Characterizing Optimal Mixed Policies: Where to Intervene and What to Observe

Sanghack Lee  
Department of Computer Science  
Columbia University  
New York, NY 10027  
sanghacklee@cs.columbia.edu

Elias Bareinboim  
Department of Computer Science  
Columbia University  
New York, NY 10027  
eb@cs.columbia.edu

Abstract

Intelligent agents are continuously faced with the challenge of optimizing a policy based on what they can observe (see) and which actions they can take (do) in the environment where they are deployed. Any policy can be parametrized in terms of these two dimensions, i.e., as a function of what can be seen and done given a certain situation, which we call a mixed policy. In this paper, we investigate several properties of the class of mixed policies and provide an efficient and effective characterization, including optimality and non-redundancy. Specifically, we introduce a graphical criterion to identify unnecessary contexts for a set of actions, leading to a natural characterization of non-redundancy of mixed policies. We then derive sufficient conditions under which one strategy can dominate the other with respect to their maximum achievable expected rewards (optimality). This characterization leads to a fundamental understanding of the space of mixed policies and a possible refinement of the agent’s strategy so that it converges to the optimum faster and more robustly. One surprising result of the causal characterization is that the agent following a more standard approach – intervening on all intervenable variables and observing all available contexts – may be hurting itself, and will never achieve an optimal performance.

1 Introduction

Agents are deployed in complex and uncertain environments where they are bombarded with high volumes of information and are expected to operate efficiently, safely, and rationally. The discipline of causal inference (CI) offers a compelling set of tools and a language that allows one to reason with the structural invariances present in complex environments [1–5]. Whenever the causal mechanisms of an underlying environment are sufficiently well-understood, the agent can design very precise interventions, bringing a certain desired state of affairs about swiftly and cleanly (e.g., personalized medical treatments, inequality-reducing tax policies). In the field of ML, bandits and reinforcement learning (RL) constitute the de facto framework in which agents are designed such that a certain policy is optimized and the corresponding goals can be efficiently achieved [6–8].

There is a growing literature exploring how these two frameworks (RL and CI) are related, and how this understanding can be translated into more efficient decision-making in more challenging and realistic settings. Recently, the more explicit connection between these frameworks has been made by eliciting how causal knowledge – unobserved confounders and the causal relations between actions, contexts, and rewards – can be used to improve decision-making in a variety of settings, including for both interventional [9,11] and counterfactual [12–13] reasoning (see also [14–17] and [18–21]). Outside more traditional RL, causal inference researchers have embraced the idea of sequential decision making in terms of conditional plans or dynamic treatment regimes, while focusing on, e.g., the identifiability of causal effects from observational data [22–26].

Preprint. Under review.
In this paper, we consider exploiting causal relationships for systematic decision making in the context of mixed policies. For concreteness, consider an agent deployed in an environment represented as a causal graph $G$ (Fig. 1a), where $C$, $X = \{X_1, X_2\}$, $Y$ represent the context, two action variables, and the reward variable, respectively. Graphically, dashed-bidirected edges represent unobserved confounders (UCs, for short) affecting both ends of the arrow. The agent’s task is to maximize the reward $\mu_{\pi} = \mathbb{E}_{\pi}[Y]$ under a policy $\pi \in \Pi$, where $\Pi$ is a policy space. A standard contextual bandit (CB) optimizes a policy $\pi(x|c)$ (Fig. 1b), a (stochastic) mapping from contexts to actions, which can be represented as a pair of decision rules $\pi(x|c) = \pi(x_1|c)\pi(x_2|x_1,c)$ (Fig. 1c). Unfortunately, the optimized policy $\pi^*$ may be suboptimal (i.e., $\mu_{\pi^*} = \mu_c^* < \mu^*$, where $\mu^*$ is the optimal expected reward). To ground what this means, let every variable be binary and $U_1$ and $U_2$, the UCs adjacent to $X_1$ and $X_2$, be fair coins and $\epsilon$ be a noise over $X_1$ following $P(\epsilon = 1) = 0.2$. Also, let the unobserved causal mechanisms be specified as $X_1 \leftarrow U_1 \oplus \epsilon, C \leftarrow U_1, X_2 \leftarrow U_2 \oplus X_1 \oplus C,$ and $Y \leftarrow (1 - (X_2 \oplus U_2)) \lor C$, where $\oplus$ is the exclusive-or operator. Since the policy determines $X_2$, irrelevant to $U_2$ and $C$ is independent to $U_2$, we can elicit that $\mu_{\pi^*} = 0.75$. In this setting, the best policy is intervening only on $X_1$ given $C$, i.e., $\pi(x_1|c)$ as depicted in Fig. 1d. With $X_1 = C$, the policy suppresses the noise $\epsilon$ over $X_1$ and makes $X_2 = U_2$ so that $\mu_{(d)}^* = 1.0$.

Given a causal graph, a mixed policy compatible with the graph consists of a set of decision rules on a subset of intervenable variables and the corresponding contexts. In the example of Fig. 1a, there are 15 ways for an agent to interact with the system — where to intervene and what to observe for each intervened (shown in Appendix A). These different modes of interaction can be represented as induced graphs and can be classified based on two desiderata: non-redundancy and optimality. We explain these desiderata through an illustration (Fig. 2) of the four strategies ($a, c, d, e$) above, where we annotate their relationships with a superset symbol $\supset$, whether one strategy has more actions or contexts than the other, and with a comparison symbol $\geq$ (or $=)$, whether one’s optimal reward is at least as good as the other’s. Roughly speaking (to be formalized later on), non-redundancy refers to the condition of a strategy such that removing any of its actions or contexts can negatively affect the performance of the policy. Since $(c) \supset (e)$ while $\mu_{(c)}^* = \mu_{(e)}^*$, the CB policy (Fig. 1c) is redundant and the CB agent wastes its resources not only for intervening on $X_1$ (a redundant action) but also for taking $X_1$ into account for $X_2$ (a redundant context). Furthermore, optimality represents that there is no policy strictly better than the one with respect to their optimal rewards in every world compatible with the graph. For example, $(d)$, when optimized, is at least as good as $(a)$ (i.e., $\mu_{(d)}^* \geq \mu_{(a)}^* = \mu$), and can outperform it in some environments (i.e., $\mu_{(d)}^* > \mu_{(a)}^*$). However, $(e)$ is not comparable to $(a)$ nor $(d)$. Strategies $(c, d, e)$ meet the optimality criterion since no other strategy can be shown to outperform them in any possible model. Both desiderata are satisfied only by strategies $(d)$ and $(e)$. This example demonstrates that an intelligent agent should judiciously intervene on a carefully chosen subset of variables with side information (context) relevant to attaining an optimal reward.

**Contributions** In this work, we investigate mixed policies with respect to their expected rewards. Our contributions are as follows. (i) We developed a graphical criterion that detects the redundancy of contexts relative to a collection of actions taking advantage of properties pertain to optimal mixed policies. (ii) We established sufficient conditions under which one strategy can outperform another,
characterizing the partial order defined over the space of strategies with respect to their maximum expected rewards achievable. We believe these results have practical implications for the design of intelligent agents providing the basis for efficient and effective explorations of the policy space. One fundamental implication of our analysis is that the agent following a standard approach (i.e., intervening and observing whenever possible) may be hurting itself, and, regardless of the number of interactions, will never be able to achieve an optimal performance.

**Preliminaries**  Let us denote a variable by an uppercase letter $X$, whose value is denoted by its corresponding lowercase letter $x$. A set of variables will be denoted by a bold uppercase letter $X$ with its value $x$. We follow notational conventions from literature on measure theory, sets algebra, and causal inference. We use structural causal models (SCMs) \(^1\) Ch. 7 as the semantical framework to represent an underlying environment. An SCM $M$ is a quadruple $\langle U, V, P(U), F \rangle$, where $U$ is a set of exogenous variables following a joint distribution $P(U)$, and $V$ is a set of endogenous variables whose values are determined following a collection of functions $F = \{ f_i \}_{V_i \in V}$ such that $V_i = f_i(v_i, u_i)$ where $V_i \subseteq V \setminus \{ V_i \}$ and $U_i \subseteq U$. The observational distribution $P(v)$ is defined as $\sum_u \prod_{V_i \in V} P(v_i | v_i, u_i) P(u)$. Further, $\text{do}(X=x)$ represents the operation of fixing a set $X$ to a constant $x$ regardless of their original mechanisms. Such intervention induces a submodel $M_x$, which is $M$ with $f_X$ replaced to $x$ for $X \in X$. Then, an interventional distribution $P_x(v | x)$ (or also $P(v | \text{do}(x))$) follows from $M_x$, and is such that $P_x(v | x) = \sum_u \prod_{V_i \in V \setminus X} P(v_i | v_i, u_i) P(u)$.

Graphically, each SCM (model, for short) is associated with a directed acyclic graph (DAG) with latents $G = \langle V, E \rangle$, where each type of edge represents a different relationship among variables: (i) $X \rightarrow Y$ represents a direct causal relationship; and (ii) $X \leftarrow Y$ the existence of an unobserved confounder (UC), that is, $X$ and $Y$ share common unobserved variables as argument for $f_X$ and $f_Y$. From the agent’s perspective, only the causal graph $G$ of the environment $M$ is available, while its reward is validated through $F$. We operate in the non-parametric setting, where no assumption about the form or shape of the pair $P(U)$, $F$ is made, but for the structural knowledge encoded in $G$. Whenever not even $G$ is known, the agent can perform active interventions to learn it; for example, see \(^2\)[27,28]. We denote by $G_{XZ}$ an edge subgraph of $G$ which removes edges incoming to $X$ and outgoing from $Z$. A submodel $M_x$ can be presented as $G_{XZ}$ with $X$ fixed to $x$. Hence, causal relationships among other variables are captured in $G_{XZ}$, which is the subgraph of $G$ over $V \setminus X$. We denote by $G(V')$ the latent projection of $G$ onto $V'$, the causal graph retaining causal relationships among $V'$ \(^3\) [29]. We adopt familial notation, $ch$, $pa$, $an$, $de$ for children, parents, ancestors, and descendants, respectively, with $Ch$, $Fa$, $An$, $De$ including arguments. Our work utilizes d-separation \(^4\) [30,31] and do-calculus \(^5\) [32], classic graphical rules to ascertain equalities between distributions (for further details, see Appendix B). Also, the omitted proofs and derivations are provided in the appendix.

2 **Mixed Policies: Fundamentals & Basic Results**

One of the main tasks in decision-making is to optimize the parameters associated with a specific policy. The scope of each policy is usually fixed, in the sense that the set of actions and contexts are pre-specified, a priori. By and large, the literature considers policies with scope that is (1) observational, where the system is allowed to evolve without any intervention; (2) fully experimental, where all the action variables are intervened on and all the context variables are observed. The former tends to be more common in CI while the latter tends to be more common in RL. As discussed in the previous section, a causal understanding of the world gives rise to a rich spectrum of policies with different scopes, allowing agents to choose how to interact with the environment, meaning, which variables to intervene and to observe. Formally, we can then define the space of mixed policies.

**Definition 1** (Mixed Policy Scope (MPS)). Let $G$ be a causal graph, $Y$ be a specific reward variable, $X^* \subseteq V \setminus \{ Y \}$ a set of interventional variables, and $C^* \subseteq V \setminus \{ Y \}$ a set of contextualizable variables. A mixed policy scope $S$ is defined as a collection of pairs $\langle X, C_X \rangle$ such that $X \in X^*$, $C_X \subseteq C^* \setminus \{ X \}$, and $G_S$ is acyclic, where $G_S$ is defined as $G$ with edges onto $X$ removed and directed edges from $C_X$ to $X$ added for every $\langle X, C_X \rangle \in S$.

For concreteness, given a causal graph $G$ (Fig. 1a), an MPS $S_1 = \{ \langle X_1, \{ C \} \rangle, \langle X_2, \{ X_1, C \} \rangle \}$ induces a graph (Fig. 1c) and $S_2 = \{ \langle X_2, \{ C \} \rangle \}$ induces Fig. 1e The observational case is an MPS

\(^1\)We discuss connections to multi-agent or multi-reward setting in Appendix C
$S_0 = \{\}$. An MPS represents a class of mixed policies that share the same graphical characteristics manifested by $G_S$, an induced graph for $M_\pi$.

**Definition 2** (Mixed Policy). Given $(G, Y, X^*, C^*)$ and an SCM $M \sim G$ with $X_Y \subseteq \mathbb{R}$, a mixed policy $\pi$ is a realization of a mixed policy scope $S$ compatible with the tuple $\pi = \{\pi_{X|C_X}(x,c_x) : (x,c_x) \in S\}$, where $\pi_{X|C_X} : X_X \times X_C \mapsto [0,1]$ is a proper probability mapping.

For concreteness, if we consider MPSes $S_0, S_1, S_2$ discussed above, the mixed policies are $\pi_0 = \{\}$, $\pi_1 = \{\pi_1(x_1, x_2) = \{\pi_1(x_1) \}, \pi_1(x_2 | x_1)\}$, and $\pi_2 = \{\pi_2(x_2) \}$, which are specific instantiations of the parameters with respect to the corresponding scope. Given an underlying SCM $M$, a mixed policy $\pi$ induces a variant of SCM $M_\pi$ where the function for $X \in X(\pi)$ is replaced by the corresponding $\pi_{X|C_X}$ (see [33] for a detailed account). We denote by $P_\pi$ the joint distribution over the variables from the system under the policy $\pi$. Throughout the paper, $G, Y, C^*$, and $X^*$ are oftentimes implicit including an underlying SCM $M \sim G$ and, thus, $\Pi$, as well.

**Expected Reward** We define the expected reward of a mixed policy. To begin with, we define intervened variables $X(S) = \{ X \mid (X, C_X) \in S \}$ and active contexts $C(S) = \bigcup_{(X,c_x) \in S} C_X$. Similarly, given $\pi \sim S$ (a mixed policy following the MPS), $X(\pi) = X(S)$ and $C(\pi) = C(S)$. Let $C^- = C(\pi) \setminus X(\pi)$ be the non-action contexts. Then, the expected reward for $\pi$ can be expressed as

$$\mu_\pi = \sum_{y,x,c} yP_y(x,c) \prod_{X_X \in X(\pi)} \pi(x|c_x).$$  (1)

The expression separates the atomic interventional probability (first factor), which is inherent to the underlying world and not affected by the policy $\pi$, from the likelihood of a specific intervention given contexts (second factor), which is optimizable and defined by $\pi$. The expected reward can also be written focusing only on a subset of intervened variables. Given $X' \subseteq X(\pi)$, let $C' = \bigcup_{X_X \in X'} C_X \setminus X'$, and $Q' = P_\pi|_{X'}$. Then, $\mu_\pi = \sum_{y,x',c} yQ'_y(x',c) \prod_{X_X \in X'} \pi(x|c_x)$.

**Optimality and deterministic mixed policy** A mixed policy $\pi$ is said to be optimal in the given environment if and only if $\mu_\pi = \mu^* = \max_{\pi' \in \Pi} \mu_{\pi'}$. Restricting our attention to $\Pi_S = \{ \pi \in \Pi \mid \pi \sim S \}$, we define $\mu^*_S = \max_{\pi' \in \Pi_S} \mu_{\pi'}$, an optimal policy $\pi$ with respect to $S$. We call a mixed policy deterministic if, for every $\pi_{X|C_X} \in \pi$, $x$ is determined by a function of $c_x$.

**Proposition 1**. Given a mixed policy scope, there always exists a deterministic mixed policy, which is optimal with respect to the given scope.

Not surprisingly at this point [34][36], a stochastic policy is no better than the best deterministic policy. Still, this result has a particular importance to the treatment provided here due to its implications to the d-separation criterion [37], which will be instrumental and discussed in depth in Sec. 3.1. Another key implication is shown next.

**Proposition 2** (Separation of Actions and Contexts). Given an MPS $S$, there always exists a deterministic mixed policy $\pi \in \Pi$ such that $X(\pi)$ and $C(\pi)$ are disjoint and $\mu^*_S = \mu_{\pi}$.

A deterministic policy gives rise to the autonomy of each action allowing them to be determined only by non-action contexts. For concreteness, consider the example shown in Fig. 3a. A mixed policy (Fig. 3a) includes $X_2$ listening to $X_1$, which enables some coordination between $X_1$ and $X_2$. The proposition implies that $X_2$ can rather listen to $C$ (which is the context of $X_1$) directly (Fig. 3b). Further, in Fig. 3c, $X_2$ utilizes both $X_1$ and $C$. However, it is sufficient to make use of only $C$. By noting that the policy relative to Fig. 3b can achieve optimality, while being simpler than the one relative to Fig. 3c, we investigate how to capture non-redundancy within MPSes.

---

2Throughout the paper, we usually project graphs onto a subset of $\{Y\} \cup C^* \cup X^*$ and any endogenous variables that are not $\{Y\} \cup C^* \cup X^*$ will be treated as unobserved variables.
3 Non-Redundant Mixed Policy

Optimizing a mixed policy involves assessments of the effectiveness of its scope so that an agent can avoid intervening or observing on unnecessary actions or contexts. Here, we define and characterize non-redundancy of MPS. We say \( S \) subsumes \( S' \), denoted by \( S' \subseteq S \), if \( X(S') \subseteq X(S) \) and \( C_{S'} \subseteq C_X \), for every \( X, C_X \in S' \). Further, we denote by \( \pi' \subseteq \pi \), where \( \pi' \sim S' \) and \( \pi \sim S \) if \( \pi'(x|c_x) = \sum_{c'_x} \pi(x|c_x)c_{X}(c'_x|c_x) \), for every \( X \in X(S') \) where \( C_{X}^\prime = C_X \setminus C'_{X} \).

**Definition 3.** Given \( \{G, Y, X^*, C^*\} \), an MPS \( S \) is said to be non-redundant if there exists an SCM \( M \sim G \) and \( \pi \sim (S, M) \) such that \( \mu_\pi \neq \mu_{\pi'} \) for every \( \pi' \subseteq \pi \).

The constraint on \( \pi' \) ensures that the definition of non-redundancy of MPS is focused on the differences in actions or contexts while the behavior (i.e., \( \pi(\cdot|\cdot) \)) remains the same — \( \pi'(x|c_x) = Q(x|c_x) \) if \( C_{X}^\prime \neq C_X \) and \( Q(x|c_x) = \pi'(x|c_x) = \pi(x|c_x) \), otherwise. Hence, the constraint provides a basis to characterize non-redundancy of MPS utilizing well-established graphical criteria.

**Theorem 1.** Let \( S = \{\langle X, C_X \rangle \}_{X \in X} \) be an MPS and let \( H = \mathcal{G}_S \). \( S \) is non-redundant if and only if (i) \( X \not\subseteq an(Y) \) and (ii) \( (C \nsubseteq Y \mid C_X \setminus \{C\}) \) in \( H \setminus \{X\} \), for every \( X \in X \) and \( C \in C_X \).

The condition (i) can be seen through rule 3 of do-calculus such that the change of the mechanism of \( X \) has a consequence on the reward \( Y \). The condition (ii) coincides with rule 2 of do-calculus \( Q(y|x, c_x \setminus \{c\}) = Q_x(y|c_x \setminus \{c\}) \), where \( Q = P_x \). In words, the path from \( C \) to \( Y \) can be concatenated with \( X \rightarrow C \) to form a back-door path from \( X \) to \( Y \). Consider the example in Fig. 4, where both \( X_1 \) and \( X_2 \) are ancestors of \( Y \) (condition (i)). Regarding condition (ii), \( C_1 \) is adjacent to \( Y \) and \( C_2 \) has a path to \( Y \) through \( X_3 \), and a path \( C_3 \rightarrow C_2 \rightarrow C_1 \rightarrow Y \) demonstrates that every context is non-redundant. We provide an efficient algorithm for obtaining a unique, maximal, non-redundant MPS (nr-mps, Alg. 2) of a given MPS in Appendix B.

### 3.1 Non-Redundancy under Optimality

Non-redundancy of MPS (Def. 3) based on a stringent constraint imposed on \( \pi' \) is insufficient to understand, e.g., whether a context of an action would be still relevant even when \( \pi \sim S \) is fully-optimized. Hence, we characterize the non-redundancy of MPS under optimality, which has practical implications to an agent adapting its suboptimal policy. Recall Fig. 4, where \( X_2 \) listens to \( X_1 \) as context. We showed that the dependence is vanished under the optimality (Fig. 5). That is, the agent should avoid learning \( \pi(x_2|c_x) \) at the beginning, but optimize \( \pi(x_2|c_x) \) instead.

**Definition 4** (Non-Redundancy under Optimality (NRO)). Given \( \{G, Y, X^*, C^*\} \), an MPS \( S \) is said to be non-redundant under optimality if there exists an SCM \( M \) compatible with \( G \) such that \( \mu_\pi \neq \mu_{\pi'} \) for every strictly subsumed MPS \( S' \subseteq S \), i.e., \( \exists M \sim G, \forall S' \subseteq S(\mu_\pi(S') \neq \mu_{\pi'}(S')) \).

We will investigate a criterion more general than Thm. 1 — whether a set of contexts \( C' \subseteq C_X \) are relevant for a set of actions \( X' \subseteq X^* \) while taking account of deterministic relationships (Prop. 1).

**Proposition 3.** Given an MPS \( S \), let \( X' \subseteq X(S) \) and \( C' \subseteq C_X \) be actions and contexts of interest, and let \( Q' = P_{\pi}(X') \). A mixed policy \( \pi \sim S \) optimal with respect to \( S \), if there exist decision rules \( \{\pi'(x|c_x')\}_{x \in X'} \) such that \( \mu_\pi = \sum_{y,c',x} y Q_x(y,c') \prod_{X' \in X'} \pi'(x|c_x') \), then, \( C_X \setminus \{C\} \) are redundant to \( X' \) under optimality, and \( S' = (S \setminus X') \cup \{\langle X, C' \cap C_X \rangle \}_{X \in X'} \) satisfies \( \mu_{\pi'} = \mu_{\pi} \).

As a first step, we discuss the implication of deterministic relationships to the d-separation criterion. The graphical criterion handles deterministic mechanisms (i.e., conditional intervention) by excluding them appearing as common causes, e.g., \( \leftarrow X \rightarrow \), in a trail [37]. This corresponds to adding those **implies** variables to the conditionals, in which we explicitly represent with an operation \( \cdot \) for clarity. Given conditions \( Z \), the implied variables with respect to \( Z \) is computed as follows. Initially setting \( |Z| \leftarrow Z \), we update \( |Z| \leftarrow \{X \in X(S) \mid C_X \subseteq |Z|\} \) until it is converged. Then, given \( \pi \sim S \),

3This condition was leveraged in the atomic interventions case to establish minimality [16,17,38].

4The relevance of contextual information has been discussed in the influence diagrams literature [39,20]. More recently, this condition was used in the case of singleton decisions (i.e., \( |X(S)| = 1 \)), see [40,41].
We revisit Fig. 4 where we will show that, indeed, such that represents a maximal, non-redundant MPS under an optimal condition for Fig. 4.

**Theorem 2.**

Theorem 2 is a dependency graph specifying the (conditional) fixability of — whether each of its parent is either given as a remaining context or fixable (i.e., ) — with , literally speaking, checks whether the iterated application of sub converges to (with the implied operation). In other words, it examines whether the discarding contexts is fixable conditioned on the remaining contexts for each action X so that it can be optimally determined only by.

We revisit Fig. 4 where we will show that, indeed, and are redundant contexts under optimality. With and , Z becomes . With order , , the presented criteria and MPS, the aforementioned phenomena can be arbitrarily complex. We present a general criterion to test such redundancies. Given an order over V, a subset of V preceding V is denoted by . We denote by the minimal subset of T such that .

**Theorem 2.** Given an MPS , which satisfies non-redundancy (Thm. 1), let for a determinable policy optimal with respect to . Let , actions of interest, is contexts to examine its redundancy relative to . Let , , and . Let be an acyclic graph over such that has as its parents in if is without bidirected edges. If an order exists over , while being compatible with respectively over and , such that the following three conditions hold,

1. \((Y \perp \pi_X | [W])_{\mathcal{H}}\),
2. \((C \perp \pi_X | X < C | [W'_C])_{\mathcal{H}}\) for every , and
3. \(\text{sub}_{\mathcal{I}}(\pi_X') = C_X\) where if \(\{\}\) for every .

then, .

Given this, we revisit Fig. 4, where we will show that, indeed, and are redundant contexts under optimality. With and , Z becomes . With order , , the presented criteria and MPS, the aforementioned phenomena can be arbitrarily complex. We present a general criterion to test such redundancies. Given an order over V, a subset of V preceding V is denoted by . We denote by the minimal subset of T such that .

\[
\mu_S = \sum_{u,z} Q(u) \prod_{z \in X} Q(z | v, u) \sum_y Q(y | c_x') \prod_{x \in X} Q(x | c_x')
\]

\[
\text{to be fixed, then removed}
\]

\[J\] is a dependency graph specifying the (conditional) fixability of — whether each of its parent is either given as a remaining context or fixable (i.e., ) — with , literally speaking, checks whether the iterated application of sub converges to (with the implied operation). In other words, it examines whether the discarding contexts is fixable conditioned on the remaining contexts for each action X so that it can be optimally determined only by.

We revisit Fig. 4 where we will show that, indeed, and are redundant contexts under optimality. With and , Z becomes . With order , , the presented criteria and MPS, the aforementioned phenomena can be arbitrarily complex. We present a general criterion to test such redundancies. Given an order over V, a subset of V preceding V is denoted by . We denote by the minimal subset of T such that .

\[
\mu_S = \sum_{u,z} Q(u) \prod_{z \in X} Q(z | v, u) \sum_y Q(y | c_x') \prod_{x \in X} Q(x | c_x')
\]

\[\text{to be fixed, then removed}\]

\[J\] is a dependency graph specifying the (conditional) fixability of — whether each of its parent is either given as a remaining context or fixable (i.e., ) — with , literally speaking, checks whether the iterated application of sub converges to (with the implied operation). In other words, it examines whether the discarding contexts is fixable conditioned on the remaining contexts for each action X so that it can be optimally determined only by.

We revisit Fig. 4 where we will show that, indeed, and are redundant contexts under optimality. With and , Z becomes . With order , , the presented criteria and MPS, the aforementioned phenomena can be arbitrarily complex. We present a general criterion to test such redundancies. Given an order over V, a subset of V preceding V is denoted by . We denote by the minimal subset of T such that .

\[
\mu_S = \sum_{u,z} Q(u) \prod_{z \in X} Q(z | v, u) \sum_y Q(y | c_x') \prod_{x \in X} Q(x | c_x')
\]

\[\text{to be fixed, then removed}\]
the first two conditions (and additional rules) yield the following expression for the expected reward,

\[ \mu^*_S = \sum_{c_{23}, u} Q(u)Q(c_3)Q(c_2|u, c_3) \sum_{y, x, c_1} yQ_x(y, c_1)Q(x_1|c_1)Q(x_2|c_2). \]

Here, \( U \) (the only UC) and \( C_3 \) are marginally fixed to \( u^* \) and \( c_3^* \) (no need to be specified), and \( C_2 \) will be fixed given \( u \) and \( c_3 \), \( c_2^*(u^*, c_3^*) \). Then, there exists \( \pi' \) such that

\[ \mu^*_S \leq \sum_{y, x, c_1} yQ_x(y, c_1)Q(x_1|c_1)Q(x_2) = \sum_{y, x, c_1} yQ_x'(y, c_1)\pi'(x_1|c_1)\pi'(x_2). \]

Alternatively, the same result can be elicited as a two-step process: Removing \( C_3 \) from \( X_2 \) makes \( X_2 \) marginally fixable, invalidating a trail \( C_2 \rightarrow X_2 \rightarrow Y \). Then, \( C_1 \) becomes redundant for \( X_1 \), and the MPS is refined to \( \{ \langle X_1, \{ C_1 \} \rangle, \langle X_2, \emptyset \rangle \} \) (Fig. 5e) (Appendix E.1 for elaborate details).

4 A Partial Order over Mixed Policies and Possible-Optimality

Equipped with the notion of non-redundancy under optimality (NRO, Def. 3), an agent can more efficiently optimize its policy than relying on generic non-redundancy (Def. 3). Yet, an important question is whether an MPS is worth to explore for an agent to converge to an optimal policy. Consider for an instance, see Figs. 4a [4b] which represents various MPSes and they are NRO. However, even without interacting with an environment, we can claim \( \mu \leq \mu^*_S \leq \mu^*_S' \leq \mu^*_S'' \), that is, the next MPS is better than or equal to (simply better or improved hereinafter) the one regarding their optimal expected rewards in any model: First, \( \mu \leq \mu^*_S \) since there exists an optimal \( X_1 \) value, \( x_1^* \); Next, \( \mu^*_S \leq \mu^*_S' \), there exists an optimal \( X_2 \) value, and can be determined without conditional on \( X_1 \), which is implied; Finally, \( \mu^*_S' \leq \mu^*_S'' \) since \( X_1 \) can better behave taking \( C \) into account. Therefore, the agent can only optimize parameters involving \( S'' \) (Fig. 6a). Against this background, we characterize such a partial order over the space of MPSes with respect to their maximum expected rewards achievable: when one MPS is better than the other. To begin a formal discussion, we introduce possible-optimality of MPS.

**Definition 5** (Possibly-Optimal MPS). Given \( (G, X^*, C^*, Y) \), let \( S \) be a set of NRO MPSes. An MPS \( S \in S \) is said to be possibly-optimal if there exists \( M \sim G \) such that \( \mu_S > \max_{S' \in S} \mu^*_{S'} \).

In the partial order sense, POMPSes are the maximal elements in NRO MPSes. To study the partial order, we present two operations which take an MPS and return an improved MPS: (i) adding observations for existing actions and (ii) adding new interventions. These two operations offer sufficient conditions for non-POMPSes.

**Proposition 4.** Given an MPS \( S \) and \( X \in X(S) \), adding \( C \in C^* \setminus X \) as a context of \( X \), resulting \( S' = S \setminus \{ X \} \cup \{ \langle X, C, X \cup \{ C \} \rangle \} \) improves \( S \) if \( C \not\in \text{de}(X)_{G_S} \) and \( C \text{ } \perp \perp \text{ } Y \text{ } \mid \text{ } [X] \text{ } \text{ in } \mathcal{H}\setminus\{X\} \).

This proposition is straightforward. Note however that the resulting MPS may not be NRO as an added observation can cancel the relevance of the existing contexts, e.g., Prop. 2 can be viewed as adding observations and removing now irrelevant observations. Further, any set of observations that can be added to a set of actions to improve an MPS can also simply be added sequentially.

**Adding new interventions** Intervening replaces the natural mechanism for \( X \in X^* \) with an artificial one \( \pi_{X|Z} \). To guarantee that the alternative one can perform as good as the natural one, we should understand what information \( X \) originally takes and whether the new contexts \( Z \) carry information tantamount to the original one. If every parent of \( X \in X^* \) is contextualizable (i.e., no UC), the problem becomes trivial (e.g., Markovian). Otherwise, we examine the existence of a back-door path: Let \( Q = P_\pi \) and \( H = \mathcal{G}_\pi \) for some \( S^{-} \pi \). Given \( X \in X^* \setminus X(\pi) \) and \( Z \subset C^* \), if (i) \( Y \perp X \mid [Z] \) \( \mathcal{H}_\pi \) and (ii) \( X \not\in \text{An}(Z) \) \( \mathcal{H}_\pi \), then \( \mu_\pi = \sum_{y, x, z} yQ_x(y|z)\pi'(x|z) \leq \sum_{y, z} yQ_x(y|z)Q(x|z)\pi'(x|z) \leq \sum_{y, x, z} yQ_x(y|z)Q(x|z)\pi'(x|z) \leq \mu_\pi' \), for some \( \pi' \). Hence, \( \mu^*_S \leq \mu^*_S(\pi_{X|Z}) \text{ since } \pi' \text{ can be optimized. However, naively generalizing}

\[ 16, 17 \text{ studied 'possibly-optimal' atomic interventions } (C^* = \emptyset) \text{ where their conclusions can be essentially reduced to finding actions with no back-door path to } Y \text{ while varying the strengths of UCs.} \]
Given an MPS $S$, let $S' \neq S$ be an MPS with $X(S) \subseteq X(S')$ and $X' = (X(S') \setminus X(S)) \cup \{ X \in X(S) \mid C'_X \neq C_X \}$. Let $H''$ be the union of induced graphs, $G_S \cup G_{S'}$, where the unobserved confounders adjacent to $X'$ are made explicit and let $S'' = \{(X, \text{pa}(X)_{H''}) \mid X \in X'\}$. If $H''$ is acyclic and $\mu^*_{S''} = \mu^*_S$ can be elicited by Thm. 2, then, $\mu^*_{S} \leq \mu^*_{S''}$.

Given an MPS $S$, an intermediate MPS is constructed adding new contexts to a subset of $X^*$ while assuming that any non-contextualizable variables can be used as contexts. Fig. 7B depicts such an MPS to test $S' = \{(X_1, C_1), (X_2, C_2)\}$. Then, we can check the redundancy of contexts to remove non-contextualizable contexts (Thm. 2). As illustrated, Fig. 7C we can elicit $\mu \leq \mu^*_{S''}$ confirming that an observational policy is not a POMPS. By allowing $X^*$ to intersect with $X(S)$, the theorem not only adds new inventions but also can replace the contexts of existing interventions.

**Refining the space of MPSes**  Equipped with the characterizations, we can refine the space of MPSes, hence, the space of mixed policies, by filtering out MPSes that are either redundant or dominated by other MPS, eliciting a superset of POMPSes in a given setting. This can be achieved in a brute-force manner by enumerating all MPSes, and examining whether any of Thms. 2 and 3 and prop. 4 is applicable. One barrier to design a more principled approach (e.g., dynamic programming [16]) to obtaining POMPSes (or a superset of) is that contexts are interleaved in both terms in Eq. (1) representing a reward mechanism and a policy. Nevertheless, we investigate simplifying a mixed policy setting while preserving its POMPSes. First, one may think that the descendants of $Y$ can be ignored since neither intervening them changes the reward nor observing them is feasible. Surprisingly, Fig. 7C where $X_1$ and $C$ take place after the reward is evaluated, demonstrates the opposite. With $X_1$ intervened, $C$ can become a context for $X_2$ (Fig. 7C). This implies that contexts in the descendants of the reward becomes usable if interventions can break the relationships. Second, $X \in X^*$ that cannot affect $C^*$ or $Y$ is not intervene-worthy — if $de(X^*)_G \cap (C^* \cup \{Y\}) = \emptyset$, there exists no MPS that makes $X \in an(C^* \cup \{Y\})$, and, thus, $X$ can be excluded from $X^*$.

**Proposition 5.** Given $(\mathcal{G}, Y, X^*, C^*)$, let $X' = \{ X \in X^* \mid de(X)^*_G \cap (C^* \cup \{Y\}) \neq \emptyset \}$, $X'' = de(Y)'_{G_{X^*}} \cap X'$, and $Z = de(Y)'_{G_{X^*}}$. The POMPSes for $(\mathcal{G}, Y, X^*, C^*)$ are the same as those for $(\mathcal{G}_{X'' \setminus \{Y\}}, Y, X^*, C^* \setminus X'' \setminus \{Z\})$.

**5 Conclusions**

In this paper, we studied the space of mixed policies that emerges through the empowerment of an agent to determine the mode it will interact with the environment — i.e., which variables to intervene on and which contexts it decides to look into. Facing new challenges to optimize this new mode of interaction, which has many additional degrees of freedom, we studied the topological structure induced by the different mixed policies, which could in turn be leveraged to determine partial orders across the policy space w.r.t. the maximum expected rewards achievable. As a practical result, we provided a general characterization of the space of mixed policies with respect to properties that allow the agent to detect inefficient and suboptimal strategies. One of the surprising implications of this characterization provided here is that agents following a more standard approach (i.e., intervening on all intervenable variables and observing all available contexts) may be hurting themselves, and may never be able to achieve an optimal performance regardless of the number of interactions performed.
References


A Discussion – Introductory Example

We provide further elaboration on the example discussed in the introduction (Fig. 1a), which is shown again for convenience in Fig. 8. We recall that \( X_1 \) and \( X_2 \) are intervenable and \( X_1 \) and \( C \) are observable variables (i.e., can be used as context), so we write \( X^* = \{X_1, X_2\} \) and \( C^* = \{C, X_1\} \), following the corresponding notation. There are 15 distinct ways for an agent to interact with the system, which is explicitly shown in Fig. 9. This plot is known as a Hasse diagram (i.e., a diagram with transitive interaction). Given their indistinguishability in terms of achievable rewards, (i.e., \( \mu^* = \mu^*_\alpha \)) we usually call the nodes connected through these edges an equivalence class, given their indistinguishability in terms of achievable rewards. Regarding the subsumption relationship, a red directed edge \( \alpha \rightarrow \beta \) represents a subsumption relationship meaning that \( \beta \) has more actions or contexts than \( \alpha \), so is able to mimic it. The goal is usually to find policies that achieve higher rewards (relative to dimension (a)) and are more parsimonious, or simpler (relative to dimension (b)).

Regarding the dominance relation, a blue directed edge \( \alpha \rightarrow \beta \) corresponds to \( \mu^*_\alpha \leq \mu^*_\beta \) and a gray dotted undirected edge \( \alpha \sim \beta \) represents the equivalence in their maximum achievable expected rewards, (i.e., \( \mu^*_\alpha = \mu^*_\beta \)); we usually call the nodes connected through these edges an equivalence class, given their indistinguishability in terms of achievable rewards. Regarding the subsumption relation, a red directed edge \( \alpha \rightarrow \beta \) represents a subsumption relationship meaning that \( \beta \) has more actions or contexts than \( \alpha \), so is able to mimic it. The goal is usually to find policies that achieve higher rewards (relative to dimension (a)) and are more parsimonious, or simpler (relative to dimension (b)). For grounding the discussion, we start with dimension (a) and consider \( \mu \), the expected reward for the observational policy, and \( \mu^*_{\pi(x_1|c)} \), the maximum expected reward with the policy intervening on \( X_1 \) given \( C \). We will show below \( \mu \leq \mu^*_{\pi(x_1|c)} \) by using do-calculus,

\[
\mu = \sum_y y P(y) = \sum_{y,x_1,c} y P(y,x_1,c) P(x_1|c) P(c) = \sum_{y,x_1,c} y P_x(y|c) P_x(x_1|c) P_x(c) \leq \sum_{y,x_1,c} y P_x(x_1|c) P_x(c) = \mu^*_{\pi(x_1|c)},
\]

where the last equality comes from the expression for the expected reward (Eq. 1); the last inequality comes from the fact that \( \pi(x_1|c) \) can simply be the natural \( P(x_1|c) \), but also can be optimized to yield a higher expected reward. For further illustration of the dominance relation, we relate the optimized policies \( \pi(x_1|c) \pi(x_2|x_1) \) and \( \pi''(x_2|c) \) through the following derivation,

\[
\mu^*_{\pi(x_1|c)} \pi(x_2|x_1) = \sum_{y,x_1,c} y P_x(y,c) \pi(x_1|c) \pi(x_2|x_1) = \sum_{y,x_1,c} y P_{x_2}(y,c) \pi(x_1|c) \pi(x_2|x_1) \quad \text{by Eq. (1) (9)}
\]

Rule 3 of do-calculus (10)
with the analysis of dominance. Specifically, if a policy does not have an incoming red edge from the root, it is non-redundant. To better understand how the dominance and subsumption dimensions are related, we superimpose both relations in Fig. 10. The policies forming an equivalence class are clustered and highlighted within a gray rectangle, and marked its boundary with black for optimality. For each equivalence class, there exists one non-redundant policy, highlighted in yellow. For instance, in the aforementioned equivalence class (the

\[
\sum_{y, x_2, c} y P_{x_2}(y, c) \sum_{x_1} \pi(x_1 | c) \pi(x_2 | x_1) \quad \text{algebra (11)}
\]

There exists a probability mapping \( \pi' \) that can listen to \( C \) (while preserving the equality),

\[
= \sum_{y, x_2, c} y P_{x_2}(y, c) \sum_{x_1} \pi(x_1 | c) \pi'(x_2 | x_1, c)
\]

by construction (12)

\[
= \sum_c \sum_{x_1} \pi(x_1 | c) y P_{x_2}(y, c) \pi'(x_2 | x_1, c)
\]

algebra (13)

There exists a value \( x_1 \) for each \( c \) that can maximize the expression inside \( \sum_c \),

\[
= \sum_c \sum_{x_1} \pi(x_1 (c) | c) y P_{x_2}(y, c) \pi'(x_2 | x_1 (c), c)
\]

by definition (14)

\[
\leq \sum_{y, x_2, c} y P_{x_2}(y, c) \pi'(x_2 | x_1 (c), c)
\]

algebra (15)

Since \( x_1 (c) \) is determined by \( c \), there exists \( \pi'' \) such that

\[
= \sum_{y, x_2, c} y P_{x_2}(y, c) \pi''(x_2 | c)
\]

by construction (16)

\[
= \mu_{\pi''(x_2 | c)}
\]

by Eq. 11 (17)

Therefore, \( \mu_{\pi(x_1 | c) \pi(x_2 | x_1)} \leq \mu_{\pi''(x_2 | c)} \). It is not immediately obvious how we can formally derive such inequalities between two optimal expected rewards for arbitrary environments. Throughout the paper, we build graphical and algorithmic criteria that tell whether one policy can dominate another.

After all, \( \pi(x_1 | c) \) (top-right in Fig. 9a) dominates its neighbors and can attain optimality. Sometimes, a set of policies forming an equivalence class can achieve optimality, i.e.,

\[
\{ \pi(x_2 | c), \pi(x_1 | c) \pi(x_2 | c), \pi(x_1 | c) \pi(x_2 | c, x_1), \pi(x_1 | c) \pi(x_2 | c, x_1), \pi(x_1 | c) \pi(x_2 | c, x_1) \}
\]

Now, we turn our attention to the subsumption relation as shown in Fig. 9b. We first note that the construction of this diagram is based on the scope of the given policy (Def. 1), as defined in the paper, namely, the set of actions (before conditioning bar) and the corresponding context (after the conditioning bar). The construction is graph-insensitive, but will play a key role when combined with the analysis of dominance. Specifically, if a policy does not have an incoming red edge from other policy in its equivalence class, the policy is non-redundant. To better understand how the dominance and subsumption dimensions are related, we superimpose both relations in Fig. 10. The policies forming an equivalence class are clustered and highlighted within a gray rectangle, and marked its boundary with black for optimality. For each equivalence class, there exists one non-redundant policy, highlighted in yellow.
We empirically validate that the use of refined policies leads to a better performance measured. In particular, we use the cumulative regret (the lower the better), i.e., $\sum_{t=1}^T Y_t$, where $T$ is the number of time steps (i.e., interactions) and $Y_t$ is a random variable for the reward at time $t$. Further, we demonstrate that the CB approach cannot achieve the optimal reward in a certain environment, incurring a linear cumulative regret. The basic experimental setup is, for each time step, the agent plays only the 7 non-redundant policies. Knowing that 5 out of the 7 policies are redundant, we exemplified next an environment (structural causal model) compatible with the example discussed above, which will validate the right most equivalence class in Fig. 10, $\pi(x_2|c)$ is subsumed by other policies while maintaining the same optimal reward. In fact, this implies that $\pi(x_2|c)$ will be preferred over its counterparts in the equivalence class given its capability of achieving the optimality while being the most parsimonious within its class. Comparing whether one policy subsumes the other outside the equivalence class makes less sense since they are not comparable. For example, $\pi(x_2|c)$ subsumes $\pi(x_2)$ and $\emptyset$ through the red arrows, but belongs to a different equivalence class, so they are non-comparable, one is not preferred over the other. In this example, we can see through Fig. 10 that there are 7 non-redundant policies (yellow): $\emptyset$ (observational policy), $\pi(x_1)$, $\pi(x_1|c)$, $\pi(x_2)$, $\pi(x_2|x_1)$, $\pi(x_1|c)\pi(x_2|x_1)$, and $\pi(x_2|c)$. Putting this information together, we can see in Fig. 10 that only $\pi(x_1|c)$ and $\pi(x_2|c)$ satisfy both optimality (black boundaries) and non-redundancy (yellow), which are then marked in green.

Once the intelligent agent has access to causal information (e.g., in the form of the causal graph), it can explore the underlying environment with policies that can achieve optimal reward efficiently. We now describe different approaches the agent can take. A standard approach would be taking all policies are experimented. A more efficient route would be to avoid redundant policies, where the agent plays only the 7 non-redundant policies. Knowing that 5 out of the 7 policies are no better than the other 2, the most efficient approach would be to assess those, i.e., $\pi(x_1|c)$ and $\pi(x_2|c)$. We name these approaches as CB, BF, NRO, and POMPS, where the exact meaning of NRO (Non-Redundant under Optimality) and POMPS (Possibly-Optimal Mixed Policy Scope) will become clearer through out the paper.

We empirically validate that the use of refined policies leads to a better performance measured. In particular, we use the cumulative regret (the lower the better), i.e., $T \mu^{*} - \sum_{t=1}^T Y_t$, where $T$ is the number of time steps (i.e., interactions) and $Y_t$ is a random variable for the reward at time $t$. Further, we demonstrate that the CB approach cannot achieve the optimal reward in a certain environment, incurring a linear cumulative regret. The basic experimental setup is, for each time step, the agent assesses policies using samples from posterior reward distributions (i.e., Thompson sampling) based on its interaction history, and executes the chosen policy. We exemplified next an environment (structural causal model) compatible with the example discussed above, which will validate the

---

6Here, we mean a policy by fully-specified decision rules. For instance, there are four discrete policies corresponding to a strategy $\pi(x_1|c)$ with binary $C$ and $X_1$. 

A-3
Figure 11: Performance comparison for different approaches (CB, BF, NRO, POMPS). (a,b) Each line represents cumulative regrets averaged over 100 repetitions (the lower the better), and its shade represents standard deviation. (c,d) Probability the agent selects the best policies (with CB 0% and the lines are smoothed with moving average). The figures in the left side highlight the first 2,000 time steps and ones in the right side the whole 10,000 steps.

Rewards, and is unknown by the agent:

\[
\mathcal{M} = \begin{cases} 
C \leftarrow U_C \oplus U_1 \\
X_1 \leftarrow U_1 \oplus U_{X_1} \\
X_2 \leftarrow X_1 \oplus C \oplus U_2 \oplus U_{X_2} \\
Y \leftarrow X_2 \oplus C \oplus U_2 \oplus U_Y 
\end{cases}
\]

where the unobserved confounders \( U_1 \) (between \( C \) and \( X_1 \)) and \( U_2 \) (between \( X_2 \) and \( Y \)) are fair coins and each of \( U \in \{U_{X_1}, U_{X_2}, U_C, U_Y\} \) is binary and follows \( P(U = 1) = 0.1 \).

The corresponding simulation is shown in Fig. 11 reporting two types of plots based on (a,b) cumulative regrets and (c,d) the probability selecting the optimal policy. It is evident from the specification that the CB agent cannot optimize its policy to achieve the optimality, regardless of the number of interactions with the environment. This is demonstrated as a linear cumulative regret (Figs. 11a and 11b) and 0% probability selecting an optimal policy in Figs. 11c and 11d. The BF approach is almost equally inefficient up to around 2000 steps, but is still able to find the optimal since it includes the possible optimal policies, POMPSes. As expected, the performance improves with the use of smaller number of policies. After all, the CB approach does not guarantee the optimality, while BF, NRO, POMPS are always guaranteed to converge. Further, