we obtained a clustering in the first stage where COP11 and COP16 were in the same cluster and, therefore, not tested again in the second stage. In the C2 run, the first stage did not include this cluster, and during the second stage there was a condition by which COP11 and COP16 were separated. Althought in principle the purification method should remove one of these two indicators in C1 if they were not meant to be clustered together, or no rule should separate COP11 and COP16 in C2 if they were not meant to be separated, with small sample sizes there is no guarantee of a reliable choice. This is also a reason why it is useful to report different l-interpretations. A similar situation happened between COP11 and COP18.

In order to evaluate how the split of theoretical clusters into subclusters was helpful, we evaluated the fit of models C1, C2 and C3 by merging subclusters of the same theoretical concept into single ones, one at a time. For C1, all three submodels have p-values less than 0.03. For C2, we first removed indicators STR17, DEP36 and COP06 to remove the effect of having a theoretically wrong clustering. The resulting p-value is roughly the same, 0.79. We then merged the stress, depression and coping pairs of clusters, one pair at a time. Merging the depression indicators result in a model of p-value 0.09, and the difference of chi-squares between the original model and the merged indicators model is not significant at a level 10^{-6}, favoring the more complex model. Merging the stress indicators results in a model with a p-value of 0.21, and the difference of chi-square statistics has a p-value not significant at a level 10^{-2}. Merging the coping indicators results in a model with a p-value 0.73, and the difference in chi-squares has now a p-value of 0.22, providing evidence that this cluster might have been spuriously divided.

When looking at the descriptions of items {COP02, COP03, COP04, COP11} there is actually a significant degree of semantical cohesion: there are all items concerning "fighting difficult situations". Items in cluster {COP15, COP16, COP18} are not as clearly grouped, but one can still argue that among all items given in the questionnaire they are the ones more directly related to "possible sources of advice" in a more general sense. Interestingly, the former cluster can then be seen as a special case of the latter. Anyway, the fact that in C1 we had COP11 and COP16 clustered together, and in C3 we had COP11 and COP18 together provide extra evidence that these clusters might have been better interpreted when merged.

Concerning merging clusters of C3, when we merge the stress clusters the resulting model has a p-value of 0.006, and the difference of chi-squares highly favours the more complex model. When the depression clusters are merged, the new p-value is 0.004, and again the more complex model is favoured. Finally, when the two clusters for coping are merged, the p-value is 0.21, but the difference of chi-squares implies a p-value of only 0.002, which still indicates lack of evidence supporting the less complex model compared to the one found by our procedure.

In conclusion, by analysing the models obtained from the automated latent discovery procedure, one can verify that they largely match theoretical expectations and, more than that, are slightly more comprehensive than the purification C0 obtained by using the original questionnaire as a starting point. C1, C2 and C3 also maintain excellent indices of fit, despite their larger complexity with respect to C0.

Single-mothers' self-efficacy and children's development: Jackson and Scheines (2004) analysed a longitudinal study on single black mothers with one child in New York City from 1996 to 1999. The goal of the study was to detect the relationship among perceived self-efficacy, mothers' employement, maternal parenting and child outcomes. Overall, there were nine factors used in this study. Three of them, age, education and income, are represented directly by one indicator each (here represented as W2moage, W2moedu and W2faminc, respectively). The other six factors are latent variables measured by a varied number of indicators:

1. *financial strain* (3 indicators, represented by W2finan1, W2finan2, W2finan3)
2. *parenting stress* (26 indicators, represented by W2paroa - W2paroz)
3. *emotional support from family* (20 indicators, represented by W2suf01 - W2suf20)
4. *emotional support from friends* (20 indicators, W2sufr01 - S2sufr20)
5. *tangible support* (i.e., more material than psychological. 4 indicators, W2ssupta - W2ssuptd)
6. *problem behaviors of child* (30 indicators, W2mneg1 - W2mneg30)

We do not reproduce the original questionnaire here due to its size. The questionnaire is based on previous work on creating scales for such latents. As before, we evaluate how our

algorithm output compares to the theoretical model. The extra difficulty here is that the distribution of the variables, which are ordinal categorical, are significantly skewed. Some of the categories are very rare, and we smoothed the original levels by collapsing values that were adjacent and represented less than 5% of the total total number of cases. Several variables ended up binary by doing this transformation, which reduces the efficiency of models based on multivariate Gaussian distributions. 1 out of the 106 variables was also removed (W2sufr04) since 98% of the points fell into one of the two possible categories. The sample size is 178, relatively large for this kind of study, but it still considerably small for exploratory data analysis.

As before, the algorithm was applied 10 times with a different random choice of variable ordering each time. On average we got 21 indicators (standard deviation of 3.35) excluding clusters with only one variable. On average, 7.3 latents were discovered (standard deviation of 1.5). Counting only latents with at least three indicators, we had on average 4.3 latents (standard deviation of 0.86). Moreover, comparing each result to the theoretical model and taking the proportion of indicators that were wrongly clustered, we had an average percentage of 0.08, with standard deviation of 0.07.

It was noticeable that the small theoretical clusterings ("financial strain" and "tangible support") did not show up in the final models, but we claim that errors of omission are less harmful than those of comission, i.e., wrong clustering. However, it was relatively unexpected that the clusterings obtained in the first stage of our implementation (i.e., the output of FindInitialSelection) were larger in number of indicators than the ones obtained at the end of process. This can be explained by the fact that the initial step is a more constrained search, and therefore less prone to overfit. Since our data set is noisier than in the previous cases, we choose to evaluate only the three largest clusters obtained from FindInitialSelection. In this case, we had an average proportion of 0.037 wrongly clustered items (standard deviation: 0.025), 4.9 clusters (deviation: 0.33), 4.6 clusters of size at least three (deviation: 0.71) and 24.2 indicators (deviation: 2.8). Notice that the clusters were less fragmented than in the previous case, i.e., we had less clusters, more indicators per clustering, and a insignificant number of clusters with less than three indicators.

The largest clusters in this situations were the following:

1. **Cluster D1 (p-value: 0.46):**
   W2sufr02 W2sufr05 W2sufr08 W2sufr13 W2sufr14 W2sufr19 W2sufr20
   W2mneg14 W2mneg15 W2mneg2 W2mneg22 W2mneg26 W2mneg28 W2mneg29
   W2suf01 W2suf05 W2suf08
   W2paro2e W2paro2j W2paro2t W2paro2w
   W2suf07 W2suf12 W2suf17

2. **Cluster D2 (p-value: 0.22):**
   W2sufr01 W2sufr08 W2sufr10 W2sufr12 W2sufr13 W2sufr14 W2sufr19 W2sufr20
   W2suf04 W2suf05 W2suf10
   W2paro2e W2paro2j W2paro2t W2paro2w
   W2paro2k W2suf12 W2suf17
   W2mneg2 W2mneg5 W2mneg12 W2mneg14 W2mneg21 W2mneg22 W2mneg26

3. **Cluster D3 (p-value: 0.29):**

   W2mneg2 W2mneg10 W2mneg22 W2mneg26 W2mneg28 W2mneg29
   W2sufr01 W2sufr05 W2sufr08 W2sufr09 W2sufr12 W2sufr13 W2sufr14 W2sufr19
   W2suf02 W2suf04 W2suf05 W2suf11 W2suf13 W2suf20
   W2paro2e W2paro2j W2paro2t W2paro2w
   W2paro2k W2suf12 W2suf17

One can see that such models largely agree with those formed from prior knowledge. However, sucess in this domain is not as interesting as in the previous two cases: unlike in the test anxiety and spiritual coping models, the covariance matrix of the latent variables has a majority number of very small entries, resulting in a considerably easier clustering by just observing marginal independencies among items.

Still, the cases where theoretical clusters were split seem to be in accordance with the data: merging the W2suf indicators in a single pure cluster in D1 will result in a model with a p-value of 0.008. Merging the W2suf variables in D2 will also result in a low p-value (0.06) even when W2paro2k is removed. Unsurprinsingly, doing a similar merging in D3 gives a model with a p-value of 0.04. This is a strong indication that W2suf12 and W2suf17 should form a cluster on their own. In fact, these two items are formulated as two very similar indicators: "members of my family come to me for emotional support" and "members of my family seek me out for companionship". No other indicator for this latent seems to fall in the same category. Why this particular pair is singled out in comparison with other indicators for this latent is a question for future studies and a simple example of how our procedure can help in understanding the latent structure of the data.

8 Discussion and future work

We introduced a novel method for automated knowledge discovery based on causal graphs with latent variables. The very general, relatively weak, assumptions by which this method has theoretical guarantees are made explicit. Although there are situations where the output of our algorithm might not be very informative, since one can expect that only a subset of the available variables forms a pure measurement model, this can also be seen as a strength of the algorithm: it does not commit itself to report features of the underlying causal model that could be explained by different mechanisms under the given set of assumptions. Assumptions are made clear instead of being buried in apparent but deceiving flexibility.

Our experiments presented evidence that such framework can be useful in practice, but as usual there are many directions where this work can be expanded:

- dependency on parametric assumptions: the tetrad equivalence class and nearly all of our causal assumptions are independent of assumptions about the probability distribution of the data. However, when it comes down to do tetrad constraint tests or scoring a measurement model for purification, probabilistic descriptions of the data are crucial. So far we have restricted ourselves to multivariate Gaussian distributions, as usual in the literature of graphical models with continuous variables. In principle, there are asymptotic distribution-free tests of tetrad constraints (Bollen, 1990) and linear

measurement errors are known to be relatively robust to the failure of the normality assumption (Fuller, 1987). However, there might be more statistically efficient ways of weakening distributional assumptions. This is also a problem for scoring DAGs as used for a heuristic purification. More flexible approaches for measurement models such as Carroll et al. (1996) could be explored in the context of discovering measurement model structure;

- finding robust score functions that will give the same score only for models in the same tetrad equivalent class. The goal is to avoid constraint-satisfaction approaches for learning graphical models and reduce the problem to hill-climbing algorithms. However, this can be a difficult task for a variety of reasons, such as the fact that multivariate Gaussian latent variable models are not curved exponential models and even approximations for them can be potentially very difficult to compute (Rusakov and Geiger, 2004). Also, just having a score equivalence class corresponding to a tetrad equivalence class is not enough to guarantee a theoretically consistent learning procedure: one would also need to prove that some non-trivial search algorithm is able to find the best scoring model;

- better treatment of discrete variables: although we hinted how discrete variables could be integrated in a tetrad equivalence class, we did not run any experiments to evaluate how this approach performs. Bartholomew and Knott (1999) survey different ways of integrating factor analysis and discrete variables that can be readily adapted. Two major problems affect discrete factor analysis: relying on underlying Gaussian random variables, which ties the structural causal assumptions to a specific probabilistic model; the computation cost of performing numerical integrations. So far no empirical studies have been performed about how such issues might affect the tetrad equivalence class here described.

- study applications of this technique for multivariate density estimation. Since density estimation in high dimensional spaces is a very difficult task, one could try a more modest goal of choosing variables that can be represented as a pure measurement model and then fit such model to the data. For instance, Zhang (2004) noticed that it is not always possible to find good fitting models for his class of pure measurement models. We therefore would search for a subset of variables that would be reasonably represented in our pure measurement model formulation;

- finding causal relationships among latent variables given a fixed measurement model for them. This was studied before in Silva (2002) with a different clustering algorithm. The natural extension is applying similar techniques with the learning algorithm developed in this work. One can then contrast the full latent variable approach against, e.g., the standard practice in social sciences of building scales, where new variables are created as deterministic functions of indicators (average, for instance) and graphical models are built using these news variables instead of introducing latents.

Acknowledgements

Research for this paper was supported by NASA NCC 2-1377 to the University of West Florida, NASA NRA A2-37143 to CMU and ONR contract N00014-03-01-0516 to the University of West Florida.

References

Attias, H. (1999). "Independent factor analysis", In *Graphical Models: foundations of neural computation*, 207-257. MIT Press.

Bach, F. and Jordan, M. (2003). "Beyond independent components: trees and clusters". *Journal of Machine Learning Research* 4, 1205-1233.

Bartholomew, D. and Knott, M. (1999). *Latent Variable Models and Factor Analysis.* 2^{nd} edition, Arnold Publishers.

Bartholomew, D.; Steele, F.; Moustaki, I. and Galbraith, J. (2002). *The Analysis and Interpretation of Multivariate Data for Social Scientists.* Chapman & Hall.

Bollen, K. (1989). *Structural Equation Models with Latent Variables.* John Wiley & Sons.

Bollen, K. (1990). "Outlier screening and a distribution-free test for vanishing tetrads". *Sociological Methods and Research* 19: 80-92.

Bollen, K. and Ting. K. (1993). "Confirmatory Tetrad analysis". In *Sociological Methodology*, p. 147-176. Blackwell Publishers.

Bron, C. and Kerbosch, J. (1973). "Algorithm 457: Finding all cliques of an undirected graph". *Communications of ACM* 16, 575-577.

Carroll, R.; Roeder, K. and Wasserman, L. (1996). "Flexible parametric measurement error models". *Biometrics* 55, 44-54.

Cattel, R. (1978). *The Scientific Use of Factor Analysis in Behavioral and Life Sciences.* Plenum Press, NY.

Cheeseman, P.; Stutz, J. (1996). "Bayesian classification (AutoClass): theory and results". In *Advances in Knowledge Discovery and Data Mining*, 153-180. AAAI Press.

Chickering, D. (2002). "Learning equivalent classes of Bayesian networks". *Journal of Machine Learning Research* 2, 445-498.

Drton, M. and Richardson, T. (2003). "Iterative conditional fitting for Gaussian ancestral grap models". Departmento of Statistics, University of Washington, Tech. Report 437.

Elidan, G. and Friedman, N. (2001). "Learning the dimensionality of hidden variables". *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence.*

Elidan, G.; Lotner, N.; Friedman, N. and Koller, D. (2000). "Discovering hidden variables: a structure-based approach." In *Neural Information Processing Systems'* 13, 479-485. MIT Press.

Friedman, N. (1998). "The Bayesian Structural EM algorithm". *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence.*

Fuller, W. (1987) *Measurement Error Models.* John Wiley & Sons.

Geiger, D., Heckerman, D., King, H. and Meek, C. (2001) "Stratified exponential families: graphical models and model selection". *Annals of Statistics* 29, 505-529.

Glymour, C.; Scheines, R.; Spirtes, P. and Kelly, K. (1987). *Discovering Causal Structure.* Academic Press.

Harman, H. (1967). *Modern Factor Analysis.* University of Chicago Press, 2nd edition.

Heckerman, D. (1998). "A tutorial on learning Bayesian networks." In *Learning Graphical Models.* MIT Press.

Heckerman, D.; Meek, C. and Cooper, G. (1999). "A Bayesian approach to causal discovery". In *Computation, Causation and Discovery*, 141-166. AAAI Press.

Hofmann, T. (2001). "Unsupervised learning by probabilistic semantic analysis". *Machine Learning* 42, p. 177-196. Kluwer Academic Publishers.

Jackson, A. and Scheines, R. (2004). "Single mother's self-efficacy, parenting in the home environment and children's development in a two-wave study". Submitted to *Social Work Research.*

Johnson, R. and Wichern, D. (2002). *Applied Multivariate Statistical Analysis*, 3^{rd} edition. Prentice Hall.

Jordan, M. (ed.) (1998). *Learning in Graphical Models.* MIT Press.

Malinowski, E. (2002). *Factor Analysis in Chemistry.* John Wiley & Sons, 3rd edition.

Meek, C. (1997). *Graphical Models: Selecting Causal and Statistical Models.* PhD Thesis on

Logic, Computation & Methodology. Carnegie Mellon University, Pittsburgh, PA.

Meek, C. (1995) "Causal inference and causal explanation with background knowledge". *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, 403 - 418. Morgan Kaufmann.

Pearl, J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Pearl, J. (2000) *Causality*. Cambridge University Press.

Pineau, J.; Montemerlo, M.; Pollack, M.; Roy, N. and Thrun, S. (2003) "Towards robotic assistants in nursing homes: challenges and results". *Robotics and Autonomous Systems* 42, 271 - 281.

Rayment, R. and Jöreskog, K. (1993). *Applied Factor Analysis in the Natural Sciences*. Cambridge University Press.

Robins, J.; Scheines, R.; Spirtes, P. and Wasserman, L. (2003). "Uniform consistency in causal inference". *Biometrika* 90, 491-515.

Rubin, D. and Thayer, D. (1982). "EM algorithms for ML factor analysis". *Psychometrika* 47, 69–76.

Rusakov, D. and Geiger, D. (2004). "Asymptotic model selection for naive Bayesian networks". In *Journal of Machine Learning Research*, to appear.

Scheines, R.; Hoijtink, H. and Boomsa, A. (1999). "Bayesian estimation and testing of structural equation models". *Psychometrika* 64, 37–52.

Shafer, G.; Kogan, A. and Spirtes, P. (1993). "Generalization of the Tetrad Representation Theorem". DIMACS Technical Report 93–68.

Silva, R. (2002) "The structure of the unobserved". Technical report CMU-CALD-02-102, Carnegie Mellon University, Pittsburgh, PA.

Silva, R.; Scheines, R.; Glymour, C. and Spirtes, P. (2003). "Learning measurement models for unobserved variables". In *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*, 543-550.

Spirtes, P; Glymour, C. and Scheines, R. (2000). *Causation, Prediction and Search*. MIT Press.

Suppes, P. and Zanotti, M. (1981). "When are probabilistic explanations possible". *Syn-*

these 48, 191-199.

Wishart, J. (1928). "Sampling errors in the theory of two factors". *British Journal of Psychology* 19, 180-187.

Zhang, N. (2004). "Hierarchical latent class models for cluster analysis". *Journal of Machine Learning Research*, to appear.

Appendix

A Proofs

Before presenting proofs for the lemmas and theorems stated in the body of this text, we will introduce the following notation. Let σ_{XY} denote the covariance of any two random variables X and Y and $\rho_{XY.Z}$ denote the partial correlation of X and Y given Z. The symbol $\{X_t\}$ will stand for a finitely indexed set of variables.

Also, let $X = \lambda_{x0}L + \sum_{i=1}^{k} \lambda_{xi}\eta_i$ and Y be random variables with zero mean, as well as $\{L, \eta_1, ..., \eta_k\}$. Let $\{\lambda_{x0}, \lambda_{x_1}, ..., \lambda_{x_k}\}$ be real coefficients. We define σ_{XYL}, the "covariance of X and Y through L", as $\sigma_{XYL} \equiv \lambda_{x0}E[LY]$.

Lemma 1 *Let $G(\mathbf{O})$ be a semilinear latent variable graph. For some set $\mathbf{O}' = \{A, B, C, D\} \subseteq \mathbf{O}$, if $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$ and for all triplets $\{X, Y, Z\}$, $\{X, Y\} \subset \mathbf{O}', Z \in \mathbf{O}$, we have $\rho_{XY.Z} \neq 0$ and $\rho_{XY} \neq 0$, then no element in $X \in \mathbf{O}'$ is an ancestor of any element in $\mathbf{O}'\backslash X$ in G with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

Proof: Since G is acyclic among observed variables, then at least one element in $\mathbf{O}'$ is not an ancestor in G of any other element in this set. By symmetry, we can assume without loss of generality that D is such node. Since the measurement model is linear, we can write A, B, C, D as linear functions of their parents:

$$\begin{aligned} A &= \textstyle\sum_p a_p A_p \\ B &= \textstyle\sum_i b_i B_i \\ C &= \textstyle\sum_j c_j C_j \\ D &= \textstyle\sum_k d_k D_k \end{aligned}$$

where on the right-hand side of each equation we have the respective parents of A, B, C and D. Such parents can be latents, another indicators or, for now, the respective error term, but each indicator has at least one latent parent besides the error term. Let $\mathbf{L}$ be the set of latent variables in G. Since each indicator is always a linear function of its parents, by composition of linear functions we have that each $X \in \mathbf{O}'$ will be a linear function of its *immediate latent ancestors*, i.e., latent ancestors L_{X_v} of X such that there is a directed path from L_{X_v} to X in G that does not contain any other element of $\mathbf{L}$. The equations above can

then be rewritten as:

$$\begin{aligned} A &= \textstyle\sum_p \lambda_{A_p} L_{A_p} \\ B &= \textstyle\sum_i \lambda_{B_i} L_{B_i} \\ C &= \textstyle\sum_j \lambda_{C_j} L_{C_j} \\ D &= \textstyle\sum_k \lambda_{D_k} L_{D_k} \end{aligned}$$

where on the right-hand side of each equation we have the respective immediate latent ancestors of A, B, C and D and λ parameters are functions of the original coefficients of the measurement model. Notice that in general the sets of immediate latent ancestors for each pair of elements in $\mathbf{O}'$ will overlap.

Since the graph is acyclic, at least one element of $\{A, B, C\}$ is not an ancestor of the other two. By symmetry, assume without loss of generality that C is such a node. Assume also C is an ancestor of D. We will prove by contradiction that this is not possible. Let L be a latent parent of C, where the edge from L into C is labeled with c, corresponding to its linear coefficient. We can rewrite the equation for C as

$$C = cL + \sum_j \lambda_{C_j} L_{C_j} \tag{2}$$

where by an abuse of notation we are keeping the same symbols λ_{C_j} and L_{C_j} to represent the other dependencies of C. Notice that it is possible that $L = L_{C_j}$ for some L_{C_j} if there is more than one directed path from L to C, but this will not be relevant for our proof. In this case, the corresponding coefficient λ is modified by subtracting c. It should be stressed that the symbol c does not appear anywhere in the polynomial corresponding to $\sum_j \lambda_{C_j} L_{C_j}$, where in this case the variables of the polynomial are the original coefficients parameterizing the measurement model and the immediate latent ancestors of C.

By another abuse of notation, rewrite A, B and D as

$$\begin{aligned} A &= c\omega_a L + \textstyle\sum_p \lambda_{A_p} L_{A_p} \\ B &= c\omega_b L + \textstyle\sum_i \lambda_{B_i} L_{B_i} \\ D &= c\omega_d L + \textstyle\sum_k \lambda_{D_k} L_{D_k} \end{aligned}$$

Each ω_v symbol is a polynomial function of all (possible) directed paths from C to $X_v \in \{A, B, D\}$, as illustrated in Figure 5. The possible corresponding $\lambda_{X_{v_t}}$ coefficient for L is adjusted in the summation by subtracting $c\omega_{X_{v_t}}$ (again, L may appear in the summation if there are directed paths from L to X_v that do not go through C). If C has more than one parent, then the expression for ω_v will appear again in some $\lambda_{X_{vt}}$. However, the symbol c *cannot* appear again into any $\lambda_{X_{vt}}$, since ω_v summarizes all possible directed paths from C to X_v. This remark will be very important later when we factorize the expression corresponding to the tetrad constraints. Notice that, by assumption, $\omega_a = \omega_b = 0$, and $\omega_d \neq 0$. We keep ω_a and ω_b in our equations to account for the next cases, where we will prove that B and A cannot be ancestors of D. The reasoning will be analogous, but the respective ωs will be nonzero.

Another important point to be emphasized is that *no term inside ω_d can appear in the expression for A and B*. That happens because D is not an ancestor of A, B or C, and at

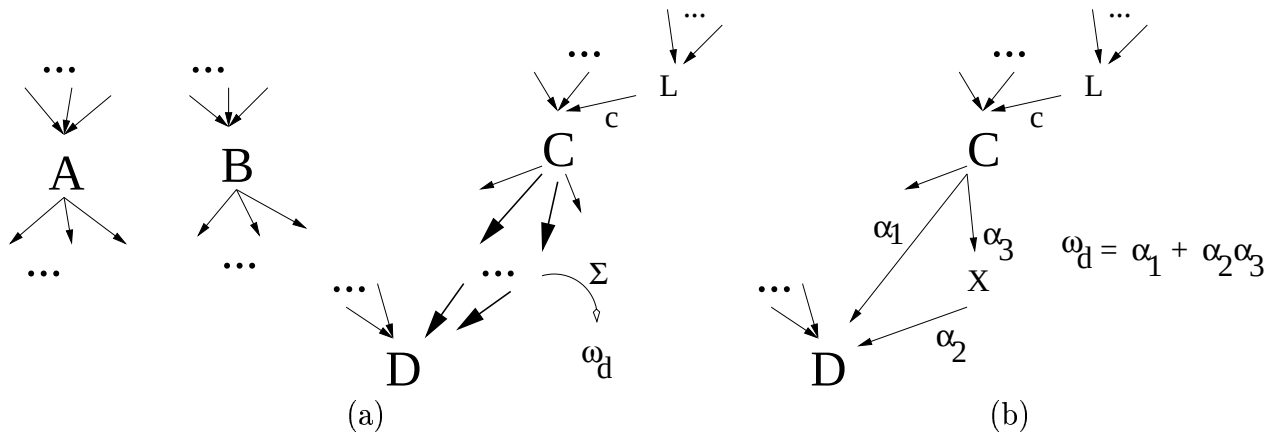

Figure 5: (a) The symbol ω_d is defined as the sum over all directed paths from C to D of the product of the labels of each edge that appears in each path. Here the larger edges represent edges in such directed paths. (b) An example: we have two directed paths from C to D. The symbol ω_d then stands for $\alpha_1 + \alpha_2\alpha_3$, where each term in this polynomial corresponds to one directed path. Notice that it is not possible to obtain any additive term that forms ω_d out of the product of some $\lambda_{A_p}, \lambda_{B_i}, \lambda_{C_j}$, since D is not an ancestor of any of them: in our example, α_1 and α_2 cannot appear in any $\lambda_{A_p}\lambda_{B_i}\lambda_{C_j}$ product (α_3 may appear if X is an ancestor of A or B).

least the edges from the parents of D to D cannot appear in any trek between any pair of elements in $\{A, B, C\}$ and every term inside ω_d contains the label of one edge between a parent of D and D. This remark will also be very important later when we will factorize the expression corresponding to the tetrad constraints.

By the definitions above, we have:

$$\begin{aligned}
\sigma_{AB} &= c^2\omega_a\omega_b\sigma_L^2 + c\omega_a\sum\lambda_{B_i}\sigma_{L_{B_i}L} + c\omega_b\sum\lambda_{A_p}\sigma_{L_{A_p}L} + \sum\sum\lambda_{A_p}\lambda_{B_i}\sigma_{L_{A_p}L_{B_i}} \\
\sigma_{CD} &= c^2\omega_d\sigma_L^2 + c\sum\lambda_{D_k}\sigma_{L_{D_k}L} + c\omega_d\sum\lambda_{C_j}\sigma_{L_{C_j}L} + \sum\sum\lambda_{C_j}\lambda_{D_k}\sigma_{L_{C_j}L_{D_k}} \\
\sigma_{AC} &= c^2\omega_a\sigma_L^2 + c\omega_a\sum\lambda_{C_j}\sigma_{L_{C_j}L} + c\sum\lambda_{A_p}\sigma_{L_{A_p}L} + \sum\sum\lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}} \\
\sigma_{BD} &= c^2\omega_b\omega_d\sigma_L^2 + c\omega_b\sum\lambda_{D_k}\sigma_{L_{D_k}L} + c\omega_d\sum\lambda_{B_i}\sigma_{L_{B_i}L} + \sum\sum\lambda_{B_i}\lambda_{D_k}\sigma_{L_{B_i}L_{D_k}}
\end{aligned}$$

Consider the polynomial identity $\sigma_{AB}\sigma_{CD} - \sigma_{AC}\sigma_{BD} = 0$ as a function of the parameters of the measurement model, i.e., the linear coefficients and error variances for the observed variables. Assume this constraint is entailed by G and its unknown latent covariance matrix. With a Lebesgue measure over the parameters, this will hold with probability 1, which follows from the fact that the solution set to non-trivial polynomial constraints has measure zero. See Meek (1997) and references within for more details. This also means that every term in this polynomial expression should vanish to zero with probability 1: i.e., the coefficients (functions of the latent covariance matrix) of every term in the polynomial should be zero. Therefore, the sum of all terms with a factor $\omega_{dt} = l_1 l_2 ... l_z$ at a given choice of exponents for each $l_1, ..., l_z$ should be zero, where ω_{dt} is some term inside the polynomial ω_d.

Before using this result, we need to identify precisely which elements of the polynomial $\sigma_{AB}\sigma_{CD} - \sigma_{AC}\sigma_{BD}$ can be factored by, say, $c^2\omega_{dt}$, for some ω_{dt}. This can include elements from any term that will explicitly show $c^2\omega_d$ when multiplying the covariance equations above among others, but we have to consider the multiplicity of the factors that compose ω_{dt}. Let $\omega_{dt} = l_1 l_2 ... l_z$. We want to factorize our tetrad constraint according to terms that contain $l_1 l_2 ... l_z$ with multiplicity 1 for each label (i.e., our terms cannot include l_1^2, for instance, or some subset of $\{l_1, ..., l_z\}$). Since C does not have some descendant X that is a common ancestor of A and D or B and D, this means that no algebraic term ω_a, ω_b or $\lambda_{A_p}, \lambda_{B_i}$ can contain some symbol in $\{l_1, ..., l_z\}$. Notice that some λ_{D_k}s will be functions of ω_{dt}: every immediate latent ancestor of C is an immediate latent ancestor of D. Therefore, for each common immediate latent ancestor parent L_q of C and D, we have that $\lambda_{D_q} = \omega_d \lambda_{C_q} + t(L_q, D) = \omega_{dt}\lambda_{C_q} + (\omega_d - \omega_{dt})\lambda_{C_q} + t(L_q, D)$, where $t(L_q, D)$ is a polynomial representing other directed paths from L_q to D that do not go through C.

For example, consider the expression $c^2\omega_a \left(\sum \lambda_{B_i}\sigma_{L_{B_i}L}\right)\left(\sum \lambda_{D_k}\sigma_{L_{D_k}L}\right)$, which is an additive term inside the product $\sigma_{AB}\sigma_{CD}$. If we group only those terms inside this expression that contain ω_{dt}, we will get $c^2\omega_a\omega_{dt}\left(\sum \lambda_{B_i}\sigma_{L_{B_i}L}\right)\left(\sum \lambda_{C_j}\sigma_{L_{C_j}L}\right)$ where the index j runs over the same latent ancestors as in (2). As discussed before, no factor of ω_{dt} can be a factor of any term in λ_{B_i}. The same holds for ω_a. Therefore, the multiplicity of each $l_1, ..., l_z$ in this term is exactly 1.

When one writes down the algebraic expression for $\sigma_{AB}\sigma_{CD} - \sigma_{AC}\sigma_{BD}$ as functions of λs, c, $\omega_a, \omega_b, \omega_{dt}$, the terms

$c^2\omega_{dt}[\sigma_L^2 \sum\sum \lambda_{A_p}\lambda_{B_i}\sigma_{L_{A_p}L_{B_i}} + \omega_a\omega_b\sigma_L^2 \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}} + \omega_a \sum \lambda_{B_i}\sigma_{L_{B_i}L} \sum \lambda_{C_j}\sigma_{L_{C_j}L} +$
$\omega_b \sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum \lambda_{C_j}\sigma_{L_{C_j}L}] -$
$c^2\omega_{dt}[\omega_b\sigma_L^2 \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}} + \omega_a\sigma_L^2 \sum\sum \lambda_{B_j}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}} + \omega_a\omega_b \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum \lambda_{C_j}\sigma_{L_{C_j}L} +$
$\sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum \lambda_{B_i}\sigma_{L_{B_i}L}]$

will be the *only* ones that can be factorized by $c^2\omega_{dt}$, where the power of c in such terms is 2, and the multiplicity of each $l_1, ..., l_z$ is 1. Since this has to be identically zero and $\omega_{dt} \neq 0$, we have the following relation:

$$f_1(G) = f_2(G) \tag{3}$$

where

$f_1(G) = c^2[\sigma_L^2 \sum\sum \lambda_{A_p}\lambda_{B_i}\sigma_{L_{A_p}L_{B_i}} + \omega_a\omega_b\sigma_L^2 \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}} + \omega_a \sum \lambda_{B_i}\sigma_{L_{B_i}L} \sum \lambda_{C_j}\sigma_{L_{C_j}L} +$
$\omega_b \sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum \lambda_{C_j}\sigma_{L_{C_j}L}]$

$f_2(G) = c^2[\omega_b\sigma_L^2 \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}} + \omega_a\sigma_L^2 \sum\sum \lambda_{B_j}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}} + \omega_a\omega_b \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum \lambda_{C_j}\sigma_{L_{C_j}L} +$
$\sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum \lambda_{B_i}\sigma_{L_{B_i}L}]$

Similarly, when we factorize terms that include $c\omega_{dt}$, where the respective powers of $c, l_1, ..., l_z$ in the term have to be 1, we get the following expression as an additive term of

$\sigma_{AB}\sigma_{CD} - \sigma_{AC}\sigma_{BD}$:

$c\omega_{dt}[\omega_a \sum \lambda_{B_i}\sigma_{L_{B_i}L} \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}} + \omega_b \sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}} +$
$2\sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum\sum \lambda_{A_p}\lambda_{B_i}\sigma_{L_{A_p}L_{B_i}}] -$
$c\omega_{dt}[\omega_a \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum\sum \lambda_{B_i}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}} + \sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum\sum \lambda_{B_i}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}} +$
$\omega_b \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}} + \sum \lambda_{B_i}\sigma_{L_{B_i}L} \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}}]$

for which we have:

$$g_1(G) = g_2(G) \tag{4}$$

where

$g_1(G) = c[\omega_a \sum \lambda_{B_i}\sigma_{L_{B_i}L} \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}} + \omega_b \sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}} +$
$2\sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum\sum \lambda_{A_p}\lambda_{B_i}\sigma_{L_{A_p}L_{B_i}}]$

$g_2(G) = c[\omega_a \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum\sum \lambda_{B_i}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}} + \sum \lambda_{A_p}\sigma_{L_{A_p}L} \sum\sum \lambda_{B_i}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}} +$
$\omega_b \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}} + \sum \lambda_{B_i}\sigma_{L_{B_i}L} \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}}]$

Finally, we look at terms multiplying ω_{dt} without c, which will result in:

$$h_1(G) = h_2(G) \tag{5}$$

where

$$h_1(G) = \sum\sum \lambda_{A_p}\lambda_{B_i}\sigma_{L_{A_p}L_{B_i}} \sum\sum \lambda_{C_j}\lambda_{C_{j'}}\sigma_{L_{C_j}L_{C_{j'}}}$$

$$h_2(G) = \sum\sum \lambda_{A_p}\lambda_{C_j}\sigma_{L_{A_p}L_{C_j}} \sum\sum \lambda_{B_i}\lambda_{C_j}\sigma_{L_{B_i}L_{C_j}}$$

Writing down the full expression for $\sigma_{AC}\sigma_{BC}$ and $\sigma_C^2\sigma_{AB}$ will result in:

$$\sigma_{AC}\sigma_{BC} = P(G) + f_2(G) + g_2(G) + h_2(G) \tag{6}$$

$$\sigma_C^2\sigma_{AB} = P(G) + f_1(G) + g_1(G) + h_1(G) \tag{7}$$

where

$$\begin{aligned} P(G) \;=\; & c^4\omega_a\omega_b(\sigma_L^2)^2 + c^3\omega_a\omega_b\sigma_L^2 \sum \lambda_{C_j}\sigma_{L_{C_j}L} + c^3\omega_a\sigma_L^2 \sum \lambda_{B_i}\sigma_{L_{B_i}L} + \\ & c^3\omega_a\omega_b\sigma_L^2 \sum \lambda_{C_j}\sigma_{L_{C_j}L} + c^2\omega_a \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum \lambda_{B_i}\sigma_{L_{B_i}L} + \\ & c^3\omega_b\sigma_L^2 \sum \lambda_{A_p}\sigma_{L_{A_p}L} + c^2\omega_b \sum \lambda_{C_j}\sigma_{L_{C_j}L} \sum \lambda_{A_p}\sigma_{L_{A_p}L} \end{aligned}$$

By (3), (4), (5), (6) and (7), we have:

$$\sigma_{AC}\sigma_{BC} = \sigma_C^2\sigma_{AB} \Rightarrow \sigma_{AB} - \sigma_{AC}\sigma_{BC}(\sigma_C^2)^{-1} = 0 \Rightarrow \rho_{AB.C} = 0$$

Contradiction. Therefore, C cannot be an ancestor of D, and more generally, of any element in $\mathbf{O}'\backslash C$.

Assume without loss of generality that B is not an ancestor of A. C is not an ancestor of any element in $\mathbf{O}'\backslash C$. If B does not have a descendant that is a common ancestor of C and D, then by analogy with the (C, D) case (where now more than one ω element will be nonzero as hinted before, since we have to consider the possibility of B being an ancestor of both C and D), B cannot be an ancestor of C nor D.

Assume then that B has a descendant X that is a common ancestor of C and D, where $X \neq C$ and $X \neq D$, since C is not an ancestor of D and vice-versa. Notice also that X is not an ancestor of A, since B is not an ancestor of A. Relations such as Equation 3 might not hold, since we might be equating terms that have different exponents for symbols in $\{l_1, ..., l_z\}$. However, since now we have an observed intermediate term X, we can make use of its error variance parameter ζ_X corresponding to the error term ϵ_X.

No term in σ_{AB} can have ζ_X, since ϵ_X is independent of both A and B. There is at least one term in σ_{CD} that contains ζ_X as a factor. There is no term in σ_{AC} that contains ζ_X as a factor, since ϵ_X is independent of A. There is no term in σ_{BD} that contains ζ_X as a factor, since ϵ_X is independent of B. Therefore, in $\sigma_{AB}\sigma_{CD}$ we have at least one term that has ζ_X, while no term in $\sigma_{AC}\sigma_{BD}$ contains such term. That requires some parameters or the variance of some latent ancestor of B to be zero, which is a contradiction.

Therefore, B is not an ancestor of any element in $\mathbf{O}'\backslash B$. In a completely analogous way, one can show that A is not an ancestor of any element in $\mathbf{O}'\backslash A$. □

Lemma 2 *Let $G(\mathbf{O})$ be a semilinear latent variable graph. Let $\{A, B, C, D\} \subset \mathbf{O}$ such that A is not an ancestor of B, C or D in G and A has a parent L in G, and no element of the covariance matrix of A, B, C and D is zero. If $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$, then $\sigma_{ACL} = \sigma_{ADL} = 0$ or $\sigma_{ACL}/\sigma_{ADL} = \sigma_{AC}/\sigma_{AD} = \sigma_{BC}/\sigma_{BD}$ with probability 1 with respect to a Lebesgue measure over the coefficient parameters.*

Proof: Since G is a linear latent variable graph, we can express A, B, C and D as linear functions of their parents as follows:

$$\begin{aligned} A &= aL + \textstyle\sum_p a_p A_p \\ B &= \textstyle\sum_i b_i B_i \\ C &= \textstyle\sum_j c_j C_j \\ D &= \textstyle\sum_k d_k D_k \end{aligned}$$

where on the right-hand side of each equation the uppercase symbols denote the respective parents of each variable on the left side, error terms included.

Given the assumptions, we have:

$$\begin{aligned} &\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC} && \Rightarrow \\ &E[a\textstyle\sum_j c_j LC_j + \sum_p \sum_j a_p c_j A_p C_j]\sigma_{BD} = E[a\sum_k d_k LD_k + \sum_p \sum_k a_p d_k A_p D_k]\sigma_{BC} && \Rightarrow \\ &a(\textstyle\sum_j c_j \sigma_{LC_j})\sigma_{BD} + \sum_p \sum_j a_p c_j \sigma_{A_p C_j}\sigma_{BD} = a(\sum_k d_k \sigma_{LD_k})\sigma_{BC} + \sum_p \sum_k a_p d_k \sigma_{A_p D_k}\sigma_{BC} && \Rightarrow \\ &a[(\textstyle\sum_j c_j \sigma_{LC_j})\sigma_{BD} - (\sum_k d_k \sigma_{LD_k})\sigma_{BC})] + [\sum_p \sum_j a_p c_j \sigma_{A_p C_j}\sigma_{BD} - \sum_p \sum_k a_p d_k \sigma_{A_p D_k}\sigma_{BC}] = 0 \end{aligned}$$

Since A is not an ancestor of B, C or D, there is no trek among elements of $\{B, C, D\}$ containing both L and A, and therefore the symbol a cannot appear in $\sum_p \sum_j a_p c_j \sigma_{A_p C_j} \sigma_{BD} - \sum_p \sum_k a_p d_k \sigma_{A_p D_k} \sigma_{BC}$ when we expand each covariance as a function of the parameters of G. Therefore, since this polynomial is identically zero, we have to have the coefficient for a equal to zero, which implies:

$$a(\sum_j c_j \sigma_{LC_j})\sigma_{BD} = a(\sum_k d_k \sigma_{LD_k})\sigma_{BC} \equiv \sigma_{ACL}\sigma_{BD} = \sigma_{ADL}\sigma_{BC}$$

Since no element in Σ_{ABCD} is zero, then $\sigma_{ACL} = 0 \Leftrightarrow \sigma_{ADL} = 0$. If $\sigma_{ACL} \neq 0$, then $\sigma_{ACL}/\sigma_{ADL} = \sigma_{AC}/\sigma_{AD} = \sigma_{BC}/\sigma_{BD}$. □

Lemma 3 *Let $G(\mathbf{O})$ be a semilinear latent variable graph. Assume $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subseteq \mathbf{O}$ and $\sigma_{X_1Y_1}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{X_3Y_1} = \sigma_{X_1X_3}\sigma_{X_2Y_1}$, $\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$, $\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$ and that for all triplets $\{A, B, C\}, \{A, B\} \subset \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}$, $C \in \mathbf{O}$, we have $\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$. Then X_1 and Y_1 do not have a common parent in G with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

Proof: Suppose X_1 and Y_1 have a common parent L in G. Let $X_1 = aL + \sum_p a_p A_p$ and $Y_1 = bL + \sum_i b_i B_i$, where each A_p, B_i are parents in G of X_1 and Y_1, respectively.

By Lemma 1 and the given constraints, an element of $\{X_1, Y_1\}$ cannot be an ancestor of the other, and neither can be an ancestor in G of any element in $\{X_2, X_3, Y_2, Y_3\}$. By definition, $\sigma_{X_1VL} = (a/b)\sigma_{Y_1VL}$ for some variable V, and therefore $\sigma_{X_1VL} = 0 \Leftrightarrow \sigma_{Y_1VL} = 0$. Assume $\sigma_{Y_1X_2L} = \sigma_{X_1X_2L} = 0$. Since it is given that $\sigma_{X_1Y_1}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_1X_3}$, by Lemma 2 we have $\sigma_{X_1Y_1L} = \sigma_{X_1X_2L} = 0$. Since $\sigma_{X_1Y_1L} = ab\sigma_L^2 + K$, where no term in K contains the factor ab, then if $\sigma_{X_1Y_1L} = 0$, with probability 1 $ab\sigma_L^2 = 0 \Rightarrow \sigma_L^2 = 0$, which is a contradiction of the assumptions. By repeating the argument, no element in $\{\sigma_{X_1X_2L}, \sigma_{X_1X_3L}, \sigma_{Y_1X_2L}, \sigma_{Y_1X_3L}, \sigma_{X_1Y_2L}, \sigma_{X_1Y_3L}, \sigma_{Y_1Y_2L}, \sigma_{Y_1Y_3L}\}$ is zero. Therefore, since $\sigma_{X_1Y_1}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{X_3Y_1} = \sigma_{X_1X_3}\sigma_{X_2Y_1}$ by assumption, from Lemma 2 we have

$$\frac{\sigma_{X_1X_3}}{\sigma_{X_3Y_1}} = \frac{\sigma_{X_1X_3L}}{\sigma_{X_3Y_1L}} \tag{8}$$

and from $\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$

$$\frac{\sigma_{Y_1Y_3}}{\sigma_{X_1Y_3}} = \frac{\sigma_{Y_1Y_3L}}{\sigma_{X_1Y_3L}} \tag{9}$$

Since no covariance among the given variables is zero,

$$\begin{aligned} \frac{\sigma_{X_1X_2}\sigma_{Y_1X_3}}{\sigma_{X_1Y_2}\sigma_{Y_1Y_3}} &= \frac{\sigma_{X_1X_3}\sigma_{Y_1X_2}}{\sigma_{X_1Y_3}\sigma_{Y_1Y_2}} \quad \Rightarrow \\ \sigma_{X_1X_2}\sigma_{Y_1Y_2} &= \sigma_{X_1Y_2}\sigma_{Y_1X_2}\frac{\sigma_{X_1X_3}\sigma_{Y_1Y_3}}{\sigma_{Y_1X_3}\sigma_{X_1Y_3}} \end{aligned}$$

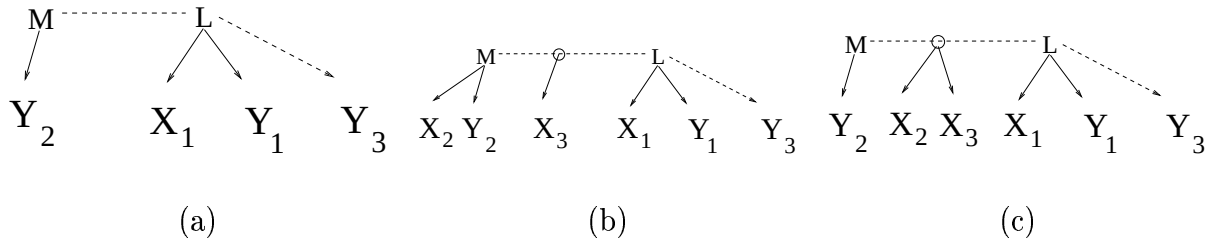

Figure 6: In (a), Y_2 and X_1 cannot share a parent, and because of the given tetrad constraints, L should d-separate M and Y_3. Y_3 is not a child of L either, but there will be a trek linking L and (not necessarily into) Y_3. In (b), a set of possible configurations for X_2 and X_3, where X_3 has some parent in the trek linking M and L. In (c), another variation where now X_2 and X_3 share a parent in that trek.

From (8), (9) it follows:

$$\begin{aligned}
\sigma_{X_1X_2}\sigma_{Y_1Y_2} &= \sigma_{X_1Y_2}\sigma_{Y_1X_2}\frac{\sigma_{X_1X_3L}\sigma_{Y_1Y_3L}}{\sigma_{Y_1X_3L}\sigma_{X_1Y_3L}} \\
&= \sigma_{X_1Y_2}\sigma_{Y_1X_2}\frac{(a/b)\sigma_{Y_1X_3L}(b/a)\sigma_{X_1Y_3L}}{\sigma_{Y_1X_3L}\sigma_{X_1Y_3L}} \\
&= \sigma_{X_1Y_2}\sigma_{Y_1X_2}
\end{aligned}$$

Contradiction. □

Lemma 4 *Let $G(\mathbf{O})$ be a linear latent variable graph. Assume $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subseteq \mathbf{O}$ and $\sigma_{X_1Y_1}\sigma_{X_2Y_2} = \sigma_{X_1Y_2}\sigma_{X_2Y_1}$, $\sigma_{X_2Y_1}\sigma_{Y_2Y_3} = \sigma_{X_2Y_3}\sigma_{Y_2Y_1}$, $\sigma_{X_1X_2}\sigma_{X_3Y_2} = \sigma_{X_1Y_2}\sigma_{X_3X_2}$, $\sigma_{X_1X_2}\sigma_{Y_1Y_2} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_1}$ and that for all triplets $\{A, B, C\}, \{A, B\} \subset \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}, C \in \mathbf{O}$, we have $\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$. Then X_1 and Y_1 do not have a common latent parent in G.*

Proof: Assume X_1 and Y_1 have a common latent parent L. Since $\sigma_{X_1Y_1}\sigma_{X_2Y_2} = \sigma_{X_1Y_2}\sigma_{X_2Y_1}$, L is a choke point for pairs $(X_1, X_2) \times (Y_1, Y_2)$ by the Tetrad Representation Theorem (Theorem 1). As a consequence, all treks between Y_2 and X_1 go through L. All treks between X_2 and Y_1 go through L. All treks between X_2 and Y_2 go through L. Such treks exist, since no correlation vanishes.

Consider $\sigma_{X_2Y_1}\sigma_{Y_2Y_3} = \sigma_{X_2Y_3}\sigma_{Y_2Y_1}$. Since there is a trek connecting X_2 and Y_1 through L, and X_2 and Y_2 through L, and L is a parent of Y_1, then L has also to be a choke point for pairs $(Y_1, Y_3) \times (X_2, Y_2)$, and all treks between Y_2 to Y_1 go through L.

Consider $\sigma_{X_1X_2}\sigma_{X_3Y_2} = \sigma_{X_1Y_2}\sigma_{X_3X_2}$. Since there is a trek connecting Y_2 and X_1 through L, and X_2 and Y_2 through L, and L is a parent of X_1, then L has also to be a choke point for pairs $(X_1, X_3) \times (X_2, Y_2)$, and all treks from X_2 to X_1 go through L.

At least one element in $\{X_2, Y_2\}$ has a trek to L that is not into L, or otherwise $\rho_{X_2Y_2} = 0$. Therefore, L is a choke point for pairs $(X_1, X_2) \times (Y_1, Y_2)$, which implies $\sigma_{X_1X_2}\sigma_{Y_1Y_2} = \sigma_{X_1Y_2}\sigma_{X_2Y_1}$. Contradiction. □

Lemma 5 *Let $G(\mathbf{O})$ be a linear latent variable graph. Assume $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\} \subseteq \mathbf{O}$ and $\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$, $\sigma_{X_1Y_2}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_2X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_2}$, $\sigma_{X_1Y_3}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_3X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_3}$, $\sigma_{X_1X_2}\sigma_{Y_2Y_3} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_3}$ and that for all triplets $\{A, B, C\}, \{A, B\} \subset \{X_1, X_2, X_3, Y_1, Y_2, Y_3\}, C \in \mathbf{O}$, we have $\rho_{AB} \neq 0, \rho_{AB.C} \neq 0$. Then X_1 and Y_1 do not have a common parent in G.*

Proof: Since we are working only with linear graphs, we will use the Tetrad Representation Theorem (Theorem 1) in this proof.

Suppose X_1 and Y_1 have a common parent L in G. Since there is a trek between X_1 and Y_1, $\sigma_{X_1Y_1}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{Y_1Y_3} = \sigma_{X_1Y_3}\sigma_{Y_1Y_2}$ holds but no observable partial correlation vanishes, then L should be a latent choke point for all pairs in $\{X_1, Y_1, Y_2, Y_3\}$.

It is also given that $\sigma_{X_1Y_2}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_2X_3} = \sigma_{X_1X_3}\sigma_{X_2Y_2}$ holds. Since $\sigma_{X_1X_2}\sigma_{Y_2Y_3} \neq \sigma_{X_1Y_2}\sigma_{X_2Y_3}$, by Lemma 3 then X_1 and Y_2 cannot share a parent. Let M be a parent of Y_2 that is connected to L through a trek that is not blocked by any observed node. If Y_3 has a parent in the trek $M - L$ that is not L, then L will not be a choke point of $(X_1, Y_3) \times (Y_1, Y_2)$, contrary to our hypothesis. Therefore no parent of Y_3 in a trek linking Y_2 and Y_3 can appear before L in the trek connecting M and L, as illustrated by Figure 6(a). Notice such treks should exist because all correlations are different from zero and, also by Lemma 3, Y_3 cannot be a child of L.

Neither X_2 nor X_3 can be on the trek $M - L$, or otherwise L will not be a choke point for the set $\{X_1, X_2, X_3, Y_2\}$. Also, assume then that X_2 and X_3 have parents before L in the trek connecting M and Y_3. It is not possible that X_2 is a child of M while X_3 is not (or vice-versa), or otherwise there will be no choke point to entail all three tetrads among $\{X_1, X_2, X_3, Y_2\}$ as required by hypothesis. Figure 6(b) illustrates this situation. Therefore, we have three possible situations: both are children of M, both share a common parent between M and L (Figure 6(c)), or each has a different parent in this trek.

In the first situation, where X_2 and X_3 are both children of M, we have a trek from X_3 to X_2 that goes through M but not through L, and a trek from X_1 to Y_3 that goes through L but not through M. No choke point exists for pairs (X_1, X_3) and (X_2, Y_3) which by the Tetrad Representation Theorem means that the tetrad $\sigma_{X_1Y_3}\sigma_{X_2X_3} = \sigma_{X_1X_2}\sigma_{Y_3X_3}$ cannot hold, contrary to our hypothesis. The second and third situations are analogous.

Assume then that X_2 is a child of L, and X_3 is a child of some other node in the trek connecting M and L (excluding L but including M). Again, that contradicts the first set of tetrad constraints given in the hypothesis, and by symmetry the same holds when X_3 is the child of L.

By an analogous argument, neither X_2 or X_3 can have a parent after L in the trek connecting M and Y_3. It is also not possible that the parent of one node lies between M and L and the other between L and Y_3: it is easy to see that there will be no choke point in all treks linking the elements in $\{X_1, X_2, X_3, Y_2\}$ or $\{X_1, X_2, X_3, Y_3\}$ as required.

Therefore, both X_2 and X_3 must be children of L, but then L will be a choke point between (X_1, Y_3) and (X_2, Y_2), which entails $\sigma_{X_1X_2}\sigma_{Y_2Y_3} = \sigma_{X_1Y_2}\sigma_{X_2Y_3}$, contrary to our hypothesis. □

Lemma 6 *CS3 is not sound for semilinear latent variable graphs.*

Proof: In order to show this, one has only to construct a semilinear latent variable graph with a latent covariance Σ_L such that it entails all constraints of CS3 but where X_1 and Y_1

L_1	L_2	L_3	L_4	L_5
1.0				
0.4636804781967626	1.0			
0.31177237495755117	0.1445627639088577	1.0		
0.8241967922523632	0.6834605230188671	0.45954945371001815	1.0	
0.5167659523766029	0.428525239857415	0.28813447630828753	0.7617079965565864	1.0

Table 8: A counterexample that can be used to prove Lemma 6.

have a same parent. Notice that the definition of entailment in semilinear graphs allows us to choose specific latent covariance matrices but the constraints should hold for any choice of linear coefficients and error variances.

Consider the graph G with five latent variables $L_i, 1 \leq i \leq 5$, where L_1 has X_1 and Y_1 as its only children, X_2 is the only child of L_2, X_3 is the only child of L_3, Y_2 is the only child of L_4 and Y_3 is the only child of L_5. Also, $\{X_1, X_2, X_3, Y_1, Y_2, Y_3\}$, as defined in CS3, are the only observed variables, and each observed variable has only one parent besides its error term. Error variables are independent.

The following simple randomized algorithm will choose a covariance matrix Σ_L for $\{L_1, L_2, L_3, L_4, L_5\}$ that entails CS3. The symbol σ_{ij} will denote the covariance of L_i and L_j.

1. Choose positive random values for all $\sigma_{ii}, 1 \leq i \leq 5$
2. Choose random values for σ_{12} and σ_{13}
3. $\sigma_{23} \leftarrow \sigma_{12}\sigma_{13}/\sigma_{11}$
4. Choose random values for σ_{45}, σ_{25} and σ_{24}
5. $\sigma_{14} \leftarrow \sigma_{12}\sigma_{45}/\sigma_{25}$
6. $\sigma_{15} \leftarrow \sigma_{12}\sigma_{45}/\sigma_{24}$
7. $\sigma_{35} \leftarrow \sigma_{13}\sigma_{45}/\sigma_{14}$
8. $\sigma_{34} \leftarrow \sigma_{12}\sigma_{45}/\sigma_{15}$
9. Repeat from the beginning if Σ_L is not positive definite or if $\sigma_{14}\sigma_{23} = \sigma_{12}\sigma_{34}$

Table 8 provides an example of such matrix. □.

Theorem 2 *There is some Σ_L such that $LT_M(\Sigma)$ and $ST_M(\Sigma, \Sigma_L)$ are not equal.*

Proof: Follows immediately from Lemmas 5 and 6. □

Theorem 3 *There is no locally sound tetrad constraint set of domain size less than 6 for deciding if two nodes A and B do not have a common parent in a latent variable graph G, if $\rho_{X_1X_2.X_3} \neq 0$ and $\rho_{X_1X_2} \neq 0$ for all $\{X_1, X_2\}$ in the domain of the constraint set and observed variable X_3.*

Proof: We will show the result for linear latent variable graphs. Since linear graphs are more constrained than semilinear ones, it will suffice. Moreover, we will be able to make use of the Tetrad Representation Theorem and the equivalence of d-separations and vanishing partial correlations, facilitating the proof.

This is trivial for domains of size 2 and 3, where no tetrad constraint can hold. For domains of size 4, let $\{A, B, C, D\}$ be our four variables. We will show that it does not matter which tetrad constraints hold among these four (excluding logically inconsistent constraints), there exist two linear latent variable graphs with observable variables $\{A, B, C, D\}$, G' and G'', where in the former A and B do not share a parent, while in latter they do have a parent in common.

Suppose first that all possible three tetrad constraints hold in the covariance matrix Σ of $\{A, B, C, D\}$, i.e., $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$. By the Tetrad Representation Theorem, there is a choke point CP in common among all treks linking these nodes in the unknown G. No choke point can be in $\{A, B, C, D\}$, or otherwise some vanishing partial correlations will hold. Therefore, during this proof we will only consider latent variables as choke points.

Let G' have two latent nodes L_1 and L_2, where L_1 is a common parent of A and L_2, and L_2 a parent of B, C and D. Let G'' have a latent node L_1 as the only parent of A, B, C and D, and no other edges, and the result will hold for this case.

Suppose now only one tetrad constraint hold instead of all three. First assume the choke point is between pairs $(A, B) \times (C, D)$. Create G'' with two latents, L_1 and L_2, where L_2 is parent of C and D, L_1 is a parent of L_2, A and B. For G', introduce another latent L_0, make it the parent of A, and change L_1 to be a parent of L_0 and B only.

Suppose now the only tetrad constraint that holds is the one entailed by a choke point between pairs $(A, C) \times (B, D)$ (the analogous case would be the pairs $(A, D) \times (B, C)$). Create G' again by using two latents L_1 and L_2, making L_2 a parent of B and C, and making L_1 a parent of L_2, A and D. Create G'' from G', by adding the edge $L_1 \rightarrow B$.

If no tetrad constraints hold, let G' be a linear latent variable graph with independent common latent parents for every pair in $\{A, B, C, D\} \times \{A, B, C, D\}$ but (A, B), and make B a parent of A. Let G'' be equal to G' with an added latent node L that is a common parent of A and B.

Now suppose our domain $\mathbf{S} = \{A, B, C, D, E\}$ has five variables, where Σ will now denote the covariance matrix of $\mathbf{S}$. Again, we will show how to build graphs G' and G'' in all possible consistent combinations of vanishing and non-vanishing tetrad constraints. This case is more complicated, and we will divide it in three major subcases.

Case 1: there is $X \in \{A, B\}$, such that no tetrad constraint holds in the covariance matrix of $\{X, C, D, E\}$. Suppose no tetrad constraint holds in $\{A, C, D, E\}$. We will deal with two main subcases, where the covariance matrix of $\{B, C, D, E\}$ has tetrad constraints or not.

Case 1.1: Suppose first that no tetrad constraint holds in the covariance matrix of $\{B, C, D, E\}$. If there is some other tetrad constraint with A and B in the opposite sides of the choke point, e.g. $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$, consider three scenarios. First, $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC} = \sigma_{AB}\sigma_{CD}$: if $\sigma_{AC}\sigma_{BE} = \sigma_{AE}\sigma_{BC}$, then it can be shown that we will also have $\sigma_{AE}\sigma_{BD} = \sigma_{AD}\sigma_{BE}$, and it can be shown by using the Tetrad Representation Theorem

that this will imply all three tetrads in $\{A, C, D, E\}$. Therefore, $\sigma_{AC}\sigma_{BE} \neq \sigma_{AE}\sigma_{BC}$ and $\sigma_{AE}\sigma_{BD} \neq \sigma_{AD}\sigma_{BE}$. Also, if there is some choke point $(A, E) \times (B, C)$ (or, analogously, $(A, E) \times (B, D)$) a similar reasoning will show that some choke point $(B, E) \times (C, D)$ will be implied, and therefore such choke point $(A, E) \times (B, C)$ cannot exist. The case $(A, C) \times (B, E)$ is analogous, and that covers all possible cases: build G' with four latents L_1, L_2, L_3, L_4 and L_5, where L_1 is a parent of A and L_2; L_2 is a parent of B, C, D, and E; L_3 a parent of E and C; L_4 a parent of E and D and L_5 a parent of B and E. Build G'' by adding an edge $L_2 \rightarrow A$.

Second, $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC} \neq \sigma_{AB}\sigma_{CD}$: still if $\sigma_{AC}\sigma_{BE} = \sigma_{AE}\sigma_{BC}$, then it can be shown that we will also have $\sigma_{AE}\sigma_{BD} = \sigma_{AD}\sigma_{BE}$. However, this will be analogous to a case in the previous paragraph, where some tetrad in $\{A, C, D, E\}$ will be implied. Assume for now that $\sigma_{AC}\sigma_{BE} \neq \sigma_{AE}\sigma_{BC}$ and $\sigma_{AE}\sigma_{BD} \neq \sigma_{AD}\sigma_{BE}$. The choke point $(A, D) \times (B, C)$ cannot happen, or otherwise this will imply $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC} = \sigma_{AB}\sigma_{CD}$. All cases $(A, E) \times (B, C), (A, C) \times (B, E), (A, E) \times (B, D)$ and $(A, D) \times (B, E)$ can be show to be impossible in exactly the same way as in the previous paragraph. Build G' with four latents: L_1, L_2, L_3 and L_4, where L_1 is a parent of A and L_2; L_2 is a parent of B, L_3 and L_4; L_3 is a parent of C, D, and L_4; L_4 is a parent of E. As before, add independent common parents between C and E, and D and E. Build G'' by adding an edge $L_2 \rightarrow A$. If $\sigma_{AC}\sigma_{BE} \neq \sigma_{AE}\sigma_{BC}$ and $\sigma_{AE}\sigma_{BD} = \sigma_{AD}\sigma_{BE}$, do the same as before, but add independent latent common parents between C and E, D and E.

Third, $\sigma_{AC}\sigma_{BD} \neq \sigma_{AD}\sigma_{BC} = \sigma_{AB}\sigma_{CD}$. It can be shown using the Tetrad Representation Theorem that we cannot both have $(A, B) \times (C, E)$ and $(A, B) \times (D, E)$, or otherwise we will have a choke point $(A, B) \times (C, D)$. Assume then that $(A, B) \times (D, E)$ does not hold. For the same reason, $(A, D) \times (B, C)$ cannot hold. And by a similar reasoning $(A, E) \times (B, C)$ and $(A, E) \times (B, D)$ cannot hold at the same time.

Assume for now that $(A, B) \times (C, E)$ holds. Then we cannot have $(A, E) \times (B, C)$ or $(A, B) \times (C, E)$, or otherwise we will have a $(A, C) \times (D, E)$ choke point (again, it follows from the Tetrad Representation Theorem). Suppose $(A, E) \times (B, D)$ holds. Again, one can show that $(A, E) \times (C, D)$ has to exist, which is a contradiction. A similar result is obtained with $(A, D) \times (B, E)$. Therefore, let's build G' with $(A, C) \times (B, D)$ and $(A, B) \times (C, E)$ only: L_1 as a parent of A and L_2; L_2 as a parent of B, C, D and E; extra independent common parents for C and E, D and E, B and D. Build G'' by adding the edge $L_2 \rightarrow A$.

Assume then that $(A, B) \times (C, E)$ does not hold. The remaining choke points that can be considered are $\{(A, E) \times (B, C), (A, C) \times (B, E), (A, E) \times (B, D), (A, D) \times (B, E)\}$ and we know that $(A, E) \times (B, C)$ and $(A, C) \times (B, E)$ cannot hold at the same time (or otherwise it will imply $(A, B) \times (C, E)$) and the same happens for $(A, E) \times (B, D)$ and $(A, D) \times (B, E)$. If none holds, this will be equivalent to the previous case with extra common parents for A and E, and for B and E. Since $(A, E) \times (B, C)$ and $(A, E) \times (B, D)$ cannot happen at the same time, assume for now that $(A, E) \times (B, C)$ holds but $(A, E) \times (B, D)$ does not. It turns out we cannot have $(A, E) \times (B, C)$ and $(A, D) \times (B, E)$ at the same time: $(B, D) \times (C, E)$ would hold. Build G' and G'' in the same way as before, but the independent common parents are now between B and D, as well as D and E. For the case where $(A, D) \times (B, E)$ holds instead of $(A, E) \times (B, C)$, build G' as follows: L_1 a parent of A, C and L_2; L_2 a parent of D and L_3; L_3 a parent of E and B. Add also a independent latent common parent for C and E.

For G'', add the edge $L_3 \to A$.

Case 1.2: Suppose now that no tetrad constraint holds in $\{A, C, D, E\}$, but some holds in $\{B, C, D, E\}$. Build a graph G_0 over $\{B, C, D, E\}$ such that latent L_1 is a parent of L_2, B and E; latent L_2 is a common parent of C and D, i.e., assume $\sigma_{BC}\sigma_{DE} = \sigma_{BD}\sigma_{CE} \neq \sigma_{BE}\sigma_{CD}$. Then since $\sigma_{AC}\sigma_{DE} \neq \sigma_{AD}\sigma_{CE}$, by multiplying both sides by $\sigma_{BD}\sigma_{BC}$ we get $\sigma_{AC}\sigma_{BD} \neq \sigma_{AD}\sigma_{BC}$, and therefore there is no choke point for pairs $(A, B) \times (C, D)$ if there is a choke point for $(B, E) \times (C, D)$. From this result, it is possible to show that we cannot have both choke points $(A, B) \times (C, E)$ and $(A, B) \times (D, E)$ simultaneously. Suppose without loss of generality we do not have $(A, B) \times (D, E)$. It is also possible to show from algebraic manipulation of tetrads that we have $(A, C) \times (B, E)$ if and only if $(A, D) \times (B, E)$. On the other hand, $(A, E) \times (B, C)$ and $(A, E) \times (B, D)$ cannot hold at the same time or otherwise $(A, E) \times (C, D)$ will be implied. We have all possible subsets of $\mathbf{T} = \{\{(A, B) \times (C, E)\}, \{(A, C) \times (B, E), (A, D) \times (B, E)\}, \{(A, E) \times (B, C)\}, \{(A, E) \times (B, D)\}\}$ to handle (i.e., ways of choosing which of these four sets hold and which do not).

If none of them hold, constructing G' and G'' will be trivial: add a latent L_0 to G_0 that will be a parent of A and L_1, and just add common parents to each pair (A, X), $X \in \{C, D, E\}$ to G_0, and in G'' make L_1 also a parent of A.

The second and last sets of $\mathbf{T}$ cannot hold at the same time, because they imply $(A, B) \times (D, E)$, discarded by hypothesis. It is actually possible to show that any two of the first three imply the third. So, suppose only $(A, E) \times (B, D)$ holds. Do the same as before to create G' and G'', but remove the common parent of D and A.

Now assume all the first three sets of $\mathbf{T}$ hold. But then there will be a choke point CP in $\{A, B, C, E\}$, and since $(B, E) \times (C, D)$ holds and all treks from B and E to C go through CP, that means CP is also a choke point of $(B, E) \times (C, D)$, and $(A, E) \times (C, D)$ will hold, which is a contradiction. Therefore, at most one of each of the first three sets can hold at the same time. Suppose $(A, B) \times (C, E)$ is the only one holding. Then build G' similar to the previous cases, but in this situation there will be extra common parents between A and C, and A and D only. Suppose $(A, E) \times (B, C)$ is the only one holding. Then build G' similar to the previous cases, but in this situation there will be extra common parents between A and E, and A and D only. Finally, suppose $\{(A, C) \times (B, E), (A, D) \times (B, E)\}$ is the only one holding. Then build G' similar to the previous cases, but in this situation there will be extra common parents between A and C, and A and D only.

If $\sigma_{BC}\sigma_{DE} = \sigma_{BD}\sigma_{CE} = \sigma_{BE}\sigma_{CD}$, from simple algebraic manipulations as discussed at the beginning of the previous case, we cannot have any $(A, B) \times (X_1, X_2) \subset \{C, D, E\}$ choke point. Also, if $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD}$, by multiplying both sides by $\sigma_{BE}\sigma_{DE}$, one gets $\sigma_{AB}\sigma_{DE} = \sigma_{AD}\sigma_{BE}$, and by symmetry, one gets that if there is a choke point $(A, X_1) \times (B, X_2)$ for some $\{X_1, X_2\} \subset \{C, D, E\}$, then there are choke points for all $\{X_1, X_2\} \subset \{C, D, E\}$. Similar to a argument given in the previous case, if all $(A, X_1) \times (B, X_2)$ exist, then there will be some choke point $(A, C) \times (D, E)$ for instance. Therefore, assume the only tetrad constraints in Σ are $\sigma_{BC}\sigma_{DE} = \sigma_{BD}\sigma_{CE} = \sigma_{BE}\sigma_{CD}$. Build G' with latent L_1, parent of A and L_2; L_2, parent of B, C, D and E. Add independent latent common causes between A and C, A and D, A and E. Build G'' from G' by adding the edge $L_2 \to A$.

Case 2: there is $X \in \{A, B\}$, such that all tetrad constraints hold in the covariance

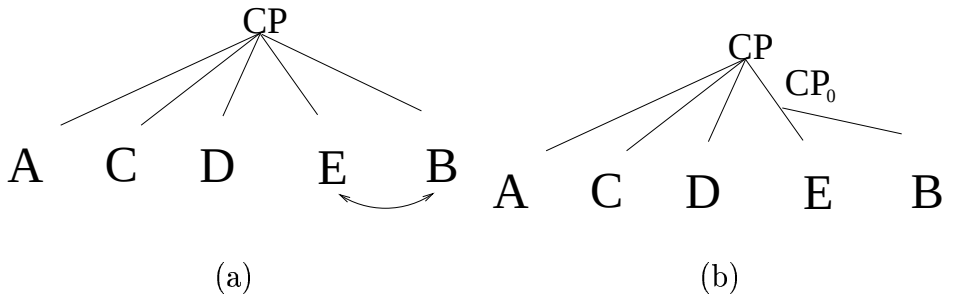

Figure 7: In (a), CP is a choke point for $(B, E) \times (C, D)$. Another trek has to exist between B and E to satisfy the assumptions. It is not possible then to have a choke point between $(A, E) \times (B, C)$ (or pair B, D) even if there are no other treks. In (b), CP_0 is a choke point for $(B, E) \times (C, D)$. Again, no choke point $(A, E) \times (B, C)$ is possible.

matrix of $\{X, C, D, E\}$. Suppose that all tetrad constraints holds among $\{A, C, D, E\}$ and assume for now that the same holds for $\{B, C, D, E\}$. Since there is a choke point CP in all treks among $\{C, D, E\}$, it is clear that all treks from A or B to $\{C, D, E\}$ go through CP, and therefore there is a choke point for all pairs $\{A, B\} \times \{X_1, X_2\} \subset \{C, D, E\}$.

If $\sigma_{AC}\sigma_{BD} = \sigma_{AB}\sigma_{CD}$, then by multiplying both sides by σ_{AE} one obtains $\sigma_{AE}\sigma_{BD} = \sigma_{AB}\sigma_{DE}$. Therefore, if for some $\{X_1, X_2\} \subset \{C, D, E\}$ we have a choke point between $\{A, X_1\} \times \{B, X_2\}$, then all three tetrads constraints will hold any subset of four variables in $\mathbf{S}$. So, first assume this happens. Build G' with two latents L_1 and L_2, where L_1 is parent of A and L_2, and L_2 is a parent of all elements in $\mathbf{S} - \{A\}$. Build G'' by using one latent L_1 and making it the parent of all elements of $\mathbf{S}$.

If $\sigma_{AX_1}\sigma_{BX_2} \neq \sigma_{AB}\sigma_{X_1X_2}$ for all $\{X_1, X_2\} \subset \{C, D, E\}$, build G' with three latents L_1, L_2 and L_3: make L_1 parent of A and L_2, make L_2 parent of L_3 and B, make L_3 parent of C, D and E. Build G'' from G' by adding an edge $L_2 \to A$.

Suppose that all tetrad constraints holds among $\{A, C, D, E\}$ and assume now that only one holds for $\{B, C, D, E\}$. Without loss of generality, let $\sigma_{BC}\sigma_{DE} = \sigma_{BD}\sigma_{CE}$ be such constraint. This also implies $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$ and therefore there is also a choke point between $(A, B) \times (C, D)$. Similarly, it is implied there is no choke point for pairs $(A, B) \times (C, E)$ and $(A, B) \times (D, E)$.

We cannot have choke points for pairs $(A, E) \times (B, C)$ or $(A, E) \times (B, D)$: since there is a choke CP for the set $\{A, C, D, E\}$ (and CP cannot be in $\mathbf{S}$ or we would have some vanishing partial correlations in this set), there are only two possible ways of finding a choke point for $(B, E) \times (C, D)$. One is illustrated in Figure 7(a): the choke point is CP, but then we have to introduce another trek between E and B. Even if there is no other trek between A and B, because of the extra trek between B and E, we cannot have a choke point between pairs $(A, E) \times (B, C)$, for instance. The second alternative is having the choke point for $(B, E) \times (C, D)$ as depicted in Figure 7(b). Even if there are no other treks between A and B, again we cannot have a choke point between pairs $(A, E) \times (B, C)$.

Suppose that $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} (= \sigma_{AD}\sigma_{BC})$. Since $\sigma_{AE}\sigma_{CD} = \sigma_{AC}\sigma_{DE} = \sigma_{AD}\sigma_{BE}$ is given, we would have $\sigma_{AB}\sigma_{DE} = \sigma_{AE}\sigma_{BD}$ and $\sigma_{AB}\sigma_{CE} = \sigma_{AE}\sigma_{BC}$, and the converse holds.

Therefore, if there is a choke point for some pair in $\{(A,C)\times(B,D),(A,D)\times(B,C),(A,D)\times(B,E),(A,C)\times(B,E)\}$, then it holds for all pairs in this set. Suppose this is true. Graph G' can be constructed as follows: use latents L_1 and L_2, make L_1 parent of A, C, D and L_2, and L_2 parent of B and E. In order to create G'', just add an edge $L_1 \rightarrow B$.

Now suppose that there are no choke points for all pairs in $\{(A,C)\times(B,D),(A,D)\times(B,C),(A,D)\times(B,E),(A,C)\times(B,E)\}$. Graph G' can be constructed as follows: latent L_1 as a parent of A and L_2; latent L_2 as a parent of B and L_3; latent L_3 as a parent of C, D and E; another latent as an independent common parent of B and E; and, finally, B as a parent of A. G'' can be constructed from G' by adding a edge from L_2 to A.

Case 3: for all $X \in \{A,B\}$, exactly one tetrad constraint holds in the covariance matrix of $\{X,C,D,E\}$. Without loss of generality, assume the only tetrad constraint holding among $\{A,C,D,E\}$ is $\sigma_{AC}\sigma_{DE} = \sigma_{AD}\sigma_{CE}$. We have two possible choices for the only constraint in $\{B,C,D,E\}$: $(B,E)\times(C,D)$ and $(B,C)\times(D,E)$. (The case $(B,D)\times(C,E)$ is analogous to $(B,C)\times(D,E)$ by symmetry.)

Case 3.1: Suppose the only tetrad constraint in $\{B,C,D,E\}$ is due to the choke point $(B,E)\times(C,D)$. Since $\sigma_{AC}\sigma_{DE} = \sigma_{AD}\sigma_{CE}$ and $\sigma_{BC}\sigma_{DE} = \sigma_{BD}\sigma_{CE}$, this also implies $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$, i.e., a choke point $(A,B)\times(C,D)$. Suppose there is a choke point $(A,B)\times(C,E)$. We have that $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$ and $\sigma_{AC}\sigma_{BE} = \sigma_{AE}\sigma_{BC}$ will imply a choke point $(A,B)\times(D,E)$ and the converse holds by symmetry.

If $(A,C)\times(B,E)$ exists, then since $\sigma_{BD}\sigma_{CE} = \sigma_{BC}\sigma_{DE}$, together with tetrad $\sigma_{AB}\sigma_{CE} = \sigma_{AE}\sigma_{BC}$ we will have $\sigma_{AB}\sigma_{DE} = \sigma_{AE}\sigma_{BD}$, i.e., the existence of a choke point $(A,C)\times(B,E)$ implies $(A,D)\times(B,E)$, and the converse holds by symmetry.

Assume choke point $(A,C)\times(B,D)$ exists. That means there is a single choke point linking all elements in $\{A,B,C,D\}$. But, then by an argument similar to the one described in Case 2, one cannot have choke points $(A,B)\times(C,E)$ or $(A,B)\times(D,E)$, or otherwise one would have other tetrad constraints holding in the covariance matrix of $\{B,C,D,E\}$. Similarly, no choke points $(A,C)\times(B,E)$ or $(A,E)\times(B,C)$ are allowed. So, in this situation where $\sigma_{AB}\sigma_{CD} = \sigma_{AD}\sigma_{BC}$, construct G' with latent L_1 as parent of A and L_2, latent L_2 as parent of B, C, D, E, and another independent common parents of A and E, and/or B and E. For G'', just add an edge $L_2 \rightarrow A$.

Assume then there is no choke point $(A,C)\times(B,D)$. We have last four cases to handle in this first half of Case 3: the possibility of the set of constraints $\{(A,B)\times(C,E),(A,B)\times(D,E)\}$ and $\{(A,C)\times(B,E),(A,D)\times(B,E)\}$ both holding, each holding separately, or none holding. If both hold at the same time, construct G' with three latents, where L_1 is a parent of L_2 and A; L_2 is a parent of L_3, B and E; L_3 is a parent of C and D; make also B a parent of A. Construct G'' from G' with an extra edge $L_2 \rightarrow A$. If none holds at the same time, we have G' and G'' similar to the previous paragraph, but now with an extra edge $B \rightarrow A$.

Assume choke points $\{(A,B)\times(C,E),(A,B)\times(D,E)\}$ exist, and $\{(A,C)\times(B,E),(A,D)\times(B,E)\}$ do not. Construct G' with latent L_1, parent of A and L_2; latent L_2, parent of B, L_3 and L_4; latent L_3, parent of C, D and L_4; latent L_4, parent of E. Add also $B \rightarrow A$. Construct G'' from G' by adding the edge $L_2 \rightarrow A$.

Assume choke points $\{(A,B)\times(C,E),(A,B)\times(D,E)\}$ do not exist, and $\{(A,C)\times$

$(B,E),(A,D)\times(B,E)\}$ do. Build G' with three latents: L_1 as a parent of L_2, L_3 and A; L_2 as a parent of L_3, B and E; L_3 as a parent of C and D. For G'', add the edge $L_2 \to A$.

Case 3.2: finally, suppose now the only tetrad constraint in $\{B,C,D,E\}$ is due to the choke point $(B,C)\times(D,E)$. This is a slightly more complicated case.

First, let's consider which combinations of choke points $(A,B)\times(X_1,X_2)\subset \mathbf{S}$ are possible. $(A,B)\times(C,D)$ cannot hold, or otherwise $\sigma_{AC}\sigma_{BD}=\sigma_{AD}\sigma_{BC}$ combined with $\sigma_{AC}\sigma_{DE}=\sigma_{AD}\sigma_{CE}$ will imply the constraint $\sigma_{BD}\sigma_{CE}=\sigma_{BC}\sigma_{DE}$, a $(B,E)\times(C,D)$ choke point discarded by hypothesis. By symmetry, $(A,B)\times(D,E)$ cannot hold. Suppose now $(A,B)\times(C,E)$ holds. Let $CP\notin\mathbf{S}$ be a choke point for pairs $(A,E)\times(C,D)$ closest to A. Since all treks from A to C and from E to C go through CP, and because $\sigma_{AE}\neq 0$ we know these treks cannot be both into CP, and therefore we know there is a trek linking A and E that goes through CP.

CP has to be a choke point in $(A,B)\times(C,E)$. Otherwise, suppose for instance CP_2 is a choke point, and CP_2 is in a trek between E and CP. There should be a trek linking B and E that does not go through CP, or otherwise CP would be a choke point. Then, since $(B,C)\times(D,E)$ holds and all correlations are nonzero, there should be a trek connecting B and D that does not go through CP. But then there would be no choke point for $(B,C)\times(D,E)$, since from the pair $(A,E)\times(C,D)$ all treks between C and E go through CP. Contradiction.

Since all treks from B to E go through CP and all treks from C to E also go through CP, then CP is also a choke point in $(B,C)\times(D,E)$. And since all correlations are different from zero, CP is a choke point in all treks linking elements in $\{A,C,D,E\}$, contrary to the hypothesis.

Similarly, suppose there is a choke point CP for $(A,C)\times(B,D)$ closest to B. As in the previous case, there will be a trek from B to D through CP, and all treks from C to D are through CP. As in the previous case, we can show that CP will also be the choke point for $(B,C)\times(D,E)$, or otherwise there will be no choke point for $(A,E)\times(C,D)$. This will imply all three tetrads in $\{B,C,D,E\}$, a contradiction. Therefore, there is no choke point for $(A,C)\times(B,D)$ and neither, by symmetry, for $(A,D)\times(B,E)$.

Similarly, suppose there is a choke point CP for $(A,C)\times(B,E)$ closest to B. As in the previous case, there will be a trek from B to E through CP, and all treks from C to E are through CP. As in the previous case, we can show that CP will also be the choke point for $(B,C)\times(D,E)$, or otherwise there will be no choke point for $(A,E)\times(C,D)$. This will imply also that CP is in all treks between A and C, and therefore imply all tetrads in $\{A,C,D,E\}$, which is a contradiction. Therefore, there is no choke point for $(A,C)\times(B,E)$.

Suppose now there is a choke point CP for $(A,D)\times(B,C)$. Build G' with latent L_1 as a parent of L_2 and A; L_2 as a parent of L_3 and D; L_3 as a parent of B and C. Build G'' by adding edges $L_2\to A$ and $L_2\to B$ to G'. The same holds for $(A,E)\times(B,D)$ by symmetry. $(A,D)\times(B,C)$ is also equivalent to $(A,E)\times(B,C)$ because of the constraint given by $(B,C)\times(D,E)$: $\sigma_{AB}\sigma_{CD}=\sigma_{AC}\sigma_{BD}$ and $\sigma_{BE}\sigma_{CD}=\sigma_{BD}\sigma_{CE}$ imply $\sigma_{AB}\sigma_{CE}=\sigma_{AC}\sigma_{BE}$.

The final alternative is assuming that only $(B,C)\times(D,E)$ and $(A,E)\times(C,D)$. This case is simple: for G', make L_1 parent of L_2 and D; L_2 parent of L_3, A and E; L_3 parent of B and C. Also add other independent common parents for A and E, and for B and C. For G'', simply add the edge $L_2\to B$ to G'. □

Lemma 7 *Let $G(\mathbf{O})$ be a semilinear latent variable graph. Then, if for $\{A, B, C\} \subseteq \mathbf{O}$ we have $\rho_{AB} = 0$ or $\rho_{AB.C} = 0$, then A and B cannot share a common latent parent in G with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

Proof: Let A, B, C be defined according to the following linear functions

$$\begin{aligned} A &= aL + \textstyle\sum_p a_p A_p + \epsilon_A \\ B &= bL + \textstyle\sum_i b_i B_i + \epsilon_B \\ C &= \textstyle\sum_j c_j C_j + \epsilon_C \end{aligned}$$

where L is a common latent parent of A and B, $\{A_p\}$ represents parents of A, $\{B_i\}$ are parents of B, $\{C_j\}$ parents of C, and $\{a_p\} \cup \{b_i\} \cup \{c_j\} \cup \{a, b, \zeta_A, \zeta_B, \zeta_C\}$ are parameters of the measurement model, $\{\zeta_A, \zeta_B, \zeta_C\}$ being the variances of error terms $\{\epsilon_A, \epsilon_B, \epsilon_C\}$, respectively.

Assume $\sigma_{AB} = 0$. By the equations above, $\sigma_{AB} = ab\sigma_L^2 + K$, where no term in K that has a factor ab. For this identity to hold, we therefore need $ab\sigma_L^2 = 0$. By assumption, latent variables have positive variance, so the fact that $ab\sigma_L^2 = 0$ implies $\sigma_L^2 = 0$ is a contradiction.

Since $\rho_{AB.C} = 0$ if and only if $\sigma_{AB}\sigma_C^2 - \sigma_{AC}\sigma_{BC} = 0$ for positive σ_C^2 , assume the latter. Expressing this polynomial as a function of the given coefficients, we obtain $ab\sigma_L^2\sigma_C^2 + Q$. Since C is not an ancestor of L (because L is latent) no term in $ab\sigma_L^2$ contains the symbol ζ_C, nor any coefficient $\{c_j\}$. Since every term in $\sigma_{AC}\sigma_{BC}$ that might contain ζ_C must also contain some $\{c_j\}$, then no term in $\sigma_{AC}\sigma_{BC}$ can cancel any term in $ab\sigma_L^2\zeta_C$ (which is contained in $ab\sigma_C^2\sigma_C^2$). This implies $ab\sigma_L^2\zeta_C = 0$, a contradiction. □

Lemma 8 *Let $G(\mathbf{O})$ be a latent variable graph. Let $\{A, B, C\} \subset \mathbf{O}$ be some triplet such that A and B have parents L_1 and L_2, respectively (where it is possible that $L_1 = L_2$), and C is not an ancestor of A or B. Then, if $\sigma_{L_1L_2} \neq 0$, it follows that $\rho_{AB.C} \neq 0$ with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

Proof: Let the structural equations for A, B and C be $A = aL_1 + \sum_i a_i A_i + \epsilon_a$, $B = bL_2 + \sum_j b_j B_j + \epsilon_b$ and $C = \sum_k c_k C_k + \epsilon_c$, where ϵ_a, ϵ_b and ϵ_c are independent random variables, and independent of every other random variable in G besides their respective descendants.

We have that $\rho_{AB.C} \neq 0 \Leftrightarrow \sigma_{AB}\sigma_C^2 - \sigma_{AC}\sigma_{BC} \neq 0$. We will prove that $\sigma_{AB}\sigma_C^2 - \sigma_{AC}\sigma_{BC} \neq 0$. From the above equations, we have that $\sigma_{AB}\sigma_C^2 - \sigma_{AC}\sigma_{BC} = [ab\sigma_{L_1L_2} + F_1(A, B)](F_2(C) + \zeta_c) - \sigma_{AC}\sigma_{BC}$, where no term in $F_1(A, B)$ can contain the product ab, every term in $F_2(C)$ contains some variable c_k as well as every term in $\sigma_{AC}\sigma_{BC}$, and ζ_c is the variance of the error variance of C. The term $\sigma_{L_1L_2}$ cannot contain any variable c_k, since C is not an ancestor of A or B. Therefore, no term in this polynomial can cancel the term $ab\sigma_{L_1L_2}\zeta_c$, and since $ab\sigma_{L_1L_2}\zeta_c \neq 0$, it follows that $\rho_{AB.C} \neq 0$. □

Lemma 9 *Let $G(\mathbf{O})$ be a latent variable graph with latent covariance matrix Σ_L. For any set $\{A, B, C, D\} = \mathbf{O}' \subseteq \mathbf{O}$, if $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BD}$ and for every set $\{X, Y\} \subset \mathbf{O}', Z \in \mathbf{O}$ we have $\rho_{XY.Z} \neq 0$ and $\rho_{XY} \neq 0$, then A and B do not have more than one common parent in G with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

Proof: Assume L_1 and L_2 are two common parents of A and B in G. Let the graph G' have the same structure as G, but without all edges from other possible parents of A and B not in $\{L_1, L_2\}$. Since G' is more constrained than G, if a tetrad constraint holds in G, then it holds in G'. By Lemma 1, no element in $\mathbf{O}'$ is an ancestor of any other element in this set. Let the structural equations for A, B, C and D in G' be:

$$\begin{array}{rcl} A & = & \alpha_1 L_1 + \alpha_2 L_2 \\ B & = & \beta_1 L_2 + \beta_2 L_2 \\ C & = & \sum_j c_j C_j \\ D & = & \sum_k d_k D_k \end{array}$$

Consider only the choice of coefficient and error variances by which the given constraint is entailed by G and all latent covariance matrices. As argued in previous lemmas, we know this happens with probability 1. Since the tetrad constraint $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD}$ is entailed G', we have $\sigma_{AB}\sigma_{CD} - \sigma_{AC}\sigma_{BD} = 0 \Rightarrow (\alpha_1\beta_1\sigma^2_{L_1} + \alpha_1\beta_2\sigma_{L_1L_2} + \alpha_2\beta_1\sigma_{L_1L_2} + \alpha_2\beta_2\sigma^2_{L_2})\sigma_{CD} - (\alpha_1 \sum_j c_j\sigma_{C_jL_1} + \alpha_2 \sum_j c_j\sigma_{C_jL_2})(\beta_1 \sum_k d_k\sigma_{D_kL_1} + \beta_2 \sum_k d_k\sigma_{D_kL_2}) = 0 \Rightarrow \alpha_1\beta_1(\sigma^2_{L_1}\sigma_{CD} - (\sum_j c_j\sigma_{C_jL_1})(\sum_k d_k\sigma_{D_kL_1})) + f(G) = 0$, where

$$f(G) = (\alpha_1\beta_2\sigma_{L_1L_2} + \alpha_2\beta_1\sigma_{L_1L_2} + \alpha_2\beta_2\sigma^2_{L_2})\sigma_{CD} - \alpha_2 \sum_j c_j\sigma_{C_jL_2}(\beta_1 \sum_k d_k\sigma_{D_kL_1} + \beta_2 \sum_k d_k\sigma_{D_kL_2})$$

When fully expanding $f(G)$ as a function of the linear parameters of G, the product $\alpha_1\beta_1$ cannot possibly appear, since no element in $\mathbf{O}'$ is an ancestor of any other element in this set, Therefore, since the polynomial constraint is identically zero and nothing in $f(G)$ can cancel the term $\alpha_1\beta_1$, we have:

$$\sigma^2_{L_1}\sigma_{CD} = \sum_j c_j\sigma_{C_jL_1} \sum_k d_k\sigma_{D_kL_1} \tag{10}$$

Using a similar argument for the coefficients of $\alpha_1\beta_2$, $\alpha_2\beta_1$ and $\alpha_2\beta_2$, we get:

$$\sigma_{L_1L_2}\sigma_{CD} = \sum_j c_j\sigma_{C_jL_1} \sum_k d_k\sigma_{D_kL_2} \tag{11}$$

$$\sigma_{L_1L_2}\sigma_{CD} = \sum_j c_j\sigma_{C_jL_2} \sum_k d_k\sigma_{D_kL_1} \tag{12}$$

$$\sigma^2_{L_2}\sigma_{CD} = \sum_j c_j\sigma_{C_jL_2} \sum_k d_k\sigma_{D_kL_2} \tag{13}$$

From (10),(11), (12), (13), it follows:

$$\begin{aligned}
\sigma_{AC}\sigma_{AD} &= [\alpha_1 \textstyle\sum_j c_j \sigma_{C_j L_1} + \alpha_2 \sum_j c_j \sigma_{C_j L_2}][\alpha_1 \sum_k d_k \sigma_{D_k L_1} + \alpha_2 \sum_k d_k \sigma_{D_k L_2}] \\
&= \alpha_1^2 \textstyle\sum_j c_g \sigma_{C_j L_1} \sum_k d_k \sigma_{D_k L_1} + \alpha_1\alpha_2 \sum_j c_j \sigma_{C_j L_1} \sum_k d_k \sigma_{D_k L_2} + \\
&\quad \alpha_1\alpha_2 \textstyle\sum_j c_j \sigma_{C_j L_2} \sum_k d_k \sigma_{D_k L_1} + \alpha_2^2 \sum_j c_j \sigma_{C_j L_2} \sum_k d_k \sigma_{D_k L_2} \\
&= [\alpha_1^2 \sigma_{L_1}^2 + 2\alpha_1\alpha_2 \sigma_{L_1 L_2} + \alpha_2^2 \sigma_{L_2}^2]\sigma_{CD} \\
&= \sigma_A^2 \sigma_{CD}
\end{aligned}$$

which implies $\sigma_{CD} - \sigma_{AC}\sigma_{AD}(\sigma_A^2)^{-1} = 0 \Rightarrow \rho_{CD.A} = 0$. By Lemma 8, C and D have no correlated parents, which entails $\sigma_{CD} = 0$ in G'. Since all treks between C and D in G are preserved in G', that implies $\sigma_{CD} = 0$ is entailed by G. Contradiction. □

Lemma 10 *Let $G(\mathbf{O})$ be a latent variable graph with latent covariance matrix Σ_L. For any set $\{A, B, C, D\} = \mathbf{O}' \subseteq \mathbf{O}$, if $\sigma_{AB}\sigma_{CD} = \sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BD}$ and for every set $\{X, Y\} \subset \mathbf{O}', Z \in \mathbf{O}$ we have $\rho_{XY.Z} \neq 0$ and $\rho_{XY} \neq 0$, then if A and B have a common latent parent L_1 in G, B and C have a common latent parent L_2 in G, we have $L_1 = L_2$ with probability 1 with respect to a Lebesgue measure over the coefficient and error variance parameters.*

Proof: Assume A, B and C are parameterized as follows:

$$\begin{aligned}
A &= aL_1 + \textstyle\sum_p a_p A_p \\
B &= b_1 L_1 + b_2 L_2 + \sum_i b_i B_i \\
C &= cL_2 + \textstyle\sum_j c_j C_j
\end{aligned}$$

where as before $\{A_p\} \cup \{B_i\} \cup \{C_j\}$ represents the possible other parents of A, B and C, respectively. Assume $L_1 \neq L_2$. We will show that $\rho_{L_1 L_2} = 1$, which is a contradiction. From the given tetrad constraint $\sigma_{AB}\sigma_{CD} = \sigma_{AD}\sigma_{BC}$, and the fact that from Lemma 1 we have that for no pair $\{X, Y\} \subset \mathbf{O}'$ X is an ancestor of Y, if we factorize the constraint according to which terms include ab_1c as a factor, we obtain with probability 1:

$$ab_1c[\sigma_{L_1}^2 \sigma_{L_2 D} - \sigma_{L_1 D}\sigma_{L_1 L_2}] \tag{14}$$

If we factorize such constraint according to ab_2c, it follows:

$$ab_2c[\sigma_{L_1 L_2}\sigma_{L_2 D} - \sigma_{L_1 D}\sigma_{L_2}^2] \tag{15}$$

From (14) and (15), it follows that $\sigma_{L_1}^2 \sigma_{L_2}^2 = (\sigma_{L_1 L_2})^2 \Rightarrow \rho_{L_1 L_2} = 1$. Contradiction. □

Theorem 4 *The output of* FindPattern *is a generalized measurement pattern with respect to the tetrad and vanishing partial correlation constraints of Σ with probability 1.*

Proof: Two nodes will not share a common latent parent in a measurement pattern if and only if they are not linked by an edge in graph C constructed by algorithm FindPattern and that happens if and only if some partial correlation vanishes or if any of rules CS1, CS2 or CS3 holds. But then by Lemmas 3, 4, 5 and 7 the claim is proved. The claim about undirected edges follows directly from Lemma 1. □

Theorem 5 *Let $G(\mathbf{O})$ be a latent variable graph. Then the output of* BuildPureClusters *is a valid l-interpretation for G in the family of tetrad and vanishing partial correlation constraints and a pure generalized measurement pattern.*

Proof: The output is a pure measurement model and generalized measurement pattern by construction: each node has only one latent parent, and there are no edges linking observed nodes. We only have to show that all tetrad constraints entailed by such measurement model also hold in the population covariance matrix.

Let $\{A, B, C, D\}$ be four observed nodes. If $\{A, B, C\}$ belong to the same latent parent, then all tetrad constraints will be entailed by a pure measurement model with respect to a fourth node D, and by Step 5 of Table 2, this will be guaranteed. Now suppose $\{A, B\}$ have the same latent parent, while C and D are children of other parents (where C and D might have the same parent). Then the tetrad $\sigma_{AC}\sigma_{BD} = \sigma_{AD}\sigma_{BC}$ will be entailed, and this will always hold in the covariance matrix, by Step 6 of Table 2.

The tetrad $\sigma_{AB}\sigma_{CD} = \sigma_{AD}\sigma_{BC}$ will not be entailed: if L_1 is the parent of A and B, L_2 is the parent of C and L_3 is the parent of D, this will require $\rho_{L_2 L_3.L_1} = 0$, which will hold only in some latent covariance matrices, contrary to the definition of entailment in measurement models. Similarly, if no two elements in $\{A, B, C, D\}$ share a common parent in the output, then no tetrad will be entailed in this set except for specific latent covariance matrices. □

Corollary 1 *Let $G(\mathbf{O})$ be a latent variable graph. Then the output of* BuildPureClusters *is a l-interpretation for G in the family of tetrad and vanishing partial correlation constraints even when rules CS1, CS2 and CS3 are applied an arbitrary number of times in* FindPattern *for any arbitrary subset of nodes and an arbitrary number of maximal cliques is found.*

Proof: Independently of the choice made on Step 2 of BuildPureClusters, by the end of Step 4 we will meet all the conditions used to prove Theorem 5: that nodes in different clusters cannot share a same parent nor be ancestors of each other. The rest follows directly from the proof of Theorem 5. □

B The spiritual coping questionnaire

The following questionnaire is provided to facilitate understanding of the religious/spiritual coping example given in Section 7.2. It can also serve as an example of how questionnaires are actually designed.

Section I This section intends to measure the level of stress of the subject. In the actual questionnaire, it starts with the following instructions:

Circle the number next to each item to indicate how stressful each of these events has been for you since entered your graduate program. If you have never experienced one of the events listed below, then circle number 1. If one of the events listed below has happened to you and has caused you a great deal of stress, rate that event toward the "Extremely Stressful" end

of the rating scale. If an event has happened to you while you have been in graduate school, but has not bothered you at all, rate that event toward the lower end of the scale ("Not at all Stressful").

The student then chooses the level of stress by circling a number on a 7 point scale. The questions of this section are:

1. Fulfilling responsibilities both at home and at school
2. Trying to meet peers of your race/ethnicity on campus
3. Taking exams
4. Being obligated to participate in family functions
5. Arranging childcare
6. Finding support groups sensitive to your needs
7. Fear of failing to meet program expectations
8. Participating in class
9. Meeting with faculty
10. Living in the local community
11. Handling relationships
12. Handling the academic workload
13. Peers treating you unlike the way they treat each other
14. Faculty treating you differently than your peers
15. Writing papers
16. Paying monthly expenses
17. Family having money problems
18. Adjusting to the campus environment
19. Being obligated to repay loans
20. Anticipation of finding full-time professional work
21. Meeting deadlines for course assignments

Section II This section intends to measure the level of depression of the subject. In the actual questionnaire, it starts with the following instructions:

Below is a list of the ways you might have felt or behaved. Please tell me how often you have felt this way during the past week.

The student then chooses the level of frequency that some events happened to him/her by circling a number on a 4 point scale. The scale is "Rarely or None of the Time (less than 1 day)", "Some or Little of the Time (1 - 2 days)", "Occasionally or a Moderate Amount of the Time (3 - 4 days)" and "Most or All of the Time (5 - 7 days)". The events are as follows:

1. I was bothered by things that usually don't bother me
2. I did not feel like eating; my appetite was poor
3. I felt that I could not shake off the blues even with help from my family or friends
4. I felt that I was just as good as other people
5. I had trouble keeping my mind on what I was doing
6. I felt depressed
7. I felt that everything I did was an effort
8. I felt hopeful about the future
9. I thought my life had been a failure
10. I felt fearful
11. My sleep was restless
12. I was happy
13. I talked less than usual
14. I felt lonely
15. People were unfriendly
16. I enjoyed life
17. I had crying spells
18. I felt sad
19. I felt that people disliked me
20. I could not get "going"

Section III This section intends to measure the level of spiritual coping of the subject. In the actual questionnaire, it starts with the following instructions:

Please think about how you try to understand and deal with major problems in your life. These items ask what you did to cope with your negative event. Each item says something about a particular way of coping. To what extent is your religion or higher power involved in the way you cope?

The student then chooses the level of importance of some spiritual guideline by circling a number on a 4 point scale. The scale is "Not at all", "Somewhat", "Quite a bit", "A great deal". The guidelines are:

1. I think about how my life is part of a larger spiritual force
2. I work together with God (high power) as partners to get through hard times
3. I look to God (high power) for strength, support, and guidance in crises
4. I try to find the lesson from God (high power) in crises

5. I confess my sins and ask for God (high power)'s forgiveness
6. I feel that stressful situations are God (high power)'s way of punishing me for my sins or lack of spirituality
7. I wonder whether God has abandoned me
8. I try to make sense of the situation and decide what to do without relying on God (high power)
9. I question whether God (high power) really exists
10. I express anger at God (high power) for letting terrible things happen
11. I do what I can and put the rest in God (high power)'s hands
12. I do not try much of anything; simply expect God (high power) to take my worries away
13. I pray for a miracle
14. I pray to get my mind off of my problems
15. I ignore advice that is inconsistent with my faith
16. I look for spiritual support from clergy
17. I disagree with what my religion wants me to do or believe
18. I ask God (high power) to help me find a new purpose in life
19. I try to find a completely new life through religion
20. I seek help from God (high power) in letting go of my anger