General Identification of Dynamic Treatment Regimes Under Interference

Eli Sherman  
Johns Hopkins University

David Arbour  
Adobe Inc.

Ilya Shpitser  
Johns Hopkins University

Abstract

In many applied fields, researchers are often interested in tailoring treatments to unit-level characteristics in order to optimize an outcome of interest. Methods for identifying and estimating treatment policies are the subject of the dynamic treatment regime literature. Separately, in many settings the assumption that data are independent and identically distributed does not hold due to inter-subject dependence. The phenomenon where a subject’s outcome is dependent on his neighbor’s exposure is known as interference. These areas intersect in myriad real-world settings. In this paper we consider the problem of identifying optimal treatment policies in the presence of interference. Using a general representation of interference, via Lauritzen-Wermuth-Freydenburg chain graphs (Lauritzen and Richardson, 2002), we formalize a variety of policy interventions under interference and extend existing identification theory (Tian, 2008; Sherman and Shpitser, 2018). Finally, we illustrate the efficacy of policy maximization under interference in a simulation study.

1 Introduction

In areas such as precision medicine, economics, and political science, identifying interventions that are optimally tailored to each subject is often of interest. Dynamic treatment regimes (DTRs), which are counterfactual policies used for treatment assignment, represent a promising approach to tailoring treatments. Typically, a causal model is assumed to be known, with analyst-specified exposure and outcome variables. The analyst considers setting the exposure variable according to a treatment policy which is a function of other model variables. She then estimates the counterfactual effect of several candidate policies and picks the one with the best expected outcome. This setup has been extended to sequential settings (Laber et al., 2014, Chakraborty and Moodie, 2013, Nabi et al., 2018).

A key obstacle to obtaining optimal strategies from observational data is identification. The effect of an intervention in a causal model is said to be identified if the effect can be expressed as a function of observed data. In algebraic formulations, identification requires carefully enumerating necessary assumptions. In contrast, causal graphical models provide a concise framework for representing assumptions, with numerous general results characterizing identification criteria. In the context of DTRs, Robins (1986) gave an approach for identification of node (i.e., fixed value) and policy interventions in fully-observed directed acyclic graphs (DAGs). Tian (2008) and Shpitser and Sherman (2018) extended this approach to latent-variable DAG models.

Orthogonal to treatment customization, classical causal inference assumes independence among study subjects. In many settings, however, subjects’ exposures causally affect their neighbors’ outcomes. This phenomenon, known as interference (Cox, 1958), has recently attracted substantial attention. Hudgens and Halloran (2008) serves as a seminal paper; it defines network-level effects and provides elementary identification conditions. Ogburn et al. (2014) formalizes DAG representations of interference. Several papers propose relational (Maier et al., 2013) or chain graph representations of interference (Peña, 2018, Tchetgen et al., 2017, Ogburn et al., 2018). Sherman and Shpitser (2018) is closest to the present work; it explores non-parametric identification in the presence of unobserved confounding.

A recent paper also considers policies under interference (Viviano, 2019). Our work differs substantially: Viviano focuses on welfare maximization and assumes units are identically distributed. Our characterization of policy interventions generalizes welfare maximization and our network representation is non-parametric.
Motivating Policies in Networks. In this work, we consider identification of DTDRs in the interference setting. As motivation, consider the following example from psephology (the study of elections) (Blackwell 2013): candidates running for public office target voters by purchasing television advertisements; each candidate must decide how many ads to buy and whether they should be positive (“my record is stellar”) or negative (“my opponent is scandalous”).

These dynamics can be represented via the causal evaluation throughout this manuscript, our contributions to networks of arbitrary size and topology.

We first fix notation before describing the task. We employ segregated graphs (SGs) (Shpitser 2015) to represent causal network dynamics. SGs are a class of mixed graphical model which are a super-model of latent-variable LWF chain graphs (CGs), which are themselves a super-model of Markov random fields (MRFs) and DAGs. SGs permit three edge types – undirected (−), directed (→), and bi-directed (↔) – and have the property that no variable has both an incident undirected and bi-directed edge.

We adopt standard graphical model notation. We denote random variables (interchangeably, vertices in graphs) by capital letters $V$ and their realizations in lowercase $v$, with sets in boldface, $\mathbf{V}$ and $\mathbf{v}$. We use standard genealogical notions for graphical relationships. For a variable $V \in \mathbf{V}$ in a graph $G$, parents $\text{pa}_G(V) \equiv \{W \in \mathbf{V} : W \rightarrow V \text{ in } G\}$, children $\text{ch}_G(V) \equiv \{W \in \mathbf{V} : V \rightarrow W \text{ in } G\}$, ancestors $\text{an}_G(V) \equiv \{W \in \mathbf{V} : W \rightarrow \cdots \rightarrow V \text{ in } G\}$, descendants $\text{de}_G(V) \equiv \{W \in \mathbf{V} : V \rightarrow \cdots \rightarrow W \text{ in } G\}$, neighbor $\text{nb}_G(V) \equiv \{W \in \mathbf{V} : W \leftrightarrow V \text{ in } G\}$, non-descendant $\text{nd}_G(V) \equiv \mathbf{V} \setminus \text{de}_G(V)\}$, and district $\text{dis}_G(V) \equiv \{W \in \mathbf{V} : V \leftrightarrow \cdots \leftrightarrow V \text{ in } G\}$.

Further, the anterior $\text{ant}_G(V)$ is the set of nodes with a partially directed path – a path containing only $\rightarrow$ and $\leftarrow$ edges such that no set of undirected edges can be oriented to form a directed cycle – into $V$. The exterior $\text{ext}_G(V)$ is the set of nodes with a partially directed path out of $V$. In turn, the strict exterior $\overline{\text{ext}}_G(V) \subseteq \text{ext}_G(V)$ omits $V$ and the set $\{W \in \mathbf{V} : W \leftarrow \cdots \leftarrow V\}$. By convention, $\text{ext}_G(V) \cap \text{ant}_G(V) \cap \text{dis}_G(V) = \{V\}$.

These and the above notions can be extended to sets, e.g., for $S \subseteq \mathbf{V}$, we have $\text{pa}_G(S) = \cup_{S \in \mathbf{S}} \text{pa}_G(S)$ and, disjunctively, $\text{pa}_G^c(S) = \text{pa}_G(S) \setminus S$. When the relevant graph is clear from context, we drop the $G$ subscript.

For graphs with a partial ordering $\prec$ on $\mathbf{V}$, let $\mathbf{V} \prec A$ denote A’s predecessors in the ordering. For a set $S \subseteq \mathbf{V}$ in $G$, let $G_S$ refers to the subgraph of $G$ containing only $S$ and edges connecting nodes in $S$.

Finally, we use the notion of a block to refer to a set of variables connected by an undirected path. A node with no incident bi-directed nor undirected edges is a trivial block and a trivial district. The sets of blocks, non-trivial blocks, districts, and cliques in $G$ are denoted by $\mathcal{B}(G)$, $\mathcal{B}^{\text{nt}}(G)$, $\mathcal{D}(G)$, and $\mathcal{C}(G)$ respectively. In segregated graphs $\mathcal{D}(G)$ and $\mathcal{B}^{\text{nt}}(G)$ partition $\mathbf{V}$.

2.1 Statistical Graphical Models

Segregated graphs and their submodels were originally conceived as statistical models over random variables, encoding conditional independences in their factoriza-
tion. For instance, a distribution \( p(V) \) is ‘Markov relative to’ a CG \( \mathcal{G}(V) \) if it factorizes according to the two-level factorization

\[
\prod_{B \in B(\mathcal{G})} \frac{p(B)}{p(a(B))} = \prod_{B \in B(\mathcal{G})} \frac{\prod_{C \in C^*} \phi_C(C)}{Z(pa(B))},
\]

where \( C^* = \{ C \in C(\mathcal{G}^*_Bpa_q(B)) : C \not\subseteq pa_q(B) \} \), and \( Z \) is a normalization function. \( \mathcal{G}^*_Bpa_q(B) \) is an augmented graph (Lauritzen 1996): it is undirected and contains edges in \( \mathcal{G} \) between \( B \), edges between nodes in \( pa_q(B) \) and their children in \( B \), and edges between parents. For the corresponding factorizations for DAGs and MRFs, please see the supplementary materials.

2.2 Causal Graphical Models

In contrast to statistical graphs, causal graphs represent distributions over counterfactual variables. For \( Y \in V \) and \( A \subseteq V \setminus Y \), the counterfactual \( Y(a) \) denotes \( Y \)'s value under the hypothetical scenario in which \( A \) is set to \( a \) via a node intervention (Pearl 2000).

In this paper, we assume Pearl’s functional model. In DAGs, counterfactuals \( V(a) \) are determined by structural equations \( f_Y(a, \epsilon_Y) \), which remain invariant under an intervention \( \alpha \); \( \epsilon_Y \) denotes an exogenous random variable for \( f_Y \). By recursive substitution, we can define all other variables in the model: for \( A \subseteq V \setminus \{ V \} \) and \( a \) in the state space of \( A \), \( p(V(a)) \) (sometimes written as \( p(V|do(a)) \) (Pearl 2000)) is defined as \( V(a_{pa(V)}), \{ W(a) : W \in pa(V) \setminus A \} \).

