Causal Structure Learning with Greedy Unconditional Equivalence Search

Alex Markham*,1  Danai Deligeorgaki1  Pratik Misra1  Liam Solus1

1Department of Mathematics, KTH Royal Institute of Technology, Sweden
*Correspondence: alex.markham@causal.dev

Abstract

We consider the problem of characterizing directed acyclic graph (DAG) models up to unconditional equivalence, i.e., when two DAGs have the same set of unconditional d-separation statements. Each unconditional equivalence class (UEC) can be uniquely represented with an undirected graph whose clique structure encodes the members of the class. Via this structure, we provide a transformational characterization of unconditional equivalence. Combining these results, we introduce a hybrid algorithm for learning DAG models from observational data, called Greedy Unconditional Equivalence Search (GUES), which first estimates the UEC of the data using independence tests and then greedily searches the UEC for the optimal DAG. Applying GUES on synthetic data, we show that it achieves comparable accuracy to existing methods. However, in contrast to existing methods, since the average UEC is observed to contain few DAGs, the search space for GUES is drastically reduced.

1 INTRODUCTION

A central aspect of modern causal discovery methods for causally sufficient systems without feedback loops is the subdivision of the space of DAGs into equivalence classes that constitute the distinct possible representatives of the data-generating distribution. When the available data is observational, the typical approach is to subdivide DAG space into Markov equivalence classes (MECs) and then learn the essential graph (CPDAG) representative of the class (Ander-sson and Perlman 1997) using constraint-based, hybrid or (greedy) score-based approaches. To date, some of the best performing causal discovery algorithms are greedy score-based approaches such as the Greedy Equivalence Search (GES) (Chickering 2002). GES moves between candidate Markov equivalence classes based on a transformational relation between I-MAPs known as Meek’s Conjecture (Meek 1997), which was proven to hold by Chickering (2002). This transformational relation between I-MAPs is an extension of a transformational characterization of DAGs within the same MEC, which states that any two DAGs in the same MEC are connected by a sequence of covered arrow reversals (Chickering 1995). In a similar fashion, many of the state-of-the-art causal discovery algorithms have their basis in some characterization of Markov equivalence—another example being the Verma-Pearl characterization of Markov equivalence (Verma and Pearl 1990) and the PC algorithm (Spirtes, Glymour, and Scheines 2000).

One disadvantage of algorithms such as GES is that the search space remains exponentially large, even when restricted to moves between essential graphs (Radhakrishnan, Solus, and Uhler 2017). In an effort to shrink the search space while still taking advantage of the (empirically observed) high-performance of greedy score-based methods, a number of hybrid algorithms have been proposed (Linusson, Restadh, and Solus 2021; Solus, Wang, and Uhler 2021; Tsamardinos, Brown, and Aliferis 2006). Such algorithms use a mixture of constraint-based (e.g., conditional independence) tests and greedy optimization to identify the MEC of the data-generating distribution. However, the CI tests performed for such methods can be numerous, considering a wide variety of conditioning sets, and this can lead to a high rate of false positives. Such errors propagate through greedy optimization phases, leading to significant structural disparity between the learned and true essential graphs.

To mitigate this issue, one option is to rely on only pairwise (unconditional) independence tests when limiting the search space of essential graphs. In this paper, we propose one such hybrid method based on a transformational characterization of unconditional equivalence of DAGs, i.e., where two DAGs are considered equivalent whenever they encode the same set of (unconditional) independence relations \( X_i \perp \perp X_j \). We first show that each unconditional equiva-
lence class (UEC) of DAGs can be uniquely represented by an undirected graph, which serves an analogous purpose to the essential graph of an MEC. It follows from this representation that any two DAGs in the same UEC have the same number of source nodes. We observe empirically (on 5 variables) that most UECs contain DAGs with many source nodes, and such UECs contain few DAGs. This suggests that restricting greedy search methods to a single UEC can drastically reduce the size of the search space.

We also provide a transformational characterization of unconditional equivalence of DAGs that generalizes Chickering’s transformational characterization of Markov equivalence. This characterization yields a set of moves between DAGs within the same UEC. Using these moves, we introduce the Greedy Unconditional Equivalence Search (GUES) for learning an optimal MEC based on observational data. We apply GUES to synthetic data, comparing its performance with GES and popular hybrid algorithms. We observe GUES achieving comparable accuracy with these methods, suggesting that it could be a reasonable alternative to GES.

We note the set of parents, descendants, and ancestors of a node $v$, denoted $\text{pa}(v)$, $\text{de}(v)$, and $\text{ma}(v)$, respectively. For DAGs, $\text{de}(v)$ is the collection of source nodes of $v$. A collection of cliques $\mathcal{E}$ of a graph $G$ is called a minimal edge-clique cover of $G$ if every edge of $G$ is contained in at least one clique in $\mathcal{E}$, and no proper subset of $\mathcal{E}$ satisfies this property (Roberts 1985).

Any proof not immediately following its respective lemma, theorem, or corollary can be found in the Supplementary Material.

### 2.1 UNCONDITIONAL EQUIVALENCE FOR DIRECTED ACYCLIC GRAPHS

Two DAGs $\mathcal{G} = (V, E^\mathcal{G})$ and $\mathcal{H} = (V, E^\mathcal{H})$ are unconditionally equivalent if whenever two nodes $i, j \in V$ are $\text{d}$-separated given $\emptyset$ in $\mathcal{G}$, the nodes $i$ and $j$ are $\text{d}$-separated given $\emptyset$ in $\mathcal{H}$. Markham, Das, and Grosse-Wentrup (2022, Lemma 5) show that unconditional equivalence is indeed an equivalence relation over ancestral graphs and consequently is also an equivalence relation over DAGs. The collection of all DAGs that are unconditionally equivalent to $\mathcal{G}$ is called its unconditional equivalence class (UEC). The following undirected graph serves as a representative of the UEC of $\mathcal{G}$, analogous to the essential graph of an MEC.

**Definition 1.** The unconditional dependence graph of a DAG $\mathcal{G} = (V, E)$ is the undirected graph $\mathcal{U}^\mathcal{G} = (V, \{\{v, w\} : v \text{~d
a
g~} w; v, w \in V\})$.

When the DAG $\mathcal{G}$ is clear from context, we write $\mathcal{U}$ for $\mathcal{U}^\mathcal{G}$. The following theorem provides two characterizations of the undirected dependence graph of a UEC.

**Theorem 2.** The unconditional dependence graph $\mathcal{U}$ of a DAG $\mathcal{G}$ is equivalent to each of the following:

1. $\mathcal{U}_1 = (V, \{\{v, w\} : \text{an}_G(v) \cap \text{an}_G(w) \neq \emptyset\})$, that is, two distinct nodes share an edge in $\mathcal{U}$ if and only if they have a common ancestor in $\mathcal{G}$;
2. $\mathcal{U}_2 = (V, \bigcup_{m \in \text{ma}_G(V)} \{\{v, w\} : v \in \text{de}_G(m) \times \text{de}_G(m) : v \neq w\})$.

**Proof.** Two distinct nodes $v, w$ are adjacent in $\mathcal{U}$ if and only if they are $\text{d}$-connected given $\emptyset$ in $\mathcal{G}$, i.e., there is a trek between them. Such a trek must either be (1) directed from $v$ to $w$, implying $v \in \text{an}_G(w)$, (2) directed from $w$ to $v$, implying $w \in \text{an}_G(v)$, or (3) consisting of a node $c$ with directed paths to both $v$ and $w$, implying $c \in \text{an}_G(v) \cap \text{an}_G(w)$. This happens if and only if $\text{an}_G(v) \cap \text{an}_G(w) \neq \emptyset$, so $\mathcal{U} = \mathcal{U}_1$.

We now show that $\mathcal{U}_2 = \mathcal{U}_1$. Note that if $\{v, w\} \in \mathcal{U}_1$, then $v \neq w$ and $\text{an}_G(v) \cap \text{an}_G(w) \neq \emptyset$. Since $\mathcal{G}$ is a DAG, it follows that there exists $m \in \text{an}_G(v) \cap \text{an}_G(w)$ such that $m \in \text{ma}_G(V)$. Since $v \neq w$, it follows that $\{v, w\} \in \mathcal{U}_2$. On the other hand, if $\{v, w\} \in \mathcal{U}_2$, there exists a node $m$ such that $v, w \in \text{de}_G(m)$. It follows that $m \in \text{an}_G(v) \cap \text{an}_G(w)$ and $\{v, w\} \in \mathcal{U}_1$, which completes the proof. □
Note that Theorem 2 (2) implies that the cliques of a minimal edge-clique cover of \( \mathcal{U} \) are in bijection with the source nodes of \( G \). Namely, given \( v \in \text{ma}(V) \), the maximal clique corresponding to \( v \) is the clique \( C_v := \text{deg}(v) \) and \( \mathcal{E}^{G} := \{ C_v : v \in \text{ma}(V) \} \) is a minimal edge-clique cover of \( \mathcal{U} \). The following lemma will be used.

**Lemma 3.** Suppose \( G \) and \( G' \) are two DAGs in the same UEC with unconditional dependence graph \( \mathcal{U} \). Then:

1. \( |\text{ma}(v)| = |\mathcal{E}^{G}| \),
2. \( \text{ma}(v) \) is a maximum independent set in \( \mathcal{U} \),
3. \( \mathcal{E}^{G} = \mathcal{E}^{G'} \), and
4. \( |\text{ma}(V)| = |\text{ma}(V')| \).

**Proof.** Since each \( C \in \mathcal{E}^{G} \) contains a unique node \( v \in \text{ma}(V) \) in each \( C \in \mathcal{E}^{G} \). Let \( C_v \) be the (unique) clique in \( \mathcal{E}^{G} \) containing \( v \in \text{ma}(V) \), and let \( D_v \) be the (unique) clique in \( \mathcal{E}^{G} \) containing \( v \). It follows that \( C_v \subseteq D_v \). To see this, recall that \( D_v \) contains a unique node \( w \in \text{ma}(V) \) such that \( D_v = \text{deg}(w) \). Since \( v \in D_v \), then \( v \) and \( w \) are adjacent in \( \mathcal{U} \). Since \( v \in \text{ma}(V) \) and \( v \) and \( w \) are adjacent in \( \mathcal{U} \), it follows from Theorem 2 (2) that \( w \in \text{deg}(v) \). Since any \( k \in C_v \) is also a descendant of \( v \) in \( G' \), it follows that there is a trek between \( k \) and \( w \) in \( G' \). Hence, \( k \notin G \) and so \( k \) and \( w \) are adjacent in \( \mathcal{U} \) by Definition 1. Since \( D_v \) is the maximal clique in \( \mathcal{U} \) containing \( w \), it follows that \( C_v \subseteq D_v \). By symmetry of the argument, \( C_v = D_v \). Thus, \( \mathcal{E}^{G} = \mathcal{E}^{G'} \). It follows immediately that \( |\text{ma}(V)| = |\text{ma}(V')| \). \( \square \)

Since \( \mathcal{E}^{G} = \mathcal{E}^{G'} \) for all \( G, G' \) in the UEC represented by \( \mathcal{U} \), we can set \( \mathcal{E}^{G}' := \mathcal{E}^{G} \) for any \( G \) in the class and \( \mathcal{E}^{\mathcal{U}} \) will be well-defined.

**Definition 4.** An ordered pair \( (v, w) \) of two nodes \( v, w \in V \) of a DAG \( G \) is weakly covered in \( G \) if \( \text{ma}(pa(v)) = \text{ma}(pa(v) \setminus \{v\}) \) and \( v \notin \text{an}(pa(w) \setminus \{v\}) \). The ordered pair is partially weakly covered if \( \text{ma}(pa(v)) \subseteq \text{ma}(pa(w) \setminus \{v\}) \) and \( v \notin \text{an}(pa(w) \setminus \{v\}) \), and \( pa(v) \neq \emptyset \).

Observe that a weakly covered pair \( (v, w) \) is also partially weakly covered when \( pa(v) \neq \emptyset \). Weakly covered pairs generalize the covered edges, \( v \to w \) with \( pa(v) = \text{pa}(w) \setminus \{v\} \), used in the transformational characterization of Markov equivalence (Chickering 1995). Notice also that the condition \( v \notin \text{an}(pa(w) \setminus \{v\}) \) in the definition of partially weakly covered excludes the possibility of a partially weakly covered edge also being implied by transitivity. These two conditions combine to characterize those edges that can be added to a DAG to produce another DAG in the same UEC.

**Lemma 5.** Let \( G \) be a DAG with nonadjacent nodes \( v \) and \( w \), and let \( G' \) be the identical digraph but with the edge \( v \to w \) added. Then \( G' \) is a DAG unconditionally equivalent to \( G \) if and only if \( v \to w \) is partially weakly covered or implied by transitivity in \( G \).

**Proof.** Assume \( v \to w \) is partially weakly covered in \( G \). Since \( \text{ma}(pa_G(v)) \subseteq \text{ma}(pa_G(w) \setminus \{v\}) \) and \( w \notin \text{an}(pa_G(w) \setminus \{v\}) \), it follows that \( v \notin \text{an}(pa_G(w) \setminus \{v\}) \), i.e., \( w \) is not an ancestor of \( v \) in \( G \), and hence not in \( G' \). Thus \( G' \) does not contain any cycles. Since \( \text{ma}(pa_G(v)) \subseteq \text{ma}(pa_G(w) \setminus \{v\}) \) and \( pa_G(v) \neq \emptyset \), it also follows that for every parent of \( v \) there is a trek to a parent of \( w \) in \( G \). This means there can be no \( x \) whose only trek to \( w \) is blocked by \( v \). Thus, adding edge \( v \to w \) neither creates nor removes existing directed-connecting paths given \( \emptyset \), so by Definition 1, \( \mathcal{E}^{G} = \mathcal{E}^{G'} \).

Alternatively, assume that \( v \to w \) is implied by transitivity in \( G \), i.e., \( v \notin \text{pa}(w) \) but \( v \in \text{an}(w) \). First, observe that because \( G \) is a DAG, any cycle in \( G' \) must contain \( v \to w \). However, \( v \in \text{an}(w) \), and hence there can be no cycle in \( G' \) unless there is a cycle in \( G \). Observe that \( G' \) has the same set of ancestral relations as \( G \) (i.e., adding a transitive implied edge does not change the ancestor set of any node), so by Theorem 2 (1), \( \mathcal{E}^{G} = \mathcal{E}^{G'} \).

Assume \( v \to w \) is neither partially weakly covered, nor transitively implied in \( G \). There are two cases: either \( v \perp_G w \) or \( v \not\perp_G w \). In the first case, \( v \to w \in \mathcal{E}^{G'} \) implies \( \{v, w\} \in \mathcal{E}^{G} \) and thus \( \mathcal{U}^{G} \neq \mathcal{U}^{G'} \), by Definition 1. In the second case, there must exist a trek between \( v \) and \( w \) in \( G \). Hence, either \( v \in \text{an}_G(w) \), or \( v \notin \text{an}_G(w) \), or there exists \( m \in \text{an}_G(w) \cap \text{an}_G(w) \) where \( m \notin \{v, w\} \). In the first case, \( G' \) would contain a cycle, which is a contradiction. In the second case, \( v \to w \) would be implied by transitivity in \( G \), which is also a contradiction. In the third case, notice that \( \text{pa}_G(v) \neq \emptyset \). Hence, as \( v \to w \) is neither partially weakly covered nor implied by transitivity in \( G \), there exists \( m' \in \text{ma}(pa_G(w)) \) such that \( m' \notin \text{ma}(pa_G(w) \setminus \{v\}) \). This means that \( m' \notin \text{an}_G(w) \). Hence \( \text{an}_G(m') \cap \text{an}_G(w) = \emptyset \), since \( m' \) is a maximal ancestor in \( G \). By Theorem 2 (1), \( m' \perp_G w \). On the other hand, the addition of the edge \( v \to w \) implies that \( m' \in \text{an}_G(w) \), hence \( m' \not\perp_G w \). Therefore, \( G \) and \( G' \) are not unconditionally equivalent. \( \square \)

Analogous to covered edges for MECs, weakly covered edges are precisely the edges that can be reversed to produce another DAG in the same UEC.
Lemma 6. Let $G$ be a DAG containing the edge $v \to w$, and let $G'$ be the digraph identical to $G$ but with the edge reversed, so $w \to v \in E^{G'}$. Then $G'$ is a DAG that is unconditionally equivalent to $G$ if and only if $(v, w)$ is weakly covered in $G$.

Proof. This proof is similar to that of Lemma 5 as well as the (non-weakly) covered edge case for Markov equivalence (Chickering 1995, Lemma 1).

Assume $v \to w$ is weakly covered in $G$. First, observe that $G'$ is a DAG: in order for the reversed edge $w \to v$ to form a cycle in $G'$, it would require $v \in \text{an}_G(w) \setminus \{v\}$, which by definition is not allowed for a weakly covered edge. The definition of weakly covered also implies that for every ancestor of $v$ there is a path to $w$ in $G'$; similarly, every ancestor of $w$ is $d$-connected to $v$ given $\emptyset$ in $G$. Thus, reversing $e$, we create a new unconditional independence in $G$.

Now assume $v \to w$ is not weakly covered in $G$. We show that either $G'$ contains a cycle or $G'$ is a DAG that is not unconditionally equivalent to $G$. There are three cases: (i) $v \in \text{an}_G(w) \setminus \{v\}$, (ii) there is some $c \in \text{ma}_G(v)$ but $c \notin \text{ma}_G(w) \setminus \{v\}$ such that $v$ lies in all paths from $c$ to $w$, or (iii) there is some $c \in \text{ma}_G(w) \setminus \{v\}$ but $c \notin \text{ma}_G(v)$, in which case all paths from $c$ to $v$ in $G$ are blocked by $w$ or other descendants of $v$. In the first case, reversing the edge $v \to w$ creates a cycle in $G'$. In the second case, reversing the edge blocks all paths from $c$ to $w$, creating a new unconditional independence in $G'$. In the third case, reversing the edge $v \to w$ produces a $d$-connecting path from $c$ to $w$ given $\emptyset$, removing an unconditional independence from $G$. Thus, in any case, $U^G \neq U^{G'}$, completing the proof.

Via Lemmas 5 and 6 we can show that any two DAGs $G$ and $G'$ in the same UEC are connected by a sequence of edge additions, reversals, and removals, such that after each transformation, the resulting DAG is also in the UEC. To do so, we first show that one can produce a DAG $H^{G,G'}$ in the UEC for which two nodes are adjacent in $H^{G,G'}$ if and only if they are adjacent in either $G$ or $G'$.

Let $H^{G,G'}$ denote the output of Algorithm 1 given $G$ and $G'$. Note that $H^{G,G'}$ is a DAG.

Lemma 7. Given two unconditionally equivalent DAGs $G$ and $G'$, every edge $v \to w$ in $H^{G,G'}$ is not in $G$ is either implied by transitivity or partially weakly covered in $G$.

Proof. Let $\mathcal{U}$ denote the graph $\mathcal{U}$ with all edges $\{v, w\}$ removed for which there exists a clique in $\mathcal{E}^{\mathcal{U}}$ containing both $v$ and $w$, a clique in $\mathcal{E}^{\mathcal{U}}$ containing $v$ but not $w$, and a clique in $\mathcal{E}^{\mathcal{U}}$ containing $w$ but not $v$. Suppose now that $v \to w$, $u \to t \notin E^G$ with $(v, w), (u, t)$ each partially weakly covered or implied by transitivity in $G = (V, E)$. Then $(u, t)$ is also partially weakly covered or implied by transitivity in $G' = (V, E \cup \{v \to w\})$. To see this, assume first that $(u, t)$ is implied by transitivity in $G$ i.e., $u \in \text{an}_G(t)$. Since adding an edge to $G$ implies that $\text{an}_G(S) \subseteq \text{an}_G(S)$ for all $S \subseteq V$, $(u, t)$ also satisfies $u \in \text{an}_G(t)$.

If $(v, w)$ is partially weakly covered in $G$ or implied by transitivity in $G$ then $\mathcal{ma}_G(w) = \mathcal{ma}_G(w)$, and hence $\mathcal{ma}_G(S) = \mathcal{ma}_G(S)$ for any set $S \subseteq V$. Hence, $\mathcal{ma}_G(u) = \mathcal{ma}_G(u)$ and $\mathcal{ma}_G(t) \setminus \{u\} = \mathcal{ma}_G(t) \setminus \{u\}$. If $(u, t)$ is partially weakly covered in $G$ then it follows by $\emptyset \neq \mathcal{ma}_G(u)$ and $\mathcal{ma}_G(t) \setminus \{u\}$ that $(u, t)$ is partially weakly covered in $G'$ in case $w \notin \text{an}_G(t)$, and implied by transitivity in case $u \in \text{an}_G(t)$.

This observation shows that we can add edges that are implied by transitivity or partially weakly covered in $G$ to $G$ in any order, and after each addition, by Lemma 5, the resulting graph is a DAG that is unconditionally equivalent to $G$. Let $\mathcal{U}$ denote the DAG resulting from all such edge additions. We may assume that the topological ordering $\succeq^G$ agrees with the topological ordering of $\mathcal{U}$.

Now we show that the set of edges in $\mathcal{U}$ that are not in the skeleton of $\mathcal{U}$ is equal to the set of edges in $\mathcal{U}$ that are not in $\mathcal{U}$, which implies that the skeleton of $\mathcal{U}$ is equal to $\mathcal{U}$. Let $(u, v)$ be an edge of $\mathcal{U}$. By Theorem 2 (1) there exists $m \in \mathcal{ma}_G(V)$ such that $m \in \text{an}_G(u) \cap \text{an}_G(v)$ and $\mathcal{pa}_G(u) \neq \emptyset$ and $\mathcal{pa}_G(v) \neq \emptyset$. Hence, $u$ and $v$ both belong to the maximal clique of $E^{\mathcal{U}}$ that contains $m$.

If $(u, v) \in \mathcal{U}$ is in $\mathcal{G}$ then $u$ and $v$ are adjacent in $\mathcal{U}$, by definition. Moreover, either $\mathcal{ma}_G(u) \subseteq \mathcal{ma}_G(v)$ or $\mathcal{ma}_G(v) \subseteq \mathcal{ma}_G(u)$, hence Theorem 2 (2) implies that the edge $(u, v)$ is not removed from $\mathcal{U}$ when constructing $\mathcal{U}$.

Now let $(u, v)$ and $(v, u)$ not in $E^G$. Then, $u$ and $v$ are not
We say that a DAG $G$ weakly covered in $G$ if and only if $(u, v)$ and $(v, u)$ are not implied by transitivity in $G$ or partially weakly covered in $G$. By definition, since $u$ and $v$ have at least one parent in $G$, $(u, v)$ or $(v, u)$ are neither implied by transitivity in $G$ nor partially weakly covered in $G$ if and only if if $u \notin \text{an}(pa_G(v) \setminus \{v\})$, $v \notin \text{an}(pa_G(u) \setminus \{v\})$ and $\text{ma}(pa_G(v) \setminus \{u\}), \text{ma}(pa_G(v) \setminus \{v\}) \notin \text{ma}(pa_G(u) \setminus \{v\})$, i.e., there exist $m' \in \text{ma}(pa_G(u)), m'' \in \text{ma}(pa_G(v))$ such that $m' \notin \text{ma}(pa_G(v) \setminus \{v\}), m'' \notin \text{ma}(pa_G(u) \setminus \{v\})$. Hence, it follows from Theorem 2 (2) that $u$ and $v$ are not adjacent in $\hat{U}$ if and only if if $u \notin \text{an}(pa_G(v) \setminus \{u\}), v \notin \text{an}(pa_G(u) \setminus \{v\})$ and $u$ is contained in the maximal clique of $E^{\hat{U}}$ containing $m'$ but $v$ is not, and similarly $v$ is contained the maximal clique of $E^{\hat{U}}$ containing $m''$ but $u$ is not. Since $\{u, v\} \in E^{\hat{U}}$ implies that both $u$ and $v$ belong to the maximal clique of $E^{\hat{U}}$ containing $m$, we have that $u$ and $v$ are not adjacent in $\hat{U}$ if and only if if $u$ and $v$ are not adjacent in $U$.

It follows that for any graph $G$ in the UEC represented by $U$, the edges in $U$ that are not in the skeleton of $G$ are implied by transitivity in $G$ or are partially weakly covered in $G$. Let $G'$ be another member of the UEC of $U$. Suppose that $u, v \notin \text{adjacent in } G$. Then $\{u, v\} \in E_u$ is in the skeleton of $\hat{U}$. Hence, if $u \not\sim^{G} v$ then $(u, v)$ is either implied by transitivity in $G$ or is partially weakly covered in $G$. Since adding all such edges in any order produces the graph $H^{G, G'}$, which completes the proof. 

We say that a DAG $G$ is maximal in its UEC if any DAG produced by adding an edge to $G$ is not in the same UEC as $G$. Note that the DAG $\hat{U}$ constructed in the proof of Lemma 7 is maximal for any choice of DAG $G$.

**Definition 8.** Let $\Delta(G, G') := \{v \rightarrow w \in E^G : w \rightarrow v \in E^{G'}\}$ denote the set of edges in $G$ that have opposite orientation in $G'$. Let $\Gamma(G, G') := \{v \rightarrow w \in E^G : v \rightarrow w \notin \Delta(G', G) \cup \overline{E^G}\}$ denote the set of edges between nodes adjacent in $G'$ but not $G$.

Algorithm 2 provides a generalization of the find edge algorithm of (Chickering 1995) for the identification of covered edges to an algorithm that identifies weakly covered edges.

**Lemma 9.** Let $G$, $G'$ be unconditionally equivalent DAGs with the same skeleton but at least one differently oriented edge. Then the edge $v \rightarrow w$ returned by Algorithm 2 is weakly covered in $G$.

**Proof.** There are two cases: either (i) $\text{ma}(pa_G(v)) = \emptyset$, or (ii) there is some $v' \in \text{ma}(pa_G(v))$.

In case (i), it must also be that $\text{ma}(pa_G(w) \setminus \{v\}) = \emptyset$, otherwise $v$ would be $d$-separated from all other ancestors of $w$ given $\emptyset$ in $G$, contradicting the fact that $G'$ has edge $w \rightarrow v$, producing a $d$-connecting path between $v$ and the ancestors of $w$ given $\emptyset$ in $G'$, and is unconditionally equivalent to $G$ with the same skeleton. Hence, $\text{ma}(pa_G(v)) = \text{ma}(pa_G(w) \setminus \{v\}) = \emptyset$ and $v \notin \text{an}(pa_G(w) \setminus \{v\})$, so $(v, w)$ is weakly covered in $G$.

In case (ii), for every $v' \in \text{ma}(pa_G(v))$, there must also be a directed path in $G$ from $v'$ to $w$ that does not contain $v$; otherwise, since $w$ is minimal with respect to $\prec^{G}$, the edge $w \rightarrow v$ in $G'$ would make $v$ a collider, and $v'$ would be $d$-separated from $w$ given $\emptyset$ in $G'$. Hence, $\text{ma}(pa_G(v)) \subseteq \text{ma}(pa_G(w) \setminus \{v\})$. Furthermore, using the same argument as in case (i), there can be no additional maximal ancestors of $w$, so $\text{ma}(pa_G(v)) \supseteq \text{ma}(pa_G(w) \setminus \{v\})$. Finally, because $v$ is maximal with respect to $\prec^{G}$, and because $G'$ is acyclic and unconditionally equivalent to $G$ and with the same skeleton, $v \notin \text{an}(pa_G(w) \setminus \{v\})$. Hence $(v, w)$ is weakly covered in $G$, completing the proof.

The following theorem generalizes the transformational characterization of Markov equivalence given in (Chickering 1995) to a transformational characterization of unconditional equivalence.

**Theorem 10** (Transformational Characterization). Let $G$ and $G'$ be two unconditionally equivalent graphs. There exists a sequence of $|E^{G'} \setminus E^G|$ edge insertions, followed by $|\Delta(H^{G, G'}, H^{G', G})|$ edge reversals, followed by $|E^G \setminus E^{G'}|$ edge deletions that transforms $G$ into $G'$ with the following properties:

1. Each edge inserted or deleted in $G$ is partially weakly covered or implied by transitivity.
2. Each edge reversed in $G$ is weakly covered.
3. After each operation, $G$ is a DAG and $\hat{U}^G = \hat{U}^{G'}$.
4. After all operations, $G = G'$.

**Proof.** Using Algorithm 1, $H^{G, G'}$ is produced from $G$ by adding exactly the edges in $\Gamma(G, H^{G, G'})$, of which there are $|E^{G'} \setminus E^G|$ in total. By Lemma 7, each edge added in this phase is either partially weakly covered or implied

---

**Algorithm 2: find_edge(G, G'); adapted from (Chickering 1995)**

**input**: unconditionally equivalent DAGs $G, G'$ with the same skeletons but at least one differently oriented edge

**output**: edge from $\Delta(G, G')$

1. let $w$ be the minimal node with respect to $\prec^{G}$ for which there is $w'$ such that $w' \rightarrow w \in \Delta(G, G')$
2. let $v$ be the maximal node with respect to $\prec^{G}$ for which $v \rightarrow w \in \Delta(G, G')$
3. return $v \rightarrow w$
by transitivity in $G$. Similarly, $H^{G',G}$ can be produced via Algorithm 1 by adding a sequence of $|E^G \setminus E^{G'}|$ edges to $G'$ that are all either partially weakly covered or implied by transitivity in $G'$. Moreover, as seen in the proof of Lemma 7, if $H$ is produced from $H'$ by adding $v \rightarrow w$ to $H'$ where $(v, w)$ is implied by transitivity in $H'$ or is partially weakly covered in $H'$ then all other such pairs $(v, w)$ are still implied by transitivity or partially weakly covered in $H$. Thus, reversing the sequence of edge additions used to produce $H^{G',G}$ from $G'$, the edge removed from $H'$ to produce $H$ is always partially weakly covered or implied by transitivity in $H$. It further follows from Algorithm 1 that after each edge addition or removal, the resulting graph is a DAG.

Finally, consider $H^{G',G'}$ and $H^{G',G}$. By construction, their skeletons satisfy $H^{G',G'} = H^{G',G}$. Starting with $H^{G',G}'$, by Lemma 9, Algorithm 2 identifies an edge in $\Delta(H^{G',G'}, H^{G',G})$ that is weakly covered, say $u \rightarrow v$. By Lemma 9, reversing this edge in $H^{G',G}'$ produces a DAG $H$ for which $\Delta(H, H^{G',G}) = \Delta(H^{G',G}, H^{G',G'}) \setminus \{u \rightarrow v\}$. Thus, after each edge reversal, the resulting graph is a DAG, and the cardinality of $\Delta(H, H^{G',G})$ is reduced by one. This completes the proof.

2.2 THE GUES ALGORITHM

Since the unconditional dependence graph $U$ of a DAG $G$ is a perfect I-MAP of the d-separations $X_A \perp_{G} X_B$, it follows that $U$ is an I-MAP of any distribution faithful to $G$. Hence, we can devise a hybrid causal discovery algorithm that uses the transformations in Theorem 10 in its greedy optimization phase. Namely, we define the Greedy Unconditional Equivalence Search (GUES) in Algorithm 3. In the following, for a DAG $G$ containing the edge $v \rightarrow w$, we let $G_{v \rightarrow w}$ denote the digraph $G$ with $v \rightarrow w$ reversed, for a DAG $G$ containing $v \rightarrow w$, we let $G_{v \rightarrow w}$ denote the DAG $G$ with the arrow $v \rightarrow w$ removed, and for $G$ not containing $v \rightarrow w$, we let $G_{v \rightarrow w}$ denote the digraph $G$ with the arrow added. We call an edge reversible if it is weakly covered but not covered. We call it removable (addable) if it is either partially weakly covered or implied by transitivity and in the DAG (not in the DAG). We also use $v \circ \rightarrow w$ to represent that the endpoint of the edge pointing towards $v$ could either be directed or undirected.

In Algorithm 3, beginning with a complete undirected graph on nodes $V$, we first perform pairwise independence tests removing the edge $\{v, w\}$ whenever $X_v \perp_{X_w}$. If the data-generating distribution is faithful to a DAG $G$, the resulting graph will be the unconditional dependence graph $U$ of the UEC containing $G$. Next, we apply Algorithm 4 to $U$ to produce a maximal DAG in the UEC of $U$. Since the maximal DAGs within a UEC form a Markov equivalence class (Corollary 14), and hence will all score the same for any score equivalent and decomposable score, we first compute this MEC and then enter a phase of edge deletions. At each step in this phase, we identify the optimal edge removal out of all removable edges and move to this graph. We continue in this fashion until no such removable edge exists. We then pass to a phase of edge additions, where we pick the optimal scoring addition of an edge that is either partially weakly covered or implied by transitivity. We then loop through these two phases until no edge removal or addition can be performed to further increase the score. GUES then returns the optimal scoring DAG over all runs conducted for each element of the MEC of maximal DAGs. By Theorem 10, these moves are capable of traversing the entire UEC in search of the optimal DAG.

| Algorithm 3: Greedy Unconditional Equivalence Search (GUES) |
|-------------------------------------------------------------|
| **input**: data set $S$                                      |
| **output**: optimal DAG $G^*$                                |
| 1 perform statistical hypothesis tests to get $U$           |
| 2 initialize DAG $H := initialize(U)$                       |
| 3 for $H' \in \text{MEC}(H)$ do                            |
| 4 $G_{H'} := H'$                                            |
| 5 $R := \{v \rightarrow w \in E^G : v \rightarrow w \text{ removable}\}$ |
| 6 $G := \arg \max_{G \in \{v \rightarrow w \in R\}} \text{score}(G)$ |
| 7 while $\text{score}(G) > \text{score}(G_{H'})$ do          |
| 8 $G_{H'} := G$                                              |
| 9 repeat lines 5–6                                          |
| 10 end                                                      |
| 11 while $\text{score}(G) > \text{score}(G_{H'})$ do         |
| 12 $G_{H'} := G$                                             |
| 13 $A := \{v \rightarrow w \in E^{G_{H'}} : v \rightarrow w \text{ addable}\}$ |
| 14 $G := \arg \max_{G \in \{v \rightarrow w \in A\}} \text{score}(G)$ |
| 15 while $\text{score}(G) > \text{score}(G_{H'})$ do          |
| 16 $G_{H'} := G$                                              |
| 17 repeat lines 13–14                                       |
| 18 end                                                      |
| 19 repeat lines 5–10                                         |
| 20 end                                                      |
| 21 $G^* = \arg \max_{G \in \{G_{H'} : H' \in \text{MEC}(H)\}} \text{score}(G)$ |
| 22 return $G^*$                                              |

In the following lemma we observe that the output of step 3 of Algorithm 4 has the same skeleton as the undirected graph $U$ constructed in the proof of Lemma 7.

**Lemma 11.** Suppose that $U = (V, E)$ is the unconditional representative of a non-empty UEC of DAGs. Then the output of step 3 of initialize($U$) is a PDAG with skeleton $U$.

Lemma 11 shows that the skeleton of the output of Algorithm 4 is $U$, and hence it is equal to the skeleton of any maximal DAG $H$ contained in the UEC of $U$. The next
Algorithm 4: initialize(U)

1. **input**: undirected graph U
2. **output**: maximal DAG \( \hat{G} \) in the UEC of U
3. Initialize all paths oriented as v-structures.
4. Initialize all bidirected edges removed.
5. Initialize a total ordering of \( V^\hat{G} \) with all center nodes of v-structures in \( \hat{G} \) last.
6. Return \( \hat{G} \).

lemma is the first step towards proving that the v-structures of the output \( \hat{G} \) of Algorithm 4 and \( \hat{H} \) are also equal.

Lemma 12. Suppose that \( U = (V, E) \) is the unconditional representative of a non-empty UEC of DAGs. Then the output of initialize(U) has the same v-structures as the output of step 3 of initialize(U).

We now see that the clique structure of \( U \) characterizes the v-structures of all maximal DAGs in the UEC of \( U \).

Lemma 13. Suppose that \( U = (V, E) \) is the unconditional representative of a non-empty UEC of DAGs containing a DAG \( \hat{H} \). If \( \hat{H} \) is maximal within the UEC of \( U \) then a path \( \langle v, x, w \rangle \) is a v-structure in \( \hat{H} \) if and only if there exist cliques \( C_1, C_2 \in E^{U} \) such that \( v, x \in C_1 \) but \( w \notin C_1 \), and \( w, x \in C_2 \) but \( v \notin C_2 \), and any clique in \( E^{U} \) containing \( v \) or \( w \) also contains \( x \).

Corollary 14. Suppose that \( U \) is the unconditional representative of a non-empty UEC of DAGs. Then the maximal DAGs in the UEC form a Markov equivalence class.

The following theorem shows that Algorithm 4 is valid.

Theorem 15. Let \( U \) be an undirected graph representing a non-empty UEC of DAGs. Then the output of initialize(U) is a maximal DAG in the UEC represented by \( U \).

Proof. Since the UEC represented by \( U \) is nonempty, then, as seen in the proof of Lemma 7, the UEC contains a maximal DAG \( \hat{H} \) with skeleton \( \hat{U} \). It follows from Lemma 11 that the output \( \hat{G} \) of initialize(U) has skeleton \( \hat{U} \), the same as that of \( \hat{H} \). Hence, to prove the desired result, by Corollary 14 it suffices to show that \( \hat{H} \) and \( \hat{G} \) have the same v-structures. By Lemmas 12 and 13, it suffices to show that the v-structures of the output of step 3 of initialize(U) are exactly the paths \( \langle v, x, w \rangle \) of \( U \) for which there exist cliques \( C_1, C_2 \in E^{U} \) such that \( v, x \in C_1 \) but \( w \notin C_1 \), \( w, x \in C_2 \) but \( v \notin C_2 \), and any clique containing \( v \) or \( w \) also contains \( x \).

Suppose that \( v \rightarrow x \leftarrow w \) is a v-structure in the output of step 3 of initialize(U). Then \( \{v, x\}, \{w, x\} \in E^{U} \).

There are two cases: either \( v \) and \( w \) are not adjacent in the dependence graph \( U \), or \( v \) and \( w \) are adjacent in \( U \).

In case (i), since \( \{v, w\} \notin E^{U} \), the clique of \( E^{U} \) containing \( v \) and \( x \) does not contain \( w \), and similarly the clique of \( E^{U} \) containing \( w \) and \( x \) does not contain \( v \). In case (ii), it must be that \( \{v, w\} \) was bidirected in step 2 of initialize(U).

Hence, there exists \( s \) adjacent to \( v \) but not \( w \) and \( t \) adjacent to \( w \) but not \( v \) in \( U \), yielding the induced 2-paths \( s, v, w \) and \( v, w, t \) in \( U \).

The edge \( \{s, x\} \) must be in \( U \); otherwise the path \( \langle s, v, x \rangle \) would be induced in \( U \), yielding the v-structure \( s \rightarrow v \leftarrow x \) in step 2 of initialize(U). The bidirected edge \( v \leftrightarrow w \) would have been eliminated in step 3 of initialize(U), which contradicts the existence of the v-structure \( v \rightarrow x \leftarrow w \). By symmetry of the argument, we must also have that \( \{t, x\} \) is an edge of \( U \) in step 3.

We have shown that \( s, v, x \) lie in a clique of \( E^{U} \) that does not contain \( w \), and \( t, w, x \) lie in a clique of \( E^{U} \) that does not contain \( v \).

It remains to show that both in cases i) and ii), \( x \) is contained in every clique of \( E^{U} \) that contains \( v \) or \( w \). By symmetry, it suffices to show that there is no clique in \( E^{U} \) that contains \( x \) but not \( v \). Indeed, if such a clique existed, it would contain some \( r \) adjacent to \( v \) but not \( x \). That would imply that the path \( \langle r, v, x \rangle \) is induced in \( U \) and the output of step 2 of Algorithm initialize(U) contained a \( v \)-structure \( s \rightarrow v \leftarrow x \). Then the edge \( v \leftrightarrow x \) would have been bidirected in step 2 of initialize(U) and removed in step 3; a contradiction.

For the opposite direction, assume that there exist cliques \( C_1, C_2 \in E^{U} \) such that \( v, x \in C_1 \) but \( w \notin C_1 \), and \( w, x \in C_2 \) but \( v \notin C_2 \), and any clique containing \( v \) or \( w \) also contains \( x \). Again, there are two cases: either i) \( \{v, w\} \notin E^{U} \), or ii) \( \{v, w\} \subseteq E^{U} \). In case i), the path \( \langle v, x, w \rangle \) is induced in \( U \), hence there is a \( v \)-structure \( v \rightarrow x \leftarrow w \) introduced in step 2 of Algorithm initialize(U). In case ii), let \( s \in \text{max}_{\mathcal{H}}(V) \) be the source node of \( \hat{H} \) that lies in \( C_1 \). Then \( w \) is not adjacent to \( s \) in \( U \); otherwise Theorem 2 (2) would imply that \( w \) lies in \( C_1 \). Therefore, \( s \notin \{v, x\} \) and the paths \( s, v, w \) and \( s, x, w \) are induced in \( U \). Similarly, there exists some \( t \in \text{max}_{\mathcal{H}}(V) \cap C_2 \) such that the paths \( \langle t, v, x \rangle \) and \( \langle t, x, w \rangle \) are induced in \( U \). This implies that the bidirected edge \( v \leftrightarrow w \) and the edges \( v \leftarrow x, w \leftarrow x \) are introduced in step 2 of Algorithm initialize(U).

Both in cases i) and ii), there is no edge between \( v \) and \( w \) in the output of step 3 of initialize(U). Furthermore, there is a path \( v \leftarrow x \rightarrow w \) in the output of step 2 of initialize(U). Since the output of step 3 of initialize(U) has the same skeleton as \( \hat{U} \) (Lemma 11), and we assumed that \( v, x \in C_1 \) and \( w, x \in C_2 \), so \( \{v, x\}, \{w, x\} \in E^{U} \), there must be edges between \( v \) and \( x \) and between \( w \) and \( x \) in the output of step 3 of initialize(U). Thus, the edges \( v \leftarrow x, w \leftarrow x \) cannot be bidirected in step 3, and \( v \rightarrow x \leftarrow w \) is a v-structure in the output of step 3 of initialize(U).
We conclude that $\mathcal{H}$ and $\mathcal{G}$ have the same v-structures and skeleton, and hence are Markov equivalent. By Corollary 14, $\mathcal{G}$ is a maximal DAG in the UEC represented by $\mathcal{U}$.

Since the output of Algorithm 4 is maximal in the UEC whenever $\mathcal{U}$ encodes the unconditional independence relations of a distribution faithful to a DAG, and since we compute all members in the MEC consisting of maximal DAGs, we only use edge removals and additions in Algorithm 3, as these moves are then sufficient to traverse the entire UEC.

Since GUES is a hybrid algorithm that first performs a series of constraint-based tests to limit the size of the search space (similar to MMHC (Tsamardinos, Brown, and Aliferis 2006)), one hope is that for certain families of DAGs this reduction is significant; for instance, in the case that the MEC of maximal DAGs within the UEC is small. Alternatively, Lemma 3 shows that the DAGs in a given UEC have source nodes in bijection with the cliques of $E^U$. This suggests that causal systems with many source nodes (i.e., root causes) should have relatively small UEC size, leading to a significant reduction in the search space of GUES. To explore this hypothesis, we divided all DAGs on 5 nodes into their corresponding UECs and counted the size and number of sources in the DAGs in each class. Figure 1 displays the results. From this figure we see that having many source nodes (relative to the number of variables) appears to result in a small UEC class size. Moreover, we see that most UECs have small size and many source nodes.

### 3 COMPARISON ON SYNTHETIC DATA

For the simulation study we generated 100 random linear Gaussian DAG models on 6 nodes with mutually independent standard normal errors for each edge probability $p \in \{0.1, \ldots, 0.9\}$. Edge weights were assigned uniformly at random from $(0, 2)$. We compared the performance of GUES with GES, MMHC (Max-Min Hill-Climbing) (Tsamardinos, Brown, and Aliferis 2006), and GSP (GreedySP) (Solus, Wang, and Uhler 2021) with depth and run parameters 1. Both MMHC and GSP are hybrid algorithms that rely on CI tests. We used partial correlation tests with significance level $\alpha = 0.05$ for CI tests for all three hybrid algorithms. For GSP, increasing the depth and run parameters is known to vastly improve performance but can reduce scalability in terms of time complexity (Solus, Wang, and Uhler 2021). Here, we set both parameters equal to 1 as this choice will certainly scale to models with more variables. We used the implementation of MMHC available in the `bnlearn` package in R (Scutari 2010), and the `python` implementation of GSP in the `causaldag` package (Squires 2018). The simulation study and implementation of GUES used here is available at [https://medil.causal.dev/gues](https://medil.causal.dev/gues).

All four algorithms were tasked with learning a DAG representative of each of the 900 models based on 10,000 iid samples. The adjacency matrices of the learned CPDAGs were compared with the adjacency matrices of the corresponding true CPDAGs. The accuracy of the algorithm was measured as the proportion of entries of these two matrices that agreed. The average accuracy of each algorithm for the 100 models for each edge probability $p$ is reported in Figure 2, and the proportion of models learned exactly by each algorithm is reported in Figure 3. We see from Figure 2 that GUES and GSP both have comparable performance to GES, with both doing marginally worse than GES, and GUES performing slightly better than GSP for most values of $p$. The difference in performance of GUES and GSP relative to GES can partially be attributed to errors in the CI testing.

Figure 3 shows that GES learned the most models exactly, followed by GUES, suggesting GUES as a reasonable alter-
Figure 3: Proportion of DAGs correctly learned by GUES compared to that of other causal structure learning methods as edge sparseness of generating DAGs varies.

4 DISCUSSION

The transformational characterization of UECs given in Theorem 10 generalizes that of MECs of DAGs (Chickering 1995) while also yielding a collection of moves analogous to those described in Meek’s Conjecture, which were subsequently used to develop GES (Chickering 2002). The algorithm GUES uses these moves in analogy to GES to search for the optimal DAG without leaving the UEC. In fact, the nature of these moves implies that GUES is also consistent via the proof of consistency of the backward phase of GES, under the assumptions of faithfulness and perfectly correct independence tests. The synthetic data analysis suggests that GUES is a reasonable hybrid approximation to GES, and thus could be a useful alternative when the practitioner is willing to assume a small UEC size.

In regards to future work, the performance of GUES could likely be further optimized by introducing a turning phase, analogous to that of GIES (Hauser and Bühlmann 2012), which empirically improves performance of GES in the purely observational setting. One could also consider how the moves performed by GUES translate into moves on essential graphs to avoid the computational complexity of considering all members of the MEC of maximal DAGs within the UEC. Going further, the major factor limiting the performance of GUES appears to be the independence testing step. Performance could perhaps be improved by considering greedy approaches to estimating the UEC representative. One could also consider extensions of GUES to learning interventional DAG models given a mixture of observational and interventional data, similar to GIES and interventional GSP (Wang et al. 2017).

ACKNOWLEDGEMENTS

Danai Deligeorgaki, Alex Markham, and Liam Solus were partially supported by the Wallenberg Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. Pratik Misra was partially supported by the Brummer & Partners MathDataLab. Liam Solus was partially supported the Göran Gustafsson Stiftelse and Starting Grant No. 2019-05195 from The Swedish Research Council.

References

Andersson S. Madigan, D. and M. Perlman (1997). “A characterization of Markov equivalence classes for acyclic digraphs.” In: *Ann. Statist.* 25.2, pp. 505–541. ISSN: 0090-5364. DOI: 10.1214/aos/1031833662. URL: https://doi.org/10.1214/aos/1031833662.

Chickering, D. M. (1995). “A transformational characterization of equivalent Bayesian network structures.” In: *Proceedings of the eleventh conference on uncertainty in artificial intelligence (UAI)*, pp. 87–98. – (2002). “Optimal structure identification with greedy search.” In: *Journal of machine learning research* 3.Nov, pp. 507–554.

Hauser, A. and P. Bühlmann (2012). “Characterization and Greedy Learning of Interventional Markov Equivalence Classes of Directed Acyclic Graphs.” In: *J. Mach. Learn. Res.* 13.1, pp. 2409–2464. ISSN: 1532-4435.

Linusson, S., P. Restadh, and L. Solus (2021). “Greedy causal discovery is geometric.” In: *arXiv: 2103.03771*. 
Markham, A., R. Das, and M. Grosse-Wentrup (2022). “A distance covariance-based kernel for nonlinear causal clustering in heterogeneous populations.” In: *Proceedings of the 1st Conference on Causal Learning and Reasoning (CLEaR)*. Proceedings of Machine Learning Research. PMLR.

Meek, C (1997). “Graphical Models: Selecting causal and statistical models.” PhD thesis. PhD thesis, Carnegie Mellon University.

Radhakrishnan, A., L. Solus, and C. Uhler (2017). “Counting Markov equivalence classes by number of immoralities.” In: *Proceedings of the 2017 Conference on Uncertainty in Artificial Intelligence (UAI)*.

Roberts, F. S. (1985). “Applications of edge coverings by cliques.” In: *Discrete applied mathematics* 10.1, pp. 93–109.

Scutari, M. (2010). “Learning Bayesian Networks with the bnlearn R Package.” In: *Journal of Statistical Software, Articles* 35.3, pp. 1–22. ISSN: 1548-7660. DOI: 10.18637/jss.v035.i03. URL: https://www.jstatsoft.org/v035/i03.

Solus, L., Y. Wang, and C. Uhler (2021). “Consistency Guarantees for Greedy Permutation-Based Causal Inference Algorithms.” In: *Biometrika*. asaa104. ISSN: 0006-3444. DOI: 10.1093/biomet/asaa104. eprint: https://academic.oup.com/biomet/advance-article-pdf/doi/10.1093/biomet/asaa104/35387259/asaa104.pdf. URL: https://doi.org/10.1093/biomet/asaa104.

Spirtes, P., C. Glymour, and R. Scheines (2000). *Causation, prediction, and search*. Second. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA. ISBN: 0-262-19440-6.

Squires, C. (2018). “causaldag python package.” In.

Tsamardinos, I., L. E. Brown, and C. F. Aliferis (2006). “The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm.” In: *Machine Learning* 65.1, pp. 31–78. DOI: 10.1007/s10994-006-6889-7. URL: https://doi.org/10.1007/s10994-006-6889-7.

Verma, T. and J. Pearl (1990). “Equivalence and Synthesis of Causal Models.” In: *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*. UAI ’90. USA: Elsevier Science Inc., pp. 255–270. ISBN: 0444892648.

Wang, Y. et al. (2017). “Permutation-based causal inference algorithms with interventions.” In: *Advances in Neural Information Processing Systems* 30.
A  SUPPLEMENTARY MATERIAL

Proof of Lemma 11. Since the UEC represented by $\mathcal{U}$ is non-empty, the edge clique cover $\mathcal{E}^U$ of $\mathcal{U}$ is defined. Suppose now that $v$ and $w$ are adjacent in $\mathcal{U}$ but not in the skeleton of the PDAG $\mathcal{G}$, where $\mathcal{G}$ is the output of step 3 of initialize($\mathcal{U}$). Then it follows that $\{v, w\}$ was bidirected in step 2 of initialize($\mathcal{U}$). Hence there are two induced 2-paths $(s, v, w)$ and $(v, w, t)$ in $\mathcal{U}$. Since all edges must be in some clique of $\mathcal{E}^U$, it follows that $v, w \in C_1, s, v \in C_2$ but $w \notin C_2$, and $w, t \in C_3$ but $v \notin C_3$, for some $C_1, C_2, C_3 \in \mathcal{E}^U$. It follows that $\{v, w\}$ is removed from $\mathcal{U}$ when producing $\mathcal{U}$.

Conversely, $\mathcal{U}$ is produced by removing any edge in $\mathcal{U}$ for which there exist $C_1, C_2, C_3 \in \mathcal{E}^U$ such that $v, w \in C_1, v \in C_2$ but $w \notin C_2$, and $w \in C_3$ but $v \notin C_3$. It follows that there exist $s \in C_2$, not adjacent to $v$, and $t \in C_3$, not adjacent to $v$. Thus, there are two induced 2-paths $(s, v, w)$ and $(v, w, t)$ in $\mathcal{U}$, which implies that the edge $\{v, w\}$ is removed from $\mathcal{U}$ in step 3 of initialize($\mathcal{U}$).

Proof of Lemma 12. Suppose otherwise. Then there is a v-structure $v \rightarrow x \leftarrow w$ in $\mathcal{G}$, the output of initialize($\mathcal{U}$), that is not present in the output of step 3 of initialize($\mathcal{U}$). Since $(v, x, w)$ was an unoriented induced 2-path in the PDAG produced in step 3, we must have that $v$ and $w$ are adjacent in $\mathcal{U}$. It follows that $\{v, w\}$ was bidirected in step 2. Thus, there must exist $s$ adjacent to $v$ but not $w$ and $t$ adjacent to $w$ but not $v$ in $\mathcal{U}$, yielding two induced 2-paths $(s, v, w)$ and $(v, w, t)$.

Since the edges $(s, v)$ and $(t, w)$ are oriented as $s \rightarrow v$ and $t \rightarrow w$, respectively in step 2, but $(v, x)$ and $(w, x)$ are undirected, it must be that we also have the edges $(s, x)$ and $(t, x)$ in $\mathcal{U}$.

Since $v$ and $t$ are assumed to be non-adjacent in $\mathcal{U}$ (similarly for $s$ and $w$), we get induced 2-paths $(v, x, t)$ and $(s, x, w)$ in $\mathcal{U}$. However, such paths would be oriented as $v \rightarrow x \leftarrow t$ and $s \rightarrow x \leftarrow w$ in step 2, contradicting the initial assumption that the v-structure $v \rightarrow x \leftarrow w$ was not already present after step 2.

Proof of Lemma 13. Let $v, x, w$ be such that $v, x \in C_1$ but $w \notin C_1$, and $w, x \in C_2$ but $v \notin C_2$, and any clique containing $v$ or $w$ also contains $x$, where $C_1, C_2 \in \mathcal{E}^U$. As $v$ and $x$ (similarly $w$ and $x$) are adjacent in $\mathcal{U}$, there exists a trek between $v$ and $x$ and a trek between $w$ and $x$ in $\mathcal{H}$ by Theorem 2 (2). If $v$ is the source node in the trek between $v$ and $x$, then $v \rightarrow x$ is implied by transitivity and maximality of $\mathcal{H}$ within the UEC of $\mathcal{U}$. If not, then $pa(v) \neq \emptyset$. Moreover, as every clique containing $v$ also contains $x$, any clique containing $v$ and a parent of $v$ would also contain $x$. So by Theorem 2 (2), we know that there exists a trek we have $ma(pa(v)) \subseteq ma(pa(x) \setminus v)$. We conclude that $\{v, x\}$ is partially weakly covered and hence $v \rightarrow x$ lies in $\mathcal{H}$, since $\mathcal{H}$ is maximal in the UEC of $\mathcal{U}$. By symmetry, it follows that the edge $w \rightarrow x$ is also in $\mathcal{H}$. Since from the the proof of Lemma 7 we know that the skeleton of $\mathcal{H}$ is the same as the skeleton of $\mathcal{U}$, there cannot be an edge between $v$ and $w$ in $\mathcal{H}$; otherwise, $\{v, w\}$ would be an edge in $\mathcal{U}$, such that $v, x \in C_1$ but $w \notin C_1$, $w, x \in C_2$ but $v \notin C_2$, and there is a clique $C_3 \in \mathcal{E}^U$ containing the edge $\{v, w\}$, a contradiction. Hence, $v \rightarrow x \leftarrow w$ is a v-structure in $\mathcal{H}$.

On the other hand, if $v \rightarrow x \leftarrow w$ is a v-structure in $\mathcal{H}$, then $v \in \text{an}(x) \cap \text{an}(v)$ and $w \in \text{an}(x) \cap \text{an}(w)$. By Theorem 2 (1), the edges $\{v, x\}$ and $\{w, x\}$ are in $\mathcal{U}$. Moreover, since $v \rightarrow x$ and $w \rightarrow x$ are oriented as $v \rightarrow x$ and $w \rightarrow x$ in $\mathcal{H}$, any trek between a node $y$ of $\mathcal{H}$ and $v$ or $w$ can be extended to a trek between $y$ and $x$. Hence, by Definition 1, if $\{y, v\}$ is in $\mathcal{E}^U$ then $\{y, x\}$ is also in $\mathcal{E}^U$. Thus, any clique in $\mathcal{E}^U$ containing $v$ has to also contain $x$. By symmetry, we also have that every clique in $\mathcal{E}^U$ containing $w$ has to contain $x$. Lastly, we need to show that there exists a clique in $\mathcal{E}^U$ containing $v$ and $x$ but not $w$, as well as a clique containing $w$ and $x$ but not $v$. If $\{v, w\} \notin \mathcal{U}$, then we have the desired clique structure since any clique containing $v$ also has to contain $x$ but not $w$, and any clique containing $w$ has to contain $x$ but not $v$. In case $\{v, w\} \in \mathcal{U}$, then we know from the proof of Lemma 7 that the skeleton of $\mathcal{H}$ is the same as the skeleton of $\mathcal{U}$. Since $v \rightarrow x \leftarrow w$ is a v-structure in $\mathcal{H}$, the edge $\{v, w\}$ must not be present in $\mathcal{U}$. Hence, there exist cliques $C_1, C_2 \in \mathcal{E}^U$ such that $v$ (and hence $x$) is in $C_1$ but $w \notin C_1$, and $w$ (and hence $x$) is in $C_2$ but $v \notin C_2$.

Proof of Corollary 14. It follows from the proof of Lemma 7 that any two maximal DAGs in the UEC represented by $\mathcal{U}$ have the same skeleton, $\mathcal{U}$. In Lemma 13, it was shown that the v-structures of a maximal DAG $\mathcal{H}$ in the UEC are independent of the choice of $\mathcal{H}$. Therefore, any two maximal DAGs in the UEC have the same skeleton and v-structures, which means that they are Markov equivalent.