An Algorithm and Complexity Results for Causal Unit Selection

Author names withheld

Editor: Under Review for CLeaR 2023

Abstract

The unit selection problem aims to identify objects, called units, that are most likely to exhibit a desired mode of behavior when subjected to stimuli (e.g., customers who are about to churn but would change their mind if encouraged). Unit selection with counterfactual (causal) objective functions was introduced relatively recently with existing work focusing on bounding a specific class of objective functions, called benefit functions, based on observational and interventional data—assuming a fully specified model is not available to evaluate these functions. We complement this line of work by proposing the first exact algorithm for finding optimal units given a relatively broad class of causal objective functions and a fully specified model in the form of a structural causal model (SCM). We show that unit selection under this class of objective functions is NP^{PP}-complete but is NP-complete when unit variables correspond to all exogenous variables in the SCM. We also provide treewidth-based complexity bounds on our proposed algorithm while relating it to a well-known algorithm for Maximum a Posteriori (MAP) inference.

Keywords: unit selection, structural causal models, counterfactual reasoning

1. Introduction

A theory of causality has emerged over the last few decades based on two parallel hierarchies, an *information hierarchy* and a *reasoning hierarchy,* often called the *causal hierarchy* (Pearl and Mackenzie, 2018; Bareinboim et al., 2021). On the reasoning side, this theory has crystalized three levels of reasoning with increased sophistication and proximity to human reasoning: associational, interventional and counterfactual, which are exemplified by the following canonical probabilities. *Associational* $Pr(y|x)$: probability of y given that x was observed (e.g., probability that a patient has a flu given they have a fever). *Interventional* $Pr(y_x)$: probability of y given that x was established by an intervention, which is different from $Pr(y|x)$ (e.g., seeing the barometer fall tells us about the weather but moving the barometer needle won't bring rain). *Counterfactual* $Pr(y_x|y', x')$: probability of y if we were to establish x given that neither x nor y are true (e.g., probability that a patient who did not take a vaccine and died would have recovered had they been vaccinated). On the information side, these forms of reasoning require different levels of knowledge, encoded as associational, causal and functional (mechanistic) models, with each class of models containing more information than the preceding one. In the framework of probabilistic graphical models (Koller and Friedman, 2009), such knowledge is encoded by Bayesian networks (Pearl, 1988; Darwiche, 2009), causal Bayesian networks (Pearl, 2000; Peters et al., 2017; Spirtes et al., 2000) and functional Bayesian networks (Balke and Pearl, 1995) also known as *structural causal models* (SCMs).

One utility of this theory has been recently crystallized through the *unit selection problem* introduced by Li and Pearl (2019) who motivated it using the problem of selecting customers to target by an encouragement offer for renewing a subscription. Let c denote the characteristics of a customer, x denote encouragement and y denote renewal. One can use counterfactuals to describe the different types of customers. A responder $(y_x, y'_{x'})$ would renew a subscription if encouraged

© 2023 .

but would not renew otherwise. An always-taker $(y_x, y_{x'})$ would always renew regardless of encouragement. An always-denier $(y'_x, y'_{x'})$ would always not renew regardless of encouragement. A contrarian $(y_{x'}, y'_x)$ would not renew if encouraged but would renew otherwise. One can then identify customers of interest by optimizing an expression, called a *benefit function* in (Li and Pearl, 2019), that includes counterfactual probabilities. In this example, the benefit function has the form $\beta Pr(\text{responder}|c) + \gamma Pr(\text{always-taker}|c) + \theta Pr(\text{always-denier}|c) + \delta Pr(\text{contrarian}|c)$ where $\beta, \gamma, \theta, \delta$ are corresponding benefits. In other words, one can use this expression to score customers with characteristics c so the most promising ones can be selected for an encouragement offer. When the above benefit function is contrasted with classical loss functions (for example, ones used to train neural networks), one sees a fundamental role for counterfactual reasoning as it gives us an ability to distinguish between objects (e.g., people, situations) depending on how they respond to a stimulus. This distinction sets apart counterfactual reasoning (third level of the causal hierarchy) from the more common, but less refined, associational reasoning (first level). It also sets it apart from interventional reasoning (second level) which is also not sufficient to make such distinctions.

Existing work on unit selection has focused on a very practical setting in which only the structure of an SCM is available together with some observational and experimental data (Li and Pearl, 2019, 2022a,b,c; Li et al., 2022b). Such data is usually not sufficient to obtain a fully specified SCM so one cannot obtain point values of the benefit function. Recent work has therefore focused on bounding probabilities of causation while tightening these bounds as much as possible (Dawid et al., 2017; Mueller et al., 2021), but with less attention dedicated to optimizing benefits based on these bounds; see (Li et al., 2022b,a) for a notable exception. In this paper, we complement this line of work by studying the unit selection problem from a different and computational direction. We are particularly interested in applying unit selection to structured units (e.g., decisions, policies, people, situations, regions, activities) that correspond to instantiations of multiple variables (called unit variables). We assume a fully specified SCM so we can obtain point values for any *causal objective function* as discussed in Section 2. By a causal objective function we mean any expression involving quantities from any level of the causal hierarchy (observational, interventional and counterfactual). This allows us to seek units that satisfy a broad class of conditions. Examples include: Which combination of activities are most effective to address a particular humanitarian need (human suffering, disease, hunger, privation)? Which regions should be focused on to reduce population movements among refugees? What incentive policy would keep customers engaged for the longest time? We then consider a particular but broad class of causal objective functions in Section 3 and formally define the computational problem of finding units that optimize these functions. We dedicate Section 4 to studying the complexity of unit selection in this setting where we show it has the same complexity as the classical *Maximum a Posteriori (MAP)* problem. We then provide an exact algorithm for solving the unit optimization problem in Section 5 by reducing it to a new problem that we call *Reverse-MAP*. We further characterize the complexity of our proposed algorithm using the notion of treewidth and provide some analysis on how its complexity can change depending on the specific objective function we use. We finally close with some concluding remarks in Section 6. Some proofs of our results are included in the main paper, the remaining ones can be found in the appendix.

2. Counterfactual Queries on Structural Causal Models

We review *structural causal models* (SCMs) in this section since the unit selection problem is defined on these models; see (Galles and Pearl, 1998; Halpern, 2000) for a comprehensive exposition.

We use uppercase letters (e.g., X) to denote variables and lowercase letters (e.g., x) to denote their states. We use bold uppercase letters (e.g., $\mathbf{X}$) to denote sets of variables and bold lowercase letters (e.g., $\mathbf{x}$) to denote their instantiations. The states of a binary variable X are denoted x and x'. We also write $x \in \mathbf{x}$ to mean that variable X has state x in instantiation $\mathbf{x}$ of variables $\mathbf{X}$.

An SCM has three components. First, a directed acyclic graph with its nodes representing variables. Root nodes are called *exogenous* and internal nodes are called *endogenous*. Second, a probability distribution $\theta(U)$ for each exogenous variable U in the model. Third, for each endogenous variable V with parents $\mathbf{P}$, the SCM has an equation, called a *structural equation*, which specifies a state for V for each instantiation $\mathbf{p}$ of its parents $\mathbf{P}$. Let $\mathbf{U}/\mathbf{V}$ be the exogenous/endogenous variables in an SCM. The distribution $Pr(\mathbf{U}, \mathbf{V})$ specified by the SCM is as follows: $Pr(\mathbf{u}, \mathbf{v}) = \prod_{u \in \mathbf{u}} \theta(u)$ if $\mathbf{V} = \mathbf{v}$ is implied by $\mathbf{U} = \mathbf{u}$ and the structural equations; otherwise, $Pr(\mathbf{u}, \mathbf{v}) = 0$.

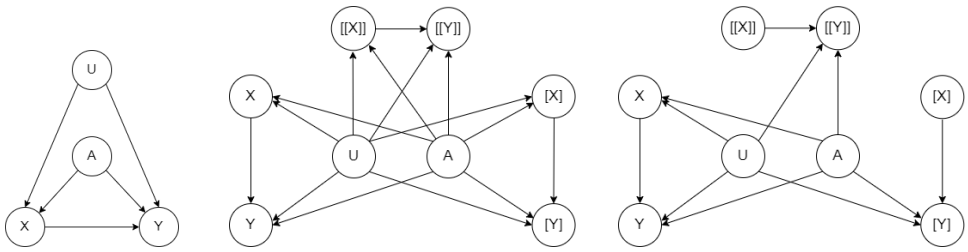
SCMs are a special type of Bayesian networks (Pearl, 1989; Darwiche, 2009) which require a conditional probability table (CPT) for each node in the network. In particular, for node V with parents $\mathbf{P}$, the CPT specifies the conditional distributions $Pr(V|\mathbf{P})$. A structural equation can be encoded as a CPT which satisfies $Pr(v|\mathbf{p}) \in \{0, 1\}$ for all v and $\mathbf{p}$. Such a CPT is said to be *functional* and this is why SCMs are sometimes called *functional Bayesian networks*.

A Bayesian network can only be used to compute *observational* probabilities such as $Pr(y|x)$ which is the probability of $Y = y$ given that we *observed* $X = x$. An SCM can also be used to compute *interventional probabilities* such as $Pr(y_x)$ which is the probability of $Y = y$ after *setting* $X = x$. An SCM can further be used to compute *counterfactual probabilities* such as $Pr(y_x, y'_{x'}|e)$ which is the probability of ($Y = y$ after setting $X = x$ and $Y = y'$ after setting $X = x'$) in a situation where we observe $E = e$.[1] We are particularly interested in this form of counterfactual probabilities as they will be used as ingredients in our objective functions. We next show how to compute such a counterfactual probability on an SCM by computing an observational probability on an auxiliary model. This will be essential for the constructions used later in the paper.

Consider the counterfactual probability $Pr(y_x, y'_{x'}|x, y)$ on the SCM in Figure 1(*a*). This query has three conflicting components: y_x, $y'_{x'}$ and x, y. The first two involve conflicting actions (x and x'). Moreover, the actions and outcomes in the first two components conflict with the observation in the third component (x, y). This is why computing counterfactual probabilities usually requires an auxiliary model that incorporates multiple worlds (real and imaginary) that all share the same causal mechanisms (exogenous variables). For the counterfactual queries we are interested in, an auxiliary model with three worlds will suffice as we discuss next.

Given an SCM G, its *triplet model* is another SCM constructed by having three copies G^1, G^2 and G^3 of G and then joining them so they share their exogenous variables; see Figure 1(*b*). If X is a variable in G^1, we will use $[X]$ to denote its copy in G_2 and $[[X]]$ to denote its copy in G_3. A triplet model is a special case of *parallel worlds models* (Avin et al., 2005) which also include *twin models* (Balke and Pearl, 1994).[2] We can now compute the counterfactual probability $Pr(y_x, y'_{x'}|x, y)$ on SCM G by operating on the triplet model as follows. First, we mutilate copies G^2 and G^3 in the triplet model by removing the edges pointing into variables $[X]$ and $[[X]]$ and setting $[X] = x$ and $[[X]] = x'$ (since we are intervening on these variables). The result is a *mutilated triplet model* shown in Figure 1(*c*). We can then evaluate $Pr(y_x, y'_{x'}|x, y)$ on the SCM

1. The class of *causal Bayesian networks* sits between Bayesian networks and functional Bayesian networks as it can be used to compute observational and interventional probabilities but not counterfactual ones (Pearl et al., 2000).
2. Twin models are sufficient to evaluate counterfactual probabilities like $Pr(y'_{x'}|x, y)$ and $Pr(y'_x, y_{x'})$ but not ones like $Pr(y_x, y'_{x'}|e)$ which we are interested in; see also (Tian and Pearl, 2000; Pearl et al., 2000).



(a) SCM (b) triplet model (c) mutilated triplet model

Figure 1: Reducing the counterfactual probability $Pr(y_x, y'_{x'}|x, y)$ on the model in (a) to an observational probability $Pr([y], [[y']] \mid [x], [[x']], x, y)$ on the model in (c).

G by computing the observational probability $Pr([y], [[y']] \mid [x], [[x']], x, y)$ on the mutilated triplet model. Intuitively, the triplet model can be viewed as capturing three worlds G^1, G^2 and G^3. World G^1 captures the observation x, y; world G^2 captures the intervention $X = x$, and world G^3 captures the intervention $X = x'$. This above treatment can be directly generalized to counterfactual queries of the form $Pr(\mathbf{y_x}, \mathbf{w_v}|\mathbf{e})$ where $\mathbf{E}, \mathbf{X}, \mathbf{Y}, \mathbf{V}, \mathbf{W}$ are sets of variables. It is precisely this class of counterfactual queries that we shall use in the rest of the paper, starting with the next section.

3. Causal Objective Functions and Unit Selection

A causal objective function can be any expression that involves observational, interventional or counterfactual probabilities where the goal of unit selection is to find objects (units) that optimize this function. However, inspired by (Li and Pearl, 2019), our treatment will be based on a specific class of causal objective functions which is a linear combination of counterfactual probabilities of the form $Pr({\mathbf{y}^i}_{\mathbf{x}^i}, {\mathbf{w}^i}_{\mathbf{v}^i}|\mathbf{e}^i, \mathbf{u})$ where $i = 1, \ldots, n$. We call $\mathbf{U}$ the *unit variables* since our goal is to find instantiations $\mathbf{u}$ of these variables (i.e., units) that optimize the objective function. Variables $\mathbf{X}^i\mathbf{V}^i$ represent treatments, variables $\mathbf{Y}^i\mathbf{W}^i$ represent outcomes, $(\mathbf{X}^i \cup \mathbf{V}^i) \cap (\mathbf{Y}^i \cup \mathbf{W}^i) = \emptyset$, and variables $\mathbf{E}^i$ represent evidence. Unit variables are shared by all components of the objective function but each component can have its own treatment, outcome and evidence variables.

We will further assume that unit variables $\mathbf{U}$ are exogenous in the SCM (i.e., root variables) while treatment, outcome and evidence variables are endogenous. However, not all exogenous variables need to be unit variables. This is consistent with the assumption in (Li and Pearl, 2022a) that unit variables (also called *characteristics*) cannot be descendants of treatment or outcome variables. This leads us to objective functions of the following form:[3]

$$L(\mathbf{u}) = \sum_{i=1}^{n} w_i \cdot Pr(\mathbf{y}^i_{\mathbf{x}^i}, \mathbf{w}^i_{\mathbf{v}^i}|\mathbf{e}^i, \mathbf{u}) \quad \text{where} \;\; w_i \geq 0 \;\; \text{and} \;\; \sum_{i=1}^{n} w_i = 1 \tag{1}$$

We can now formally define the unit selection inference problem on structural causal models.

Definition 1 (Unit Selection) *Given an SCM G, a subset $\mathbf{U}$ of its variables, and an objective function $L(\mathbf{U})$ such as Equation (1), the unit selection inference problem is to compute* $\text{argmax}_{\mathbf{u}}\, L(\mathbf{u})$.

3. The conditions we place on weights w_i are meant for convenience and they are not restrictive.

The benefit function discussed in (Li and Pearl, 2019) has the following form:

$$L(u) = \beta Pr(y_x, y'_{x'}|u) + \gamma Pr(y_x, y_{x'}|u) + \theta Pr(y'_x, y'_{x'}|u) + \delta Pr(y'_x, y_{x'}|u)$$

This class of objective functions falls as a special case of Equation (1) by setting $n = 4$, $\mathbf{E}^i = \emptyset$, $\mathbf{X}^i = \mathbf{V}^i = \{X\}$ and $\mathbf{Y}^i = \mathbf{W}^i = \{Y\}$ for $i = 1, \ldots, 4$, where X, Y are binary variables. That is, each component i of the objective function uses the same single, treatment variable X and the same single, outcome variable Y. A more general form was proposed in (Li and Pearl, 2022b) in which treatment X has values $x_1, \ldots, x_m$ and outcome Y has values $y_1, \ldots, y_k$ so the objective function can have up to k^m components, each corresponding to a distinct response type such as $Pr(y_{2_{x_1}}, y_{1_{x_2}}, y_{1_{x_3}}, y_{3_{x_4}}, y_{2_{x_5}}|u)$ when $k = 3$ and $m = 5$. This class of objective functions is more general than Equation (1) in that it allows one to express more response types but it assumes one treatment variable and one outcome variable. The class of objective functions we consider in Equation (1) allows compound treatments and outcomes. It also allows us to seek units from a particular group. For example, if A and B are two medications (binary treatments) and T and P refer to high temperature and high blood pressure (binary outcomes), and E is the age group with values $e_1, \ldots, e_4$, then the objective function can include terms such as $Pr((t, p')_{a,b}\ ,\ (t', p')_{a',b} \mid e_3, u)$, which is the probability that a member of the third age group would have high temperature and normal blood pressure if administered both medications and would have normal temperature and blood pressure if administered only the second medication. Moreover, since the objective function components can have different treatment and outcome variables, one can select units based on their responses to distinct stimuli (e.g., effect of one type of encouragement on membership renewal and the simultaneous effect of another type of encouragement on increased purchases).[4]

4. The Complexity of Unit Selection

We show next that unit selection is NP^{PP}-complete for the class of causal objective functions given in Equation (1). We also show that this problem is NP-complete when unit variables correspond to all exogenous variables in the SCM. We start by providing an efficient reduction from unit selection into a variant of the well-known MAP inference problem, which we call Reverse-MAP. We then follow by studying the complexity of Reverse-MAP and unit selection.

Recall that our goal is to find units $\mathbf{u}$ that maximize the value $L(\mathbf{u})$ of the objective function. The first step in solving this optimization problem is to be able to evaluate the objective $L(\mathbf{u})$. We next show a construction which allows us to evaluate $L(\mathbf{u})$ by evaluating a single observational probability involving unit variables $\mathbf{U}$ but on an extended and mutilated model. This construction will serve two purposes. First, it will permit us to characterize the complexity of unit selection when using objective functions in the form of Equation (1). Second, we will later use the construction to develop a specific algorithm for solving the unit selection problem using these objective functions.

Consider each term $Pr(\mathbf{y}^i_{\mathbf{x}^i}, \mathbf{w}^i_{\mathbf{v}^i}|\mathbf{e}^i, \mathbf{u})$ in Equation (1). We reviewed in Section 2 how this quantity can be reduced to a classical conditional probability on a triplet model G^i. The next step is to encode a linear combination of these conditional probabilities as a conditional probability on some model G'. This is done using the following construction.

4. Going beyond the form in Equation (1), one can use causal objective functions with ingredients like the probability of a patient being a responder given they are not a contrarian, $Pr(y_x, y'_{x'}|\neg(y'_x, y_{x'}))$, and the probability that a patient would not have had a stroke if they had been on a diet or, alternatively, had they exercised, $Pr(y'_d \vee y'_e|y, d', e')$. Such general quantities have not been treated in the literature but some discussions have argued for their significance and treated special cases; e.g., (Pearl, 2017).

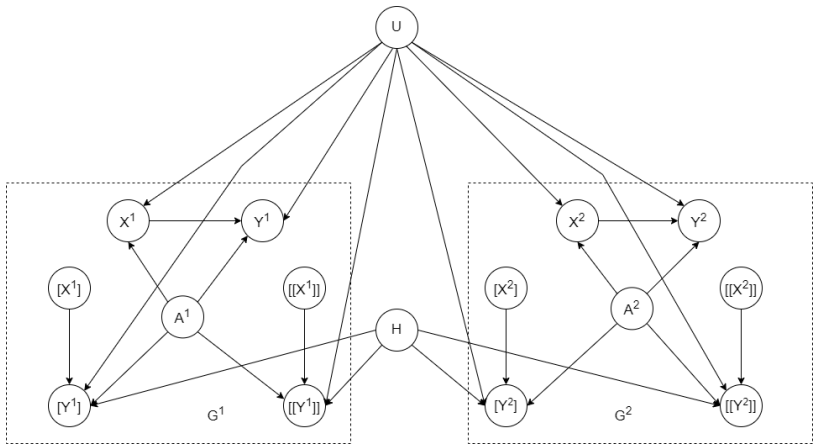


Figure 2: An objective model with two components for the SCM in Figure 1(*a*).

Definition 2 (Objective Model) *Consider an SCM G with parameters θ and the objective function L in Equation (1). The objective model G' for $\langle G, L\rangle$ has parameters θ' and constructed as follows:*

1. *Construct a triplet model G^i of G for each component $Pr(\mathbf{y}^i_{\mathbf{x}^i}, \mathbf{w}^i_{\mathbf{v}^i}|\mathbf{e}^i, \mathbf{u})$ in L (see Section 2).*

2. *Join $G^1, G^2, \ldots, G^n$ so that their unit variables $\mathbf{U}$ are shared.*

3. *Add a node H as a parent of all outcome nodes $\mathbf{Z} = \{\mathbf{Y}^i, \mathbf{W}^i\}_{i=1}^n$. Node H has states $h_1, \ldots, h_n$ with prior $\theta'(h_i) = w_i$. Each node $Z \in \mathbf{Z}$ now has parents $\mathbf{P}_Z \cup \{H\}$, where $\mathbf{P}_Z$ are the parents of Z before node H is added. Let z^i be the state of Z in instantiation $\mathbf{y}^i\mathbf{w}^i$. The new CPT for Z is:*

$\mathbf{P}_Z$	H	Z	$\theta'(Z\|\mathbf{P}_Z, H)$
$\mathbf{p}$	h_i	z^i	$\theta(z^i\|\mathbf{p})$
$\mathbf{p}$	h_i	$\bar{z}^i$	$\theta(\bar{z}^i\|\mathbf{p})$
$\mathbf{p}$	$\bar{h}_i$	z^i	1.0
$\mathbf{p}$	$\bar{h}_i$	$\bar{z}^i$	0.0

Here, $\bar{z}^i, \bar{h}_i$ denote any states of variables Z, H that are distinct from states z^i, h_i.

We will refer to variable H in Definition 2 as the *mixture variable* since it encodes a mixture of the objective function terms. We will also say that the objective model G' constructed by this definition has n *components*. This model can also be viewed as an SCM since the CPTs of its internal nodes are all functional. The CPTs for variables $\mathbf{Y}^i\mathbf{W}^i$ in model G' reduce to the original CPTs of these variables in SCM G when $H = h_i$, and imply $\mathbf{Y}^i\mathbf{W}^i = \mathbf{y}^i\mathbf{w}^i$ when $H \neq h_i$. The following theorem relates the objective $L(\mathbf{u})$ in SCM G to a classical probability in the objective model G'. The proof of theorem can be found in Appendix A.

Theorem 3 *Consider an SCM G with unit variables $\mathbf{U}$. Let L be the objective function in Equation (1) and let $\mathbf{X} = \{\mathbf{X}^i\}_{i=1}^n$, $\mathbf{Y} = \{\mathbf{Y}^i\}_{i=1}^n$, $\mathbf{W} = \{\mathbf{W}^i\}_{i=1}^n$, $\mathbf{V} = \{\mathbf{V}^i\}_{i=1}^n$ and $\mathbf{E} = \{\mathbf{E}^i\}_{i=1}^n$. If G' is an objective model for $\langle G, L\rangle$ and has distribution Pr', then $L(\mathbf{u}) = Pr'(\mathbf{y}, \mathbf{w}|\mathbf{x}, \mathbf{v}, \mathbf{e}, \mathbf{u})$.*

Consider the SCM in Figure 1(*a*) and the causal objective function $L(u) = w_1 \cdot Pr(y_x, y'_{x'}|u) + w_2 \cdot Pr(y_x, y_{x'}|u)$. Figure 2 shows a corresponding objective model G' constructed according to Definition 2. We now have $L(u) = Pr'([y^1], [[\bar{y}^1]], [y^2], [[y^2]] \mid [x^1], [[\bar{x}^1]], [x^2], [[\bar{x}^2]]; u)$.

Theorem 3 suggests that we can optimize the objective function $L(\mathbf{u})$ on an SCM G by computing the instantiation $\operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{y}|\mathbf{x}, \mathbf{e}, \mathbf{u})$ on an objective model G'. The is similar to the classical MAP problem on model G', except that the optimized variables $\mathbf{U}$ appear after the conditioning operator instead of before it. This leads to our definition of the Reverse-MAP problem.

Definition 4 (Reverse-MAP) *Consider an SCM G with distribution Pr and suppose $\mathbf{U}, \mathbf{E}_1, \mathbf{E}_2$ are disjoint sets of variables in G. The Reverse-MAP instantiation for variables $\mathbf{U}$ and instantiations $\mathbf{e}_1, \mathbf{e}_2$ is defined as follows:* $\mathrm{RMAP}(\mathbf{U}, \mathbf{e}_1, \mathbf{e}_2) \triangleq \operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{e}_1 \mid \mathbf{u}, \mathbf{e}_2)$.

To see the connection between Reverse-MAP and MAP, note that $\operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) = \operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{u}, \mathbf{e}_1, \mathbf{e}_2)/Pr(\mathbf{u}, \mathbf{e}_2)$ where $\operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{u}, \mathbf{e}_1, \mathbf{e}_2) = \operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{u}|\mathbf{e}_1, \mathbf{e}_2)$ is the known MAP problem (Pearl, 1989). In general, the MAP instantiation $\operatorname{argmax}_{\mathbf{u}} Pr(\mathbf{u}, \mathbf{e}_1, \mathbf{e}_2)$ is not the Reverse-MAP instantiation since $Pr(\mathbf{u}, \mathbf{e}_2)$ also depends on $\mathbf{U}$; see Appendix B for a concrete example that illustrates this. We now have the following result, proven in Appendix C.

Corollary 5 *There are polynomial-time reductions between the Reverse-MAP problem and the unit selection problem with objective functions in the form of Equation (1).*

We next characterize the complexity of Reverse-MAP under different conditions. Consider a decision version of the problem, D-Reverse-MAP, defined as follows.

Definition 6 (D-Reverse-MAP) *Given an SCM that specifies distribution Pr, some target variables $\mathbf{U}$, some evidence $\mathbf{e}_1, \mathbf{e}_2$ and a threshold p, the D-Reverse-MAP problems asks whether there is an instantiation $\mathbf{u}$ of $\mathbf{U}$ such that $Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) > p$?*

The next theorem shows that D-Reverse-MAP is $\mathrm{NP}^{\mathrm{PP}}$-complete, like classical MAP (Park and Darwiche, 2004). Its proof can be found in Appendix D.

Theorem 7 *D-Reverse-MAP is $\mathrm{NP}^{\mathrm{PP}}$-complete.*

We can now characterize the complexity of unit selection using Theorem 7 and Corollary 5.

Corollary 8 *Unit selection is $\mathrm{NP}^{\mathrm{PP}}$-complete assuming the objective function in Equation (1).*

In an SCM, exogenous (root) variables represent all uncertainties in the model and the endogenous (internal) variables are uniquely determined by exogenous variables. This property of SCMs significantly reduces the complexity of unit selection when the unit variables correspond to the SCM exogenous variables. This is implied by the following result which is proven in Appendix E.

Theorem 9 *D-Reverse-MAP is NP-complete if its target variables are the SCM root variables.*

Corollary 10 *Unit selection is NP-complete when the unit variables are the SCM exogenous (root) variables, assuming the objective functions in Equation (1).*

5. Unit Selection using Variable Elimination

Section 4 provided a reduction from unit selection on an SCM to Reverse-MAP on an objective model. In Section 5.1, we provide a variable elimination (VE) algorithm for Reverse-MAP which can be applied to the objective model to solve unit selection. In Section 5.2, we analyze the complexity of this method and compare it to the complexity of Reverse-MAP on the underlying SCM.

5.1. Reverse-MAP using Variable Elimination

Our VE algorithm for Reverse-MAP will employ the same machinery and techniques used in the VE algorithm for classical MAP (Dechter, 1999). Hence, we will first review the VE algorithm for MAP using the treatment in (Darwiche, 2009, Ch 10) and then discuss the algorithm for Reverse-MAP.

The VE algorithm is based on the notion of a *factor* $f(\mathbf{X})$ which maps each instantiation $\mathbf{x}$ of variables $\mathbf{X}$ into a non-negative number $f(\mathbf{x})$. VE employs a number of factor operations including multiplying two factors ($f \cdot g$), summing out a variable from a factor ($\sum_X f$), maximizing out a variable from a factor ($\max_X f$), and dividing two factors (f/g). Let G be an SCM and assume its variables $\mathbf{Z}$ are partitioned into three disjoint sets $\mathbf{U}, \mathbf{V}, \mathbf{E}$, where $\mathbf{U}$ are the *target variables* and $\mathbf{E}$ are the *evidence variables.* Let $\mathbf{S} = \mathbf{Z} \backslash \mathbf{U}$ in the following discussion. We will treat the CPT of each variable Z in SCM G as a factor over Z and its parents $\mathbf{P}$, denoted $f_Z(Z\mathbf{P})$. The SCM distribution is then $Pr(\mathbf{Z}) = \prod_{Z \in \mathbf{Z}} f_Z$. We capture evidence $\mathbf{e}$ by creating an evidence factor $\lambda_e(E)$ for each $e \in \mathbf{e}$ with $\lambda_e(e') = 1$ if $e' = e$ and $\lambda_e(e') = 0$ otherwise. The *MAP probability* is then given by[5]

$$\mathrm{MAP}_p(\mathbf{U}, \mathbf{e}) = \max_{\mathbf{u}} Pr(\mathbf{u}, \mathbf{e}) = \max_{\mathbf{u}} \sum_{\mathbf{v}} Pr(\mathbf{u}, \mathbf{v}, \mathbf{e}) = \max_{\mathbf{U}} \sum_{\mathbf{S}} \prod_{Z \in \mathbf{Z}} f_Z \prod_{e \in \mathbf{e}} \lambda_e(E) \tag{2}$$

This is in contrast to the *MAP instantiation* which is $\mathrm{argmax}_{\mathbf{u}}\, Pr(\mathbf{u}, \mathbf{e})$. With some minor bookkeeping, the VE algorithm for computing the MAP probability can also return a MAP instantiation; see, e.g., (Darwiche, 2009, Ch 10). Hence, we will focus next on computing the MAP probability.

Consider the SCM in Figure 3(*a*) and suppose $\mathbf{U} = \{A, B\}$ and $\mathbf{e} = \{E = e\}$. We then have

$$\mathrm{MAP}_p(AB, e) = \max_{AB} \sum_{CDE} f_A(A) f_B(AB) f_C(AC) f_D(BCD) f_E(CE) \lambda_e(E) \tag{3}$$

A naive evaluation of this expression will multiply all factors above to yield a factor $f(ABCDE)$ over all model variables, and then compute $\max_{AB} \sum_{CDE} f(ABCDE)$, leading to $O(n \exp(n))$ complexity where n is the number of model variables. The VE algorithm tries to compute this expression more efficiently with pseudocode provided in Algorithm 1 (MAP_VE). The product of factors $\mathcal{F}$ on Line 4 represents the join distribution $Pr(\mathbf{Z}, \mathbf{e})$ so we first sum out variables $\mathbf{S}$ from $\mathcal{F}$ on Line 5 to compute a set of factors $\mathcal{G}$ whose product represents the marginal $Pr(\mathbf{U}, \mathbf{e})$. We then maximize out variables $\mathbf{U}$ from $\mathcal{G}$ on Line 6 leading to a scalar factor p that contains the MAP probability (see Footnote 5). Algorithm 1 eliminates variables one by one using Algorithm 2 and a total variable order $\pi = \langle \pi_{\mathbf{S}}, \pi_{\mathbf{U}} \rangle$, known as an *elimination order.* MAP_VE requires variables $\mathbf{U}$ to appear last in order π since summation does not commute with maximization. An order that satisfies this constraint is known as a $\mathbf{U}$-*constrained elimination order*. The complexity of MAP_VE depends on the used elimination order π. In each elimination step of Algorithm 2, we multiply all factors

5. The left side of Equation 2 is a scalar (probability) while the right side is a factor over an empty set of variables, which is called a *scalar factor.* Such a factor maps only one instantiation, the empty one, to a scalar.

Algorithm 1 MAP_VE($G, \mathbf{U}, \mathbf{e}$)

Input: SCM G, target variables $\mathbf{U}$, evidence $\mathbf{e}$
Output: scalar factor containing $\mathrm{MAP}_p(\mathbf{U}, \mathbf{e})$

1: **procedure** MAIN
2: $\pi_{\mathbf{S}} \leftarrow$ an elimination order for non-target variables
3: $\pi_{\mathbf{U}} \leftarrow$ an elimination order for target variables $\mathbf{U}$
4: $\mathcal{F} \leftarrow \{f : f$ is a CPT of SCM $G\} \cup \{\lambda_e : \lambda_e$ is an evidence factor for $e \in \mathbf{e}\}$
5: $\mathcal{G} \leftarrow$ ELIMINATE($\sum, \mathcal{F}, \pi_{\mathbf{S}}$)
6: $p \leftarrow$ ELIMINATE($\max, \mathcal{G}, \pi_{\mathbf{U}}$)
7: **return** p
8: **end procedure**

Algorithm 2 Eliminating Variables using Sum or Max

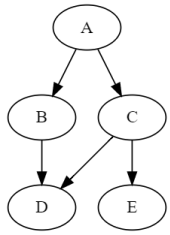
Input: a set of factors $\mathcal{F}$, a total variable order π, an operation $\bigcirc \in \{\sum, \max\}$
Output: a set of factors

1: **procedure** ELIMINATE($\bigcirc, \mathcal{F}, \pi$)
2: **for** $i = 1$ to length of order π **do**
3: $V \leftarrow i^{th}$ variable in order π
4: $\mathcal{G} \leftarrow$ factors in $\mathcal{F}$ that mentions variable V
5: $f_i \leftarrow \prod_{f \in \mathcal{G}} f$
6: $f_i \leftarrow \bigcirc_V f_i$
7: replace factors $\mathcal{G}$ in $\mathcal{F}$ with factor f_i
8: **end for**
9: **return** $\mathcal{F}$
10: **end procedure**

that mention variable $\pi(i)$ to obtain factor f_i on Line 5. The variables in f_i are called a cluster $\mathbf{C}_i$ so eliminating variables $\pi(1), \ldots, \pi(n)$ induces clusters $\mathbf{C}_1, \ldots, \mathbf{C}_n$. The *width* w of elimination order π is the size of largest cluster minus one and the complexity of MAP_VE is $O(n \exp(w))$.

Below is a trace of MAP_VE when computing the MAP probability in Equation (3) using the elimination order $\pi = E, D, C, B, A$. The trace shows that MAP_VE evaluates the following factorized expression and that the width of order π is 2:

$$\mathrm{MAP}_p(AB, e) = \max_A f_A(A) \left[\max_B f_B(AB) \left[\sum_C f_C(AC) \left[\sum_D f_D(BCD) \left[\sum_E f_E(CE) \lambda_E \right] \right] \right] \right]$$



(*a*) SCM

var	factors $\mathcal{G}$ (Line 4, Alg 2)	new factor f_i (Line 5, Alg 2)	$\mathbf{C}_i$
E	$f_E(CE)\ \lambda_E$	$f_1 = \sum_E f_E(CE)\ \lambda_E$	CE
D	$f_D(BCD)$	$f_2 = \sum_D f_D(BCD)$	BCD
C	$f_C(AC)\ f_1(C)\ f_2(BC)$	$f_3 = \sum_C f_C(AC)\ f_1(C)\ f_2(BC)$	ABC
B	$f_B(AB)\ f_3(AB)$	$f_4 = \max_B f_B(AB)\ f_3(AB)$	AB
A	$f_A(A)\ f_4(A)$	$p = \max_A f_A(A)\ f_4(A)$	A

(*b*) elimination trace

Choosing a good elimination order is critical for the complexity of VE. The *treewidth* of an SCM G is defined as the minimum width attained by any elimination order. Since MAP requires

$\mathbf{U}$-constrained orders, the $\mathbf{U}$-*constrained treewidth* of G is defined as the minimum width attained by any $\mathbf{U}$-constrained elimination order (Park and Darwiche, 2004).

We are now ready to introduce our VE algorithm for Reverse-MAP. Again, we assume that the model variables $\mathbf{Z}$ are partitioned into disjoint sets $\mathbf{U}, \mathbf{V}, \mathbf{E}$, where $\mathbf{U}$ are the target variables and $\mathbf{S} = \mathbf{Z} \setminus \mathbf{U}$. But we further partition the evidence variables $\mathbf{E}$ into $\mathbf{E}_1$ and $\mathbf{E}_2$. Again, we focus on computing the Reverse-MAP probability $\mathrm{RMAP}_p(\mathbf{U}, \mathbf{e}_1, \mathbf{e}_2)$ instead of the instantiation:

$$\max_{\mathbf{u}} Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) = \frac{Pr(\mathbf{u}, \mathbf{e}_1, \mathbf{e}_2)}{Pr(\mathbf{u}, \mathbf{e}_2)} = \frac{\sum_{\mathbf{v}} Pr(\mathbf{u}, \mathbf{v}, \mathbf{e}_1, \mathbf{e}_2)}{\sum_{\mathbf{v}} Pr(\mathbf{u}, \mathbf{v}, \mathbf{e}_2)} = \max_{\mathbf{U}} \frac{\sum_{\mathbf{S}} \prod_{Z \in \mathbf{Z}} f_Z \prod_{e \in \mathbf{e}_1 \cup \mathbf{e}_2} \lambda_e}{\sum_{\mathbf{S}} \prod_{Z \in \mathbf{Z}} f_Z \prod_{e \in \mathbf{e}_2} \lambda_e}$$

Our algorithm, called RMAP_VE, runs two passes of elimination as shown in Algorithm 3. In the first pass (Line 4), we sum out variables $\mathbf{S}$ under evidence $\mathbf{e}_1, \mathbf{e}_2$ and in the second pass (Line 5), we sum out variables $\mathbf{S}$ under evidence $\mathbf{e}_2$. This leads to two sets of factors $\mathcal{G}_1$ and $\mathcal{G}_2$ which correspond to marginal distributions $Pr(\mathbf{U}, \mathbf{e}_1, \mathbf{e}_2)$ and $Pr(\mathbf{U}, \mathbf{e}_2)$. Now we need to divide $Pr(\mathbf{U}, \mathbf{e}_1, \mathbf{e}_2)$ and $Pr(\mathbf{U}, \mathbf{e}_2)$ to compute $Pr(\mathbf{e}_1|\mathbf{U}, \mathbf{e}_2)$. We next show that this can be done efficiently by "dividing" $\mathcal{G}_1$ and $\mathcal{G}_2$ as shown on Line 8. The key idea is that if we run the two passes of elimination according to the same elimination order, then there will be a one-to-one correspondence between the factors in $\mathcal{G}_1$ and $\mathcal{G}_2$. Let (g_1^i, g_2^i) be the corresponding pairs of factors for $i = 1, \ldots, k$ where $k = |\mathcal{G}_1| = |\mathcal{G}_2|$. What we need is $\left(\prod_{i=1}^n g_1^i\right) / \left(\prod_{i=1}^n g_2^i\right)$ since this represents $Pr(\mathbf{e}_1|\mathbf{U}, \mathbf{e}_2)$. But due to the mentioned correspondence, this equals $\prod_{i=1}^n g_1^i / g_2^i$. Thus, we can divide each pair of corresponding factors to obtain the set of factors $\mathcal{G}$ as done on Line 8. We finally maximize out target variables $\mathbf{U}$ from $\mathcal{G}$ to obtain the Reverse-MAP probability (Line 9).

Algorithm 3 RMAP_VE($G, \mathbf{U}, \mathbf{e}_1, \mathbf{e}_2$)

Input: SCM G, target variables $\mathbf{U}$, evidence $\mathbf{e}_1$ and $\mathbf{e}_2$
Output: scalar factor containing $\mathrm{RMAP}_p(\mathbf{U}, \mathbf{e}_1, \mathbf{e}_2)$

1: **procedure** MAIN
2: $\pi_{\mathbf{S}} \leftarrow$ an elimination order for non-target variables
3: $\pi_{\mathbf{U}} \leftarrow$ an elimination order for target variables $\mathbf{U}$
4: $\mathcal{F}_1 \leftarrow \{f : f$ is a CPT of SCM $G\} \cup \{\lambda_e : \lambda_e$ is an evidence factor for $e \in \mathbf{e}_1, \mathbf{e}_2\}$
5: $\mathcal{F}_2 \leftarrow \{f : f$ is a CPT of SCM $G\} \cup \{\lambda_e : \lambda_e$ is an evidence factor for $e \in \mathbf{e}_2\}$
6: $\mathcal{G}_1 \leftarrow$ ELIMINATE($\sum, \mathcal{F}_1, \pi_{\mathbf{S}}$)
7: $\mathcal{G}_2 \leftarrow$ ELIMINATE($\sum, \mathcal{F}_2, \pi_{\mathbf{S}}$)
8: $\mathcal{G} \leftarrow \{g_1/g_2 : g_1, g_2$ are corresponding factors in $\mathcal{G}_1, \mathcal{G}_2\}$
9: $p \leftarrow$ ELIMINATE($\max, \mathcal{G}, \pi_{\mathbf{U}}$)
10: **return** p
11: **end procedure**

RMAP_VE has the same complexity as MAP_VE if both use the same elimination order. Suppose there are k factors in $\mathcal{G}_1/\mathcal{G}_2/\mathcal{G}$ and the largest factor has size c. The cost of division on Line 8 is $O(k \exp(c))$ while the cost of maximization on Line 9 is at least $O(k \exp(c))$ so the cost of division is dominated by the cost of maximization. Hence, the complexity of RMAP_VE is still $O(n \exp(w))$ where n is the number of variables and w is the width of used $\mathbf{U}$-constrained order π.

5.2. Bounding the Complexity of Unit Selection using Variable Elimination

We can solve unit selection using Algorithm VE_RMAP by utilizing the reduction to Reverse-MAP provided in Section 4. However, this method requires us to operate on an objective model which is larger and denser than the given SCM, causing variable elimination to be more expensive. This raises the question: how expensive is unit selection on the objective model in comparison to Reverse-MAP (or MAP)[6] on the SCM? In particular, is unit selection always tractable on the objective model when Reverse-MAP is tractable on the SCM? Our goal in this section is to shed light on these questions using the lens of treewidth which is commonly used to analyze elimination algorithms.

Our starting point is to study the treewidth of an objective model in relation to the treewidth of its underlying SCM. We will base our study on the techniques and results reported in (Han et al., 2022) which studied the complexity of counterfactual reasoning. In particular, given an elimination order π of SCM G, we next show how to construct an elimination order π' for the objective model G' while providing a bound on the width of order π' in terms of the width of order π. Recall that we use $[X]$ and $[[X]]$ to denote the copies of variable X in a triplet model where $X = [X] = [[X]]$ if X is exogenous. Moreover, if U is a unit variable, then $U = U^1 = \cdots = U^n$ in an objective model.

Definition 11 *Let G be an SCM and G' be a corresponding objective model with n components. If π is an elimination order for G, the corresponding elimination order π' for G' is obtained by replacing each non-unit variable X in π by $X^1, \ldots, X^n, [X^1], \ldots, [X^n], [[X^1]], \ldots, [[X^n]]$ then appending the mixture variable H to the end of the order.*

Consider the elimination order $\pi = A, X, Y, U$ for the SCM in Figure 1(*a*). The corresponding elimination order π' for the objective model in Figure 2 is as follows:

$$\pi' = A^1, A^2, X^1, X^2, [X^1], [X^2], [[X^1]], [[X^2]], Y_1, Y_2, [Y_1], [Y_2], [[Y_1]], [[Y_2]], U, H$$

The following bound (Theorem 13) follows from Lemma 12 and a result in (Han et al., 2022) which concerns *N-twin networks:* N copies of the SCM that share *some* of their root variables. An objective model with n components can be viewed as a $3n$-twin network but with an additional mixture node H. Lemma 12 and Theorem 13 are proven in Appendix F and Appendix G.

Lemma 12 *Consider an SCM G and suppose SCM G' is obtained from G by adding a root node H as a parent of some nodes in G. If π is an elimination order for G and has width w, then $\pi' = \langle \pi, H \rangle$ is an elimination order for G' and has width $w' \leq w + 1$.*

Theorem 13 *Consider an SCM G and a corresponding objective model G' with n components. Let π be an elimination order for G and let π' be the corresponding elimination order for G'. If π has width w and π' has width w', then $w' \leq 3n(w + 1)$.*

Corollary 14 *If w is the treewidth of an SCM G and w' is the treewidth of a corresponding objective model G' with n components, then $w' \leq 3n(w + 1)$.*

As mentioned earlier, when solving Reverse-MAP using VE, we can only use $\mathbf{U}$-constrained elimination orders in which unit variables $\mathbf{U}$ appear last. Hence, a $\mathbf{U}$-constrained elimination order for an objective model must place the mixture variable H before $\mathbf{U}$. This leads to the next definition.

6. In the context of variable elimination, MAP and Reverse-MAP have the same complexity when applied to the same SCM using the same target variables.

Definition 15 *Let G be an SCM with unit variables $\mathbf{U}$ and let G' be a corresponding objective model with n components. If π is a $\mathbf{U}$-constrained elimination order for G, the corresponding $\mathbf{U}$-constrained elimination order π' for G' is obtained by replacing each non-unit variable X in π by $X^1, \ldots, X^n, [X^1], \ldots, [X^n], [[X^1]], \ldots, [[X^n]]$ then inserting mixture variable H just before $\mathbf{U}$.*

Consider the $\mathbf{U}$-constrained order $\pi = A, X, Y, U$ for the SCM in Figure 1(*a*). The corresponding $\mathbf{U}$-constrained elimination order for the objective model in Figure 2 is

$$\pi' = A^1, A^2, X^1, X^2, [X^1], [X^2], [[X^1]], [[X^2]], Y_1, Y_2, [Y_1], [Y_2], [[Y_1]], [[Y_2]], H, U$$

We now have the following bound on the $\mathbf{U}$-constrained treewidth of objective models, which is somewhat unexpected when compared to the bound on (unconstrained) treewidth. In particular, while the bound on treewidth grows linearly in the number of components in the objective model, the bound on $\mathbf{U}$-constrained treewidth is independent of such a number. Moreover, the bound on $\mathbf{U}$-constrained treewidth depends on the number of unit variables which is not the case for treewidth.

Theorem 16 *Let G be an SCM with unit variables $\mathbf{U}$ and let G' be a corresponding objective model. If π is a $\mathbf{U}$-constrained elimination order for G with width w and π' is the corresponding $\mathbf{U}$-constrained elimination order for G' with width w', then $w' \leq \max(3w+3, |\mathbf{U}|)$. If the objective function in Equation (2) has one outcome variable ($\mathbf{Y}^i = \mathbf{W}^i = \{Y\}$ for all i), then $w' \leq 3w + 3$.*

Corollary 17 *Let G be an SCM with unit variables $\mathbf{U}$ and let G' be a corresponding objective model. If w and w' are the $\mathbf{U}$-constrained treewidths of G and G', then $w' \leq \max(3w + 3, |\mathbf{U}|)$. Moreover, if the objective function in Equation (2) has a single outcome variable, then $w' \leq 3w+3$.*

The causal objective function in (Li and Pearl, 2019), called the benefit function, has a single outcome variable which is shared by all components of the function. Hence, it satisfies the condition for the second bound in Corollary 17 which is independent of the number of unit variables. When this condition is met, unit selection using RMAP_VE on an objective model can be solved in polytime if MAP (Reverse-MAP) can be solved in polytime on the underlying SCM (assuming the target variables for MAP and Reverse-MAP are the unit variables). Otherwise, the bound in Corollary 17 does not guarantee this. We close by recalling that MAP, Reverse-MAP and unit selection using objective functions in the form of Equation (1) are all NP^{PP}-complete as shown earlier.

6. Conclusion

We studied the unit selection problem in a computational setting which complements existing studies. We assumed a fully specified structural causal model so we can compute point values of causal objective functions, allowing us to entertain a broader class of functions than is normally considered. We showed that the unit selection problem with this class of objective functions is NP^{PP}-complete, similar to the classical MAP problem, and identified an intuitive condition under which it is NP-complete. We further provided an exact algorithm for the unit selection problem based on variable elimination and characterized its complexity in terms of treewidth, while relating this complexity to that of MAP inference. In the process, we defined a new inference problem, Reverse-MAP, which is also NP^{PP}-complete but captures the essence of unit selection more than MAP does.

References

Chen Avin, Ilya Shpitser, and Judea Pearl. Identifiability of path-specific effects. In *IJCAI*, pages 357–363. Professional Book Center, 2005.

Alexander Balke and Judea Pearl. Probabilistic evaluation of counterfactual queries. In *AAAI*, pages 230–237. AAAI Press / The MIT Press, 1994.

Alexander Balke and Judea Pearl. Counterfactuals and policy analysis in structural models. In *UAI*, pages 11–18. Morgan Kaufmann, 1995.

E. Bareinboim, Juan David Correa, D. Ibeling, and Thomas F. Icard. On Pearl's hierarchy and the foundations of causal inference. 2021. Technical Report, R-60, Colombia University.

Adnan Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 2009.

Philip Dawid, Monica Musio, and Rossella Murtas. The probability of causation. *Law, Probability and Risk*, (16):163–179, 2017.

Rina Dechter. Bucket elimination: A unifying framework for reasoning. *Artif. Intell.*, 113(1-2): 41–85, 1999.

David Galles and Judea Pearl. An axiomatic characterization of causal counterfactuals. *Foundations of Science*, 3(1):151–182, 1998.

Joseph Y Halpern. Axiomatizing causal reasoning. *Journal of Artificial Intelligence Research*, 12: 317–337, 2000.

Yunqiu Han, Yizuo Chen, and Adnan Darwiche. On the complexity of counterfactual reasoning. In *A causal view on dynamical systems, NeurIPS 2022 workshop*, 2022. URL https://openreview.net/forum?id=yf9txIkzpO2.

Daphne Koller and Nir Friedman. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press, 2009.

A. Li, S. Jiang, Y. Sun, and J. Pearl. Learning probabilities of causation from finite population data. Technical Report R-519, <http://ftp.cs.ucla.edu/pub/stat_ser/r519.pdf>, Department of Computer Science, University of California, Los Angeles, CA, 2022a.

Ang Li and Judea Pearl. Unit selection based on counterfactual logic. In *IJCAI*, pages 1793–1799. ijcai.org, 2019.

Ang Li and Judea Pearl. Unit selection with causal diagram. In *AAAI*, pages 5765–5772. AAAI Press, 2022a.

Ang Li and Judea Pearl. Unit selection with nonbinary treatment and effect. *CoRR*, abs/2208.09569, 2022b.

Ang Li and Judea Pearl. Unit selection: Case study and comparison with A/B test heuristic. *CoRR*, abs/2210.05030, 2022c.

Ang Li, Song Jiang, Yizhou Sun, and Judea Pearl. Unit selection: Learning benefit function from finite population data. *CoRR*, abs/2210.08203, 2022b.

Michael L Littman, Judy Goldsmith, and Martin Mundhenk. The computational complexity of probabilistic planning. *Journal of Artificial Intelligence Research*, 9:1–36, 1998.

S. Mueller, A. Li, and J. Pearl. Causes of effects: Learning individual responses from population data. Technical Report R-505, <http://ftp.cs.ucla.edu/pub/stat_ser/r505.pdf>, Department of Computer Science, University of California, Los Angeles, CA, 2021. Forthcoming, Proceedings of IJCAI-2022.

James D Park. Map complexity results and approximation methods. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 388–396, 2002.

James D. Park and Adnan Darwiche. Complexity results and approximation strategies for MAP explanations. *J. Artif. Intell. Res.*, 21:101–133, 2004.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. MK, 1988.

Judea Pearl. *Probabilistic reasoning in intelligent systems - networks of plausible inference*. Morgan Kaufmann series in representation and reasoning. Morgan Kaufmann, 1989.

Judea Pearl. *Causality*. Cambridge University Press, 2000.

Judea Pearl. Physical and metaphysical counterfactuals: Evaluating disjunctive actions. *Journal of Causal Inference*, 5(2), 2017. doi: https://doi.org/10.1515/jci-2017-0018.

Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. Basic Books, 2018.

Judea Pearl et al. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19(2), 2000.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, 2017.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search, Second Edition*. Adaptive computation and machine learning. MIT Press, 2000. ISBN 978-0-262-19440-2.

Jin Tian and Judea Pearl. Probabilities of causation: Bounds and identification. *Ann. Math. Artif. Intell.*, 28(1-4):287–313, 2000.

Appendix A. Proof of Theorem 3

The proof of this theorem requires a lemma which requires the following definition. We will say that a set of variables $\mathbf{Z}$ *decomposes* a DAG if removing the outgoing edges from $\mathbf{Z}$ splits the DAG into at least two disconnected components.

Lemma 18 *Consider an SCM G with distribution Pr and three disjoint set of variables $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$. Suppose $\mathbf{Z}$ decomposes G into disconnected components G_1 and G_2. If $\mathbf{X}_1, \mathbf{Y}_1$ are subsets of $\mathbf{X}, \mathbf{Y}$ pertaining to G_1, and $\mathbf{X}_2, \mathbf{Y}_2$ are subsets of $\mathbf{X}, \mathbf{Y}$ pertaining to G_2, then $Pr(\mathbf{y}|\mathbf{x}, \mathbf{z}) = Pr(\mathbf{y}_1|\mathbf{x}_1, \mathbf{z})Pr(\mathbf{y}_2|\mathbf{x}_2, \mathbf{z})$.*

Proof Since $\mathbf{Z}$ decomposes G, we have $\mathrm{dsep}_G(\mathbf{X}_1, \mathbf{Z}, \mathbf{X}_2)$ and $\mathrm{dsep}_G(\mathbf{X}_1\mathbf{Y}_1, \mathbf{Z}, \mathbf{X}_2\mathbf{Y}_2)$. We have:

$$\begin{aligned} Pr(\mathbf{y}|\mathbf{x}, \mathbf{z}) &= \frac{Pr(\mathbf{y}, \mathbf{x}|\mathbf{z})}{Pr(\mathbf{x}|\mathbf{z})} = \frac{Pr(\mathbf{y}_1, \mathbf{y}_2, \mathbf{x}_1, \mathbf{x}_2|\mathbf{z})}{Pr(\mathbf{x}_1, \mathbf{x}_2|\mathbf{z})} = \frac{Pr(\mathbf{y}_1, \mathbf{x}_1|\mathbf{z})Pr(\mathbf{y}_1, \mathbf{x}_1|\mathbf{z})}{Pr(\mathbf{x}_1|\mathbf{z})Pr(\mathbf{x}_2|\mathbf{z})} \\ &= \left[\frac{Pr(\mathbf{y}_1, \mathbf{x}_1|\mathbf{z})}{Pr(\mathbf{x}_1|\mathbf{z})}\right]\left[\frac{Pr(\mathbf{y}_2, \mathbf{x}_2|\mathbf{z})}{Pr(\mathbf{x}_2|\mathbf{z})}\right] = Pr(\mathbf{y}_1|\mathbf{x}_1, \mathbf{z})Pr(\mathbf{y}_2|\mathbf{x}_2, \mathbf{z}) \end{aligned} \tag{4}$$

Equation (4) follows from $\mathrm{dsep}_G(\mathbf{X}_1, \mathbf{Z}, \mathbf{X}_2)$ and $\mathrm{dsep}_G(\mathbf{X}_1\mathbf{Y}_1, \mathbf{Z}, \mathbf{X}_2\mathbf{Y}_2)$. This concludes our proof. Although we only consider the case of two subnetworks here, it is easy to see that this lemma generalizes to an arbitrary number of subnetworks decomposed by $\mathbf{Z}$. ■

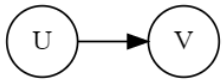
We are now ready to prove Theorem 3. By construction of G', $\mathbf{U} \cup \{H\}$ decomposes G' into its n components $G^1, G^2, \ldots G^n$. We have:

$$\begin{aligned} Pr'(\mathbf{y}, \mathbf{w} \mid \mathbf{x}, \mathbf{v}, \mathbf{e}, \mathbf{u}) &= \sum_{i=1}^{n} Pr'(\mathbf{y}, \mathbf{w}, h_i \mid \mathbf{x}, \mathbf{v}, \mathbf{e}, \mathbf{u}) \\ &= \sum_{i=1}^{n} Pr'(\mathbf{y}, \mathbf{w} \mid \mathbf{x}, \mathbf{v}, \mathbf{e}, \mathbf{u}, h_i) Pr'(h_i \mid \mathbf{x}, \mathbf{v}, \mathbf{e}, \mathbf{u}) \\ &= \sum_{i=1}^{n} Pr'(\mathbf{y}, \mathbf{w} \mid \mathbf{x}, \mathbf{v}, \mathbf{e}, \mathbf{u}, h_i) Pr'(h_i) && (5) \\ &= \sum_{i=1}^{n} \left[\prod_{j=1}^{n} Pr'(\mathbf{y}^j, \mathbf{w}^j \mid \mathbf{x}^j, \mathbf{v}^j, \mathbf{e}^j, \mathbf{u}, h_i)\right] Pr'(h_i) && (6) \\ &= \sum_{i=1}^{n} \left[\prod_{j \neq i} pr'(\mathbf{y}^j, \mathbf{w}^j \mid \mathbf{x}^j, \mathbf{v}^j, \mathbf{e}^j, \mathbf{u}, h_i)\right] Pr'(\mathbf{y}^i, \mathbf{w}^i \mid \mathbf{x}^i, \mathbf{v}^i, \mathbf{e}^i, \mathbf{u}, h_i) Pr'(h_i) && (7) \\ &= \sum_{i=1}^{n} \left[\prod_{j \neq i} 1.0\right] Pr'(\mathbf{y}^i, \mathbf{w}^i \mid \mathbf{x}^i, \mathbf{v}^i, \mathbf{e}^i, \mathbf{u})\, w_i && (8) \\ &= \sum_{i=1}^{n} w_i\, Pr(\mathbf{y}^i_{\mathbf{x}^i}, \mathbf{w}^i_{\mathbf{v}^i} \mid \mathbf{e}^i, \mathbf{u}) \\ &= L(\mathbf{u}) && (9) \end{aligned}$$

Equation (5) follows since the auxiliary root H is d-separated from $\mathbf{X} \cup \mathbf{V} \cup \mathbf{E} \cup \mathbf{U}$. Equation (6) follows from Lemma 18 since $\mathbf{U} \cup \{H\}$ decomposes G such that all triplet models are disconnected. Equation (7) follows from the construction of the new CPTs of $\mathbf{Y}^i$ and $\mathbf{W}^i$: if $H = h_i$, then the original CPTs of $\mathbf{Y}^i$ and $\mathbf{W}^i$ are preserved, and the values of $\mathbf{Y}^j$ and $\mathbf{W}^j$ are fixed to $\mathbf{y}^j$ and $\mathbf{w}$ for all $j \neq i$. Equation (8) follow from the property of the triplet network.

Appendix B. Example for MAP and Reverse-MAP

Cconsider the simple model in Figure 3. We have $\text{argmax}_u \, Pr(u, v_1) = u_2$ for MAP while $\text{argmax}_u \, Pr(v_1|u) = u_1$ for R-MAP.



(c)

U	$\theta(A)$
u_1	0.2
u_2	0.8

(d)

U	V	$\theta(V\|U)$
u_1	v_1	0.6
u_1	v_2	0.4
u_2	v_1	0.3
u_2	v_2	0.7

(e)

Figure 3: An example for illustrating the difference between MAP and R-MAP.

Appendix C. Proof of Corollary 5

We can reduce Reverse-MAP $\text{argmax}_{\mathbf{u}} \, Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2)$ to unit selection by choosing an objective function in the form of Equation (1) with the following settings: $n = 1$, $w_1 = 1$, $\mathbf{X}^1 = \{\}$, $\mathbf{Y}^1 = \mathbf{E}_1$ and $\mathbf{E}^1 = \mathbf{E}_2$. This is clearly a polynomial-time reduction. We already showed a reduction from unit selection to Reverse-MAP in Theorem 3. Let $|G|$ denote the size of SCM G and n be the number of components in the causal objective function. By inspecting Definition 2, we can immediately see that the time for constructing the objective model G' is $O(n \cdot |G|)$. Moreover, the size of objective model G' is also $O(n \cdot |G|)$.

Appendix D. Proof of Theorem 7

Membership in NP^{PP} is immediate. Given an instantiation $\mathbf{u}$ of $\mathbf{U}$, it is easy to verify if $\mathbf{u}$ is a solution by querying the PP-oracle if $Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) > p$ which is a problem known as D-MAR (**?**). To prove hardness, we show that E-MAJSAT (Littman et al., 1998) can be reduced to D-Reverse-MAP in polynomial time, based on a slight modification of the reduction to classical MAP proposed in (Park, 2002; Park and Darwiche, 2004). The E-MAJSAT problem is defined as follows. Given a Boolean formula α over Boolean variables $\mathbf{Z} = \mathbf{U} \cup \mathbf{V}$: Is there an instantiation $\mathbf{u}$ of $\mathbf{U}$ such that the majority of instantiations $\mathbf{v}$ of $\mathbf{V}$ satisfy $\mathbf{uv} \models \alpha$ (formula α holds at $\mathbf{uv}$)? We show that we can answer E-MAJSAT by answering D-Reverse-MAP on an SCM G_α that simulates the formula α and that can be constructed efficiently. The SCM G_α is constructed inductively, as shown in (Park and Darwiche, 2004),[7] and always has a single leaf node, denoted S_α. The construction is based on

7. (Park and Darwiche, 2004) intended to construct a Bayesian network, but their construction is an SCM since all internal nodes in the network have functional CPTs.

three rules: (1) If $\alpha = X$, then G_α has a single binary node X with values $\{0, 1\}$ and a uniform prior so $S_\alpha = X$; (2) If $\alpha = \neg\beta$, then G_α is constructed from G_β by adding a binary node S_α as a child of $S_\beta \in G_\beta$ with structural equation $S_\alpha = 1 - S_\beta$; and (3) If $\alpha = \beta \wedge \gamma$ ($\alpha = \beta \vee \gamma$), then G_α is constructed from G_β and G_γ by adding a binary node S_α as a child of $S_\beta \in G_\beta$ and $S_\gamma \in G_\gamma$ with structural equation $S_\alpha = S_\beta \cdot S_\gamma$ ($S_\alpha = S_\beta + S_\gamma$). We are now ready for the last step of the proof. Given a Boolean formula α over variables $\mathbf{Z} = \mathbf{U} \cup \mathbf{V}$, and given its SCM G_α that has distribution Pr, we next show that there is an instantiation $\mathbf{u}$ such that $Pr(S_\alpha = 1|\mathbf{u}) > 1/2$ (D-Reverse-MAP query) iff there is an instantiation $\mathbf{u}$ such that the majority of instantiations $\mathbf{v}$ of $\mathbf{V}$ satisfy $\mathbf{uv} \models \alpha$ (E-MAJSAT query). Let $|\mathbf{Z}| = n$ and s_α denote $S_\alpha = 1$. By construction of G_α (Park and Darwiche, 2004), we have $Pr(\mathbf{z}) = 1/2^n$ for all instantiations $\mathbf{z}$; $Pr(s_\alpha|\mathbf{z}) = 1$ if $\mathbf{z} \models \alpha$ and $Pr(s_\alpha|\mathbf{z}) = 0$ otherwise. Then $Pr(\mathbf{z}, s_\alpha) = Pr(s_\alpha|\mathbf{z})Pr(\mathbf{z}) = 1/2^n$ if $\mathbf{z} \models \alpha$ and $Pr(\mathbf{z}, s_\alpha) = 0$ otherwise. We finally have:

$$Pr(s_\alpha|\mathbf{u}) = \frac{Pr(\mathbf{u}, s_\alpha)}{Pr(\mathbf{u})} = \frac{\sum_{\mathbf{v}} Pr(\mathbf{u}, \mathbf{v}, s_\alpha)}{\sum_{\mathbf{v}} Pr(\mathbf{u}, \mathbf{v})} = \frac{\sum_{\mathbf{v}:\mathbf{uv}\models\alpha}(1/2^n)}{\sum_{\mathbf{v}}(1/2^n)} = \frac{\mathbf{card}(\{\mathbf{v} \in \mathcal{V} : \mathbf{uv} \models \alpha\})}{\mathbf{card}(\mathcal{V})}$$

Now that we have shown membership and hardness, D-Reverse-MAP is $\mathrm{NP}^{\mathrm{PP}}$-complete.

Appendix E. Proof of Theorem 9

Recall Definition 6 of D-Reverse-MAP: Is there an instantiation $\mathbf{u}$ such that $Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) > p$? Membership in NP is immediate. Since $\mathbf{U}$ is the set of exogenous variables, evidence variables $\mathbf{E}$ are functionally determined by $\mathbf{U}$. Hence, it is easy to check whether an instantiation $\mathbf{u}$ is a solution by first computing the instantiation $\mathbf{e}$ of $\mathbf{E}$ implied by $\mathbf{u}$ (using structural equations) and then checking whether $\mathbf{e}$ is consistent with $\mathbf{e}_1, \mathbf{e}_2$. If the answer is yes, then $Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) = 1$, otherwise $Pr(\mathbf{u}, \mathbf{e}_2) = 0$ or $Pr(\mathbf{e}_1|\mathbf{u}, \mathbf{e}_2) = 0$. To show hardness, we show that SAT can be reduced to D-Reverse-MAP under the conditions stated in the theorem. Given a Boolean formula α over variables $\mathbf{U}$, we construct an SCM G_α as in the proof of Theorem 7. SCM G_α has a single leaf node S_α and its root nodes are $\mathbf{U}$. By construction of G_α, we have $Pr(S_\alpha = 1|\mathbf{u}) = 1$ if $\mathbf{u}$ satisfies α and $Pr(S_\alpha = 1|\mathbf{u}) = 0$ otherwise. By choosing $p = 0$, $\mathbf{e}_1 = \{S_\alpha = 1\}$ and $\mathbf{e}_2 = \emptyset$, the D-Reverse-MAP query (is there $\mathbf{u}$ such that $Pr(S_\alpha = 1|\mathbf{u}) > 0$) answers yes iff there is an instantiation $\mathbf{u}$ that satisfies the formula α (SAT query). This concludes our proof.

Appendix F. Proof of Lemma 12

We first review the standard notions of elimination process, induced clusters and moral graphs; see (Darwiche, 2009, Ch 9) for a detailed treatment. The moral graph G_m of an SCM G is obtained from G by adding an undirected edge between every pair of common parents and then undirecting all edges. Let X_i be the i-th variable in order π. When we eliminate X_i from G, we connect every pair of neighbors of X_i in G_m, if such an edge do not exist, and then remove node X_i from G_m. The cluster $\mathbf{C}_i$ of X_i is defined to be X_i and the neighbors of $\mathbf{X}_i$ in G_m when eliminating X_i. Let G'_m be the moral graph of G' and $\mathbf{C}'_i$ be the cluster induces by eliminating variable X_i from G'. Let n be the number of variables in G. To prove Lemma 12, it suffices to prove the following statement: $\mathbf{C}'_i \subseteq \mathbf{C}_i \cup \{H\}$ for $i = 1, \ldots, n$, which we prove next by induction.

Let $\mathrm{neigh}(X)$ denote the neighbors of X in G_m and let $\mathrm{neigh}'(X)$ denote the neighbors of X in G'_m. Let $\mathbf{Z}$ denote the children of H in G'. For each $Z \in \mathbf{Z}$, let $\mathbf{P}_Z$ denote the parents of Z in

G. First, we show the statement holds when $i = 1$. When creating G'_m from G', the introduction of node H would cause two classes of edges that do not exist in G_m to be added to G'_m: 1) (Z, H) for $Z \in \mathbf{Z}$; 2) (Y, H) if Y is a parent of some node $Z \in \mathbf{Z}$, that is Y and H are common parents of some node Z. This means that before the elimination starts, for any node X, if $X \in \mathbf{Z} \cup_{Z \in \mathbf{Z}} P_Z$, we have $\text{neigh}'(X) = \text{neigh}(X) \cup \{H\}$; otherwise $\text{neigh}'(X) = \text{neigh}(X)$. Hence, $C'_1 \subseteq \{C_1\} \cup \{H\}$. Consider now the elimination of X_{i+1} assume that the statement holds for $1, 2, \ldots, i$. We observe that if $\mathbf{C}'_j \subseteq \mathbf{C}_j \cup \{H\}$, then the elimination of X_j would cause only one type of additional edges be added to G'_m, that is (Y, H) for $Y \in \text{neigh}(X_j)$. This is because the elimination of X_j will form a clique among $\text{neigh}(X_j) \cup \{H\}$ in G'_m, but $\text{neigh}(X_j)$ already forms a clique after X_j is eliminated from G_m. This implies that eliminating X_j ($j \leq i$) will never cause any additional edge to be added among any two nodes that are both not H (in other words, all additional edges added are incident on H). Thus, before we eliminate X_{i+1}, we have $\text{neigh}'(X_{i+1}) \subseteq \text{neigh}(X_{i+1}) \cup \{H\}$ and this implies $\mathbf{C}'_{i+1} \cap \mathbf{C}_{i+1} \cup \{H\}$ which concludes the proof.

Appendix G. Proof of Theorem 13

As mentioned earlier, an objective model as in Definition 2 corresponds to an N-twin network as defined in (Han et al., 2022) except for the addition of mixture node H. An N-twin network is obtained by creating N copies of a directed acyclic graph (DAG) while joining them so a *subset* of their roots are shared. The result in (Han et al., 2022) shows that if the DAG has treewidth w then the N-twin network has treewidth $\leq N(w+1)-1$. Before adding the mixture node H, an objective model with n components corresponds to a $3n$-twin network so its treewidth is $\leq 3n(w+1) - 1$. After adding node H, its treewidth is $\leq 3n(w+1)$ by Lemma 12.

Appendix H. Proof of Theorem 16

This proof uses the standard notions of elimination process, induced clusters and moral graphs which are reviewed in the proof of Lemma 12 in Appendix F.

We assume that different components of L have the same set of outcome variables, that is, $\mathbf{Y}^i\mathbf{W}^i = \mathbf{YW}$. This will not affect the correctness of the proof below. An objective model G' with n components can also be viewed as an n-twin network (Han et al., 2022) where the base network G^b is a triplet network of G with an additional root H. The n copies of G^b share roots $\mathbf{R} = \mathbf{U} \cup \{H\}$. We first show that the $H\mathbf{U}$-constrained order of G' is the same as the $H\mathbf{U}$-constrained order of G^b.

Lemma 19 *Consider a base network $\mathcal{M}$ and an n-twin network $\mathcal{M}'$ of $\mathcal{M}$ that shares roots $\mathbf{R}$. If π is an $\mathbf{R}$-constrained elimination order of $\mathcal{M}$ with width w and π' is a corresponding $\mathbf{R}$-constrained elimination order of $\mathcal{M}'$ with width w', then $w' = w$.*

This lemma can be proved using technique similar to (Han et al., 2022). It implies that the constrained treewidth of G' is independent of the number of components in L but only depend on the constrained treewidth of G^b induced by one component. Next, we show that the constrained treewidth w' of G^b and the constrained treewidth w of G satisfy $w' \leq \max(3w + 3, |\mathbf{U}|)$.

For base variables $\mathbf{X}$, we use $[\mathbf{X}] = \{[X] : X \in X\}$ and $[[\mathbf{X}]] = \{[X] : X \in X\}$ to denote the duplicates of $\mathbf{X}$ in G^b. Let $\mathbf{C}(X)$ denote the cluster corresponding to eliminating variable X from G, and let $\mathbf{C}'(X)$, $\mathbf{C}'([X])$, or $\mathbf{C}^(([X]])$ denote the cluster corresponding to eliminating variable $X, [X]$ or $[[X]]$ from G^b. Given order π/π', we first eliminate non-unit variables, then H, and

finally $\mathbf{U}$ from G/G'. For a non-unit variable X, we can show that the clusters $\mathbf{C}'(X^i)$, $\mathbf{C}'([X^i])$, or $\mathbf{C}^(([X^i]])$ are all subsets of $\mathbf{C}(X) \cup [\mathbf{C}(X)] \cup [[\mathbf{C}(X)]] \cup \{H\}$ using technique similar to (Han et al., 2022). This means that the size of the induced cluster is at most triple the size of the cluster of its base variable in G plus one. When eliminating the remaining variables $\mathbf{U} \cup \{H\}$, the induced cluster is a subset of $\mathbf{U} \cup \{H\}$. To show the second part of the theorem, note that if we restrict the children of mixture variable H in G' to have the same base variable Y, then when eliminating variable H, its neighbors (in $\mathbf{U}$) will already form a clique so eliminating H will not add more fill-in edges among these neighbors in the moral graph.