Causal CGs follow similar semantics. Each variable \( B \) in a block \( B \) is determined by a structural equation \( f_B(B \setminus B, pa(B), \epsilon_B) \), a function of other variables in \( B \), the parents of \( B \), and an exogenous variable. Each \( B \)'s joint distribution \( B \) is obtained by Gibbs sampling over the structural equations for \( B \) until equilibrium (trivial blocks equilibrate instantly). Assuming an ordering on blocks in \( \mathcal{G} \), but not on variables in each block, and iid realizations of \( \epsilon_B \), the data generating process for CGs is given by Procedure 1 (Lauritzen and Richardson 2002).

**Procedure 1 CG Data Generating Process**

1: **procedure** CG-DGP(\( \mathcal{G}, \{ f_B : B \in V \} \))
2: **for each** block \( B_i \in B(\mathcal{G}) \)
3: **repeat**
4: **for each** variable \( B_j \in B_i \)
5: \( B_j \leftarrow f_{B_j}(B_i \setminus B_j, pa_{\mathcal{G}}(B_i), \epsilon_{B_j}) \)
6: until equilibrium
7: **return** \( V \)

A parameter is identifiable in a causal model if it can be expressed as a function of observed data. In fully observed DAGs and CGs, all node intervention counterfactuals are identified by the \( g \)-formula (Robins 1986) and chain graph \( g \)-formula (Lauritzen and Richardson 2002) respectively (first two rows of Table 1).

3 Identification in Latent-Variable Causal Graphical Models

In this section, we review identification theory in latent variable causal models. The current work bridges these literatures: we posit a sound and complete algorithm for the identification of responses to policies in latent variable (LV) causal CGs.

3.1 Re-expressing the ID Algorithm

Tian and Pearl (2002) gave a general condition for identification of node interventions in latent-variable DAGs. Shpitser and Pearl (2006) re-expressed Tian’s condition as a concise algorithm and proved that it is complete. Recently, Richardson et al. (2017) rephrased the algorithm in terms of a recursive fixing operator which acts as a modified nested Markov factorization.

Richardson et al. (2017) makes clear the connections between the ID algorithm, which is a modified nested factorization of acyclic directed mixed graphs (ADMGs), and the \( g \)-formula (Table 1 first row), which is a modified DAG factorization. This formalism enables straightforward generalizations to other identification settings. For these reasons, we base our SG policy identification results on this framework. The framework relies on several concepts which we highlight here; each existing ID approach is summarized in Table 1. For a complete treatment, please see the supplement.

**Latent Projections.** Rather than considering LV-DAGs explicitly, Richardson et al. (2017) considers ADMGs. ADMGs permit directed and bi-directed edges and represent equivalence classes of LV-DAGs. Given an LV-DAG \( \mathcal{G}(V \cup H) \), with \( V \) observed and \( H \) latent, the corresponding ADMG \( \mathcal{G}(V) \) is obtained via a latent projection operation (Verma and Pearl 1991). For example, Fig. (1) is the latent projection of Fig. (1) (a). We also define conditional ADMGs (CADMGs), which partition nodes into random \( V \) and fixed \( W \) variables. CADMGs with \( W = \emptyset \) are trivially ADMGs.

Segregated graphs are the chain graph analogue of ADMGs, where SGs represent an equivalence class of LV-CGs. For a latent variable CG \( \mathcal{G}(V \cup H) \), \( H \) is block-safe (Sherman and Shpitser 2018) if no \( V \in V \) has a latent parent and no latent \( H \in H \) has an incident undirected edge. By applying the same latent projection operation mentioned above to a LV-CG with block-safe \( H \), one obtains the corresponding SG.
General Identification of Dynamic Treatment Regimes Under Interference

<table>
<thead>
<tr>
<th>Graph Type</th>
<th>Latents</th>
<th>Intervention Type ( Y^* )</th>
<th>Modified Factorization</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAG</td>
<td>No</td>
<td>Node – ( a )</td>
<td>( Y ) \mid \sum_{V \setminus A} b(V) \mid pa(V) \mid A = a )</td>
</tr>
<tr>
<td>CG</td>
<td>No</td>
<td>Node – ( a )</td>
<td>( Y ) \mid \sum_{B \setminus {a}} b(B) \mid pa(B) \mid B \cap A \mid A = a )</td>
</tr>
<tr>
<td>ADMG</td>
<td>Yes</td>
<td>Node – ( a )</td>
<td>( Y ) \mid \sum_{D \setminus {GV}} d(V) \mid pa(V) \mid D \mid A = a )</td>
</tr>
<tr>
<td>SG</td>
<td>Yes</td>
<td>Node – ( a )</td>
<td>( Y ) \mid \sum_{D \setminus {GV}} d(V) \mid pa(V) \mid D \mid A = a )</td>
</tr>
<tr>
<td>ADMG</td>
<td>Yes</td>
<td>Policy – ( f_A )</td>
<td>( Y ) \mid \sum_{D \setminus {GV}} d(V) \mid pa(V) \mid D \mid A = a )</td>
</tr>
</tbody>
</table>

Table 1: Summary of existing identification approaches. The first two rows use standard g-formulas, the third row is the ID algorithm, and the final two extend ID. The present work generalizes the last two rows. In the fifth row, \( \tilde{a} = \{A = f_A(W_A) | A \in pa_D(D) \cap A \} \) if \( pa_D(D) \cap A \neq \emptyset \) and \( \tilde{a} = \emptyset \) otherwise.

Kernels and Fixing. Whereas DAGs and CGs factorize as products of conditional distributions, ADMGs and SGs factorize as products of kernels [Lauritzen, 1996]. A kernel \( q_{\mathcal{G}}(V|W) \) is a function, mapping values of \( W \) to normalized densities on \( V \). For some \( A \subseteq V \), conditioning and marginalization are defined as:

\[
g(A|W) = \sum_{V \setminus A} q(V|W): q(V \setminus A|A, W) = \frac{q(V|W)}{q(A|W)}.
\]

The notion of fixing variables is closely tied to kernels. In a CADMG \( \mathcal{G}(V, W) \), a variable \( V \) is fixable if \( de(V) \cap \text{dis}(V) = \emptyset \). In a DAG \( \mathcal{G} \) with corresponding distribution \( p(V) \), fixing \( V \) corresponds to applying the g-formula to obtain a new distribution \( p(V \setminus V) \) and a new graph \( \mathcal{G}' \). For a CADMG \( \mathcal{G}(V, W) \) with corresponding kernel \( q(V|W) \), Richardson et al. (2017) defines similar operators, denoted \( \phi_V(\mathcal{G}) \) and \( \phi_V(q; \mathcal{G}) \). These operators yields a new CADMG \( \mathcal{G}'(V \setminus \{V\}, W \cup \{V\}) \) in which all edges into \( V \) are removed and a new kernel \( q'(V \setminus \{V\}|W \cup \{V\}) \) defined:

\[
q'(V \setminus \{V\}|W \cup \{V\}) \equiv \sum_{V \setminus A} q(V|W) = \tilde{q}_{\tilde{V}}|pa_{\tilde{V}}(\text{dis}(V)) \cup \text{dis}(V), \mathcal{W} \}
\]

respectively. These operators were used to define the nested Markov model.

Fixability also extends to sets of variables \( S \subseteq V \) in an ADMG \( \mathcal{G}(V) \) when \( q_{\mathcal{V}} \) is in the nested Markov model. If it is possible to form a sequence \( S_1, S_2, \ldots \) of the variables in \( S \) such that \( S_1 \) is fixable in \( \mathcal{G} \), \( S_2 \) is fixable in \( \phi_{S_1}(q_{\mathcal{V}}; \mathcal{G}) \) and so on, then \( S \) is fixable in \( \mathcal{G}(V \setminus S) \) \( \mathcal{G} \) and is said to be reachable in \( \mathcal{G} \). Since all valid fixing sequences on \( S \) yield the same CADMG \( \mathcal{G}(V \setminus S) \) via \( \phi \), and if \( p(V) \) is nested Markov with respect to \( \mathcal{G}(V) \), all fixable sequences on \( S \) valid in \( \mathcal{G}(V) \) yield the same kernel \( q'((V \setminus S)|S) \) via \( \phi \), the fixable operators can be defined for sets unambiguously: \( \phi_S(\mathcal{G}) \) and \( \phi_S(q; \mathcal{G}) \).

This notation permits reformulating the ID algorithm. For an ADMG \( \mathcal{G}(V) \), let \( Y, A \subseteq V \) be disjoint and \( Y^* \equiv \{Y \mid \text{do}(a) \} \) is identified in \( \mathcal{G} \) if and only if every district \( D \in D(\mathcal{G}) \) is reachable in \( \mathcal{G} \). If identified, \( p(Y | \text{do}(a)) \) is given by summing the modified factorization in row three of Table 1 over \( Y^* \setminus Y \).

Returning to our elections example (Fig. 1(c)), suppose we assume each candidate’s decision is independent of other decisions given covariates (i.e., no \( A_i - A_i \) edge). We can use this formula to consider the effect on a candidate’s polling of advertising positively and negatively in fixed proportion (say, equally, \( a = .5 \)).

As another example, consider the subgraph on \( C_1, A_1, M_1, Y_1 \) in Fig. 2(a); \( p(Y_1 | \text{do}(a_1)) \) is not identified (Shpitser and Pearl, 2006). In the \( C_2, A_2, M_2, Y_2 \) subgraph, however, \( p(Y_2 | \text{do}(a_2)) \) is identified by the front-door formula:

\[
\sum_{M_2, C_2} p(M_2 | a_2, C_2)p(C_2) \sum_{A_2} p(Y_2 | M_2, C_2, A_2)p(A_2 | C_2)
\]

3.2 Identification in Segregated Graphs

The Segregated Factorization. Extending the factorizations for ADMGs and CGs, Sherman and Shpitser (2018) defines the segregated factorization for SGs.

Recall that an SG \( \mathcal{G} \) is partitioned by variables that lie in non-trivial blocks, denoted \( B^* = \cup_{B \in B^*} B \), and those that don’t, denoted \( D^* = \cup_{D \in D} D \). An SG satisfying the segregated factorization can be expressed as the product of kernels for these two sets.

The first kernel, \( q(B^* | pa^*_B(B^*)) = \prod_{B \in B^*} p(B) | pa_B(B) \) factorizes with respect to a conditional chain graph (CCG) \( \mathcal{G}(V, W) \), which we denote by \( \mathcal{G}^b \) with \( V \) corresponding to \( B^* \) and \( W \) to \( pa^*_B(B^*) \). \( \mathcal{G}^b \) contains edges between nodes in \( B^* \) and between nodes in \( pa^*_B(B^*) \) that exist in \( \mathcal{G} \).

The second kernel, \( q(D^* | pa^*_D(D^*)) = \frac{p(D)}{q(B | pa_B(B))} \) factorizes with respect to a CADMG denoted \( \mathcal{G}^d \), with random nodes \( D^* \) and fixed nodes \( pa^*_D(D^*) \). Like \( \mathcal{G}^b \), \( \mathcal{G}^d \) contains edges between nodes in \( D^* \) and between nodes in \( pa^*_D(D^*) \) that are present in \( \mathcal{G} \).

For example, in the graph in Fig. 2(a), we have

\[
q(D^* | pa^*_D(D^*)) = p(Y_2, Y_3, A_2(C_2, M_2, M_3) \times p(Y_1, A_1, C_1 | M_1)p(A_3 | C_3)
\]
which correspond to Fig. 2(b) and (c) respectively.

### The Segregated Graph ID Algorithm

We can now describe an extension of the ID algorithm for node interventions in segregated graphs. For a SG \( G(V) \), fix disjoint \( Y, A \subseteq V \). Let \( Y^* \equiv \text{ant}_{G \setminus A}(Y) \). Define \( G^d \) and \( G^e \) to be the CADMG and CCG respectively obtained from \( G \). \( p(Y|\text{do}(a)) \) is identified in \( G \) if and only if each \( D \in \mathcal{D}(G^d) \) is reachable in \( G^d \). If identified, \( p(Y|\text{do}(a)) \) is equal to the modified factorization in row four of Table 1 summed over \( Y^* \setminus Y \).

Coming back to our elections example, Fig. 1(b), this formula is applicable when considering the effect of the left-leaning candidate taking a fixed action \( a_l \), with the right-leaning candidate’s action still having an impact on the left’s poll standing. \( p(Y_l|a_l) \) is identified by:

\[
\sum_{C_l, C_r, A_l, A_r} p(Y_l, Y_r|C_l, C_r, A_l, a_l)p(A_2|C_l, C_r)p(C_l)p(C_r)
\]

### 3.3 Policy Interventions in ADMGs

Extending node interventions, we now consider policy interventions. For an ADMG \( G(V) \) with topological ordering \( \prec \) on \( V \) and an intervention set \( A \subseteq V \), let \( f_A \) be the set of policies \( \{f_A : A \in A\} \). Each \( f_A \) is a stochastic function of some \( W_A \subseteq V_{<A} \), where \( f_A(W_A) \) maps the state space of \( W_A \) to the state space of \( A \). Intervening with \( f_A \) corresponds to removing edges \( \text{into} \ A \) in \( G \) and adding edges from \( W_A \) to \( A \), yielding a new graph \( G_{f_A} \).

\[\text{Tian} (2008)\] gave a policy-analogue of the ID algorithm for \( p((V \setminus A)(f_A)) \), which \[\text{Shpitser and Sherman} (2018)\] re-expressed via the fixing operator \( \phi \). Let \( Y^* \equiv \text{ant}_{G_{f_A}}(Y) \setminus A \). A policy-analogue of the ID algorithm follows: \( p(Y(f_A)) \) is identified in \( G \) if and only if \( p(Y^*(a)) \) is identified in \( G \); if identified, \( p(Y(f_A)) \) is obtained by summing over \( Y^* \setminus \set{A} \set{Y} \) in the modified factorization in row five of Table 1.

In our elections example, assume candidates’ decisions and outcomes are independent of each other. This formula can be used to consider the effect on a candidate’s polling of advertising based on the relevant covariates, e.g., if the election is less than 2 months away, advertise negatively, and buy positive ads until then.

### 4 Varieties of Policy Interventions

We now describe extensions of policy interventions to network data representable by SGs. These interventions correspond to replacing structural equations in Procedure 1 with new equations, under conditions we describe below such that the resulting data generating process yields a new SG. As we discuss, these policy interventions induce a variety of edge changes in SGs.

#### 4.1 Inducing Direct Causation

As in the latent-variable DAG case, we can intervene by inducing a parent-child relationship between the treatment node and other variables in the graph or modify the nature of existing relationship. In our elections example from Sec. 1, this might correspond to intervening on the left candidate’s decision \( A_l \) such that she adopts a new strategy for responding to her competitor’s characteristics \( C_r \) relative to her (observed) status quo strategy. For illustrative purposes, this type of intervention is demonstrated by the addition of the \( C_2 \rightarrow A_1 \) edge and the modification to the \( C_1 \rightarrow A_1 \) edge between Fig. 2(a) and 2(d).

#### 4.2 Inducing or Modifying Undirected Dependence

We can also consider changing the block structure of the SG. There are two types of such interventions:

1. **Modifying the functional form encoded by an existing undirected edge.** In Fig. 2(b), we can think of the undirected edge \( A_l - A_r \) as representing each candidates’ beliefs about the other candidate’s actions. In the observed data, candidates will best-respond to each other according to these beliefs. We can imagine changing the way one (or both) of the candidates reasons about their opponent’s possible actions, such as making one candidate hyper-responsive to their opponent’s anticipated action. Mechanically, we intervene on \( A_l \) (analogously \( A_r \)) with a function \( f_A \) that takes \( A_r \) as an argument. We needn’t intervene on the other candidate to maintain the undirected edge between the \( A \)’s. This type of intervention is demonstrated by the change to the \( M_2 - M_3 \) edge from Fig. 2(a) to 2(d).

2. **Inducing co-dependence by adding a new undirected edge between two nodes.** This might correspond to having a third candidate \( c \) join the race and intervening such that \( A_c - A_l \) and \( A_c - A_r \). In this case, it is necessary to intervene on both endpoint nodes for the new undirected edge in order; we modify the respective structural equations to take the other endpoint as an argument. We further restrict these interventions by requiring that they do not induce a partially directed cycle, which would violate the segregation property of the graph. We formalize this requirement below. We note that this type of intervention can be thought of as a chain graph generalization of connection interventions, proposed in \[\text{Sherman and Shpitser} (2019)\]. As an example, consider the addition of the \( A_2 - A_3 \) edge...
4.3 Removing Dependence

Finally, we can consider removing undirected dependence between nodes. Once again there are two types:

1. **Partial removal.** We intervene on a single node to make its structural equation no longer a function of the other end point of the undirected edge. In our elections example (Fig. 1(c)), this corresponds to a ‘first mover’ scenario where \( A_i \) is made to not depend on \( A_r \) and thus candidate \( r \) makes her decision before candidate \( i \). Graphically, we change the undirected edge \( A_1 \prec A_r \) to a directed edge \( A_1 \rightarrow A_r \), since \( A_r \) is still determined by candidate \( i \)’s decision; see, for instance, the \( M_1 \rightarrow M_2 \) and \( M_1 \leftarrow M_2 \) edges in Fig. 2(a) and 2(d).

2. **Complete removal.** We remove both dependences by intervening on both endpoints of an undirected edge so that the structural equations are no longer functions of each other. This corresponds to a candidate dropping out of the race in our elections example. Like dependence-inducing interventions above, this intervention type can be viewed as an SG analogue of severance interventions \cite{Sherman and Shpitser 2019}.

5 Identification of Policies in Segregated Graphs

In this section we formalize policy interventions and provide a procedure for obtaining the post-intervention graph from \( G \). We then give a criterion for the identification of policy interventions in SGs \cite{Shpitser 2015} and demonstrate application of this criterion to Fig. 2 and to our electoral example, Fig. 1. We defer proofs and derivations to the supplement.
Definition 1 A policy intervention $f_A(Z_A)$ is ‘segregation preserving’ if (a) for each $A \in A$, $Z_A \subseteq V \setminus \text{cond}(A)$, and (b) for any $A_i, A_j \in A$ if $A_i \Delta A_j$ and $A_j \Delta A_i$, we have that $A_i \in Z_{A_j}$ and $A_j \in Z_{A_i}$.

For a given intervention set $f_A$, we can construct a post-intervention graph $G_{f_A}$ according to Procedure 2, which follows from the analogous procedure for policy identification in LV-DAGs. In Lemma 1 we show that $G_{f_A}$ is an SG when $f_A$ is segregation-preserving. As an example of this procedure’s application, consider Fig. 2(a). Suppose we wish to perform an intervention $f_A(Z_A)$ as in Table 2. Then $G_{f_A}$ is given by Fig. 2

<table>
<thead>
<tr>
<th>$A \in A$</th>
<th>$A_1$</th>
<th>$A_2$</th>
<th>$A_3$</th>
<th>$M_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Z_A$</td>
<td>$C_2$</td>
<td>$C_2, C_3, A_3$</td>
<td>$A_2, C_3$</td>
<td>$A_2, C_2, M_3$</td>
</tr>
</tbody>
</table>

Table 2: Intervention variables $A \in A$ and induced dependences $Z_A$ for the intervention in Fig. 2

5.2 Identification Results

First, we show that the post-intervention $G_{f_A}$ is an SG.

Lemma 1 Given an SG $G(V)$ and a segregation-preserving intervention $f_A(Z_A)$, the post-intervention graph $G_{f_A}$ obtained via Procedure 2 is an SG.

We now present the main result of this paper. This theorem provides sufficient conditions for the identification of the effects of policy interventions in SGs.

Theorem 1 Let $G(V \cup H)$ be a causal LV-CG with $H$ block-safe, and a topological order $\prec$. Fix disjoint $Y, A \subseteq V$. Let $f_A(Z_A)$ be a segregation preserving policy set. Let $Y^* \equiv \text{ant}_{G_{f_A}}(Y) \setminus A$. Let $G^d, G^b$ be the induced C ADMGs on $G_{f_A}$ and $G_{Y^*}$, and $G_h$ the induced CCG on $G_{Y^*}$. Let $q(D^*|p_{G_{f_A}}^a(D^*)) = \prod_{D \in G^a_{f_A}} q(D|p_{G_{f_A}}^a(D))$, where $q(D|p_{G_{f_A}}^a(D)) = \prod_{D \in G^d} p(D|V \setminus D)$ if $D \cap A = \emptyset$ and $q = f_A(Z_A)$ if $D \cap A \neq \emptyset$. $p(Y|f_A(Z_A))$ is identified in $G$ if and only if $p(Y^*(a))$ is identified in $G$ for the unrestricted class of policies. If identified, $p(Y|f_A(Z_A)) = \sum_{(Y^* \cup A) \setminus Y} \left[ \prod_{B \in \text{bad}_{G^b}} p^*(B|p_{G_{f_A}}^a(B)) \right] \times \left[ \prod_{D \in \text{bad}_{G^d}} \phi_{D \setminus D}(q(D^*|p_{G_{f_A}}^a(D^*)):G^d) \right]_{A=\text{bad}}$ (2)

where (a) $\text{bad} = \{A = f_A(Z_A) : A \in p_{\text{var}}(D) \cup A\}$ if $p_{\text{var}}(D) \cap A \neq \emptyset$ and $\text{bad} = \emptyset$ otherwise, and (b) $p^*$ is obtained by running Procedure 2 over functions $g_B(B_i, \text{var}, (B_i), e_B)$ where $g_B \in f_A$ if $B_i \in A$ and $g_B$ is given by the observed distribution if $B_i \notin A$.

The outer sum over $A$ is extraneous if $f_A$ corresponds to a set of deterministic policies.

5.3 Estimands and Optimal Policy Selection

We now demonstrate how to obtain identified functionals via Eq. 2. We describe identification of the effect on $\{Y_2, Y_3\}$ in Fig. 2(a) of the intervention in Table 2 and then give the functional for our elections example, Fig. 1(b), which we estimate in the next section.

From Fig. 2(a), we obtain $G_{f_A}$ in Fig. 2(d) by applying the intervention detailed in Table 2. In turn, from this post-intervention graph we observe that $Y^* = \text{ant}_{G_{f_A}}(Y) \setminus A = \{C_2, C_3, M_2, A_1\}$ and $g_B(M_2, M_3, A_2, A_3, C_2, C_3|\emptyset)$. The block nodes factorize as a product of blocks, as in the first term of Eq. 2. Separately, we must fix sets for each $G_{Y^*}$-district $\{M_5\}, \{Y_2, Y_3\}$ in $q_D$. This yields the functional (full derivation in the supplement) for $p(Y_2, Y_3|f_A))$:

$$\sum_{A_1, A_2, A_3, M_2, M_3, C_2, C_3} p^*(A_2, A_3|C_2, C_3)p^*(M_2, M_3|A_2, A_3, C_2)$$

$$\times p(Y_2, Y_3|Y_1, A_1, M_3, C_1, C_2)p^*(C_2, C_3)$$

Similarly, we consider the effect on $Y_i$ of intervening with a policy $f_A(C_i)$ in our electoral example, Fig. 1(b). $f_A(C_i)$ corresponds to a myopic strategy in which the candidate makes decisions based only on their own covariates. Applying Eq. 2 $p(Y_i(f_A(C_i), C_r)) = \sum_{C_i, C_r, A_i, Y_r} p(A_i|C_i, C_r)p(C_i)p(C_r)$

$$\times p(Y_i, Y_r|C_i, C_r, A_r, f_A(C_i, C_r))$$

To choose an optimal action for the left candidate, we select $f_A(C_i)$ from a set of candidate policies $F_{A_i}(C_i)$:

$$f_A(C_i, C_r) = \arg \max_{f_A(C_i,C_r) \in F_{A_i}(C_i,C_r)} p(Y_i(f_A(C_i, C_r)))$$

6 Estimation

We now demonstrate how functionals identified by Eq. 2 can be estimated from observed data. Specifically, we seek optimal $f_A(C_i)$’s for versions of the functional in Eq. 3. To do so, we fit nuisance models and utilize the plug-in principle to perform indirect Q-learning for $f_A(C_i)$ in our electoral example. We discuss this in the supplement.

This distribution is identified from univariate terms but it cannot be obtained in closed-form.
General Identification of Dynamic Treatment Regimes Under Interference

policy optimization. This approach yields consistent estimates of the optimized outcome under regularity conditions, assuming correctly specified nuisance models (Chakraborty and Moodie, 2013).

For our experiments we first generate 10-node network graphs according to one of three widely-used network generators: Erdős and Rényi (1960), Watts and Strogatz (1998), and Albert and Barabási (2002). In-unit and cross-unit structures are identical to the 2-node graph in Fig. 1(b). We then generate data for each C, A, and Y using log-linear models, with C ∈ [0, 1]^3 and A, Y ∈ [0, 1]. We use Gibbs sampling to approximate undirected edges between Y’s (Tchetgen et al., 2017). We defer parametric specifications of our data generating process to the supplement. We assume partial interference: we generate 1000 samples of each network topology and use these to fit nuisance models. We run the following experiments by obtaining 1,000 bootstrap replications of the generated data and calculating a 95% confidence interval of the relevant effect:

1. **Bias from incorrectly assuming iid.** As a demonstration of the importance of using interference-aware modeling, we consider performing node interventions on each A_i obtained from our Erdős-Rényi samples, setting A_i to 1 and 0. We estimate the average causal effect (ACE) of these node interventions \(E[Y_i|A_i = 1, Y_i] - E[Y_i|A_i = 0, Y_i]\) using models implied by ID (Table 1 row three), which provides sound functionals when data are iid, as well as models implied by the SG ID algorithm (Table 1 row four) which respect the dependent nature of the data. We treat the latter models as ‘ground truth’ and calculate the bias of the ACE induced by inappropriately assuming data are iid. These results are given in Fig. 3a. Observing that bias is universally bounded away from 0 in these results, it’s clear that it’s imperative to respect network dependence in causal modeling.

2. **Benefit of optimizing interventions.** Here we demonstrate the efficacy of policy interventions for picking tailored interventions that optimize a subject’s outcome, by estimating the 10-unit version of the identified functional in Eq. 3. From our generated samples, we fit logistic regression models for \(E[Y_i|A, C]\) and \(E[Y_i|A, C, Y_i]\), where i denotes the unit we wish to optimize for. This ensures the necessary consistency properties for indirect Q-learning. Models for \(p(A, C)\) are estimated using the empirical distribution.

For each sample we estimate the effect of intervening with a policy \(f_{A_i}(C_i) \in \mathcal{F}_{A_i}(C_i) = \{C_i^{-1} \sum_{j} k_j C_{ij} : k_j \in \mathbb{R}\}\) (i.e. \(f_{A_i}\) is the set of means of linear combinations of \(C_i\’s components)). We choose \(k\) to maximize \(Y_i\) subject to the constraint that values of \(A_i\) and \(Y_i\) must remain in [0, 1]. We report the difference between the optimized and observed (‘status quo’) \(Y_i\’s. The results for the Erdős-Rényi generator can be found in Fig. 3b. Results for the other generators can be found in the supplementary material. Since \(Y\) is binary, an expected difference of .05 corresponds to a 5.0% increase in \(Y\) over the status quo. Fig. 3b demonstrates that the proposed approach virtually guarantees an improved outcome over the status quo.

7 Conclusion

In this paper we discussed identification of policy intervention effects in the interference setting. We characterized interpretations of possible interventions and gave criteria for identifying their effects in latent-variable causal chain graph models. Further, we demonstrated estimation via a simulation study. Future directions include exploring the intersection of policies and interference, and game theory, and developing robust estimation strategies for this setting.
Acknowledgements

The first author would like to thank the Adobe Research Internship program, and Sridhar Mahadevan, for supporting this work. The third author would like to thank the following organizations for supporting this work: the American Institute of Mathematics SQuaRE program, National Institutes of Health grant R01 AI127271-01A1, Office of Naval Research grant N00014-18-1-2760, and Defense Advanced Research Progress Administration grant under contract HR0011-18-C-0049. The content of the information in this paper does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

References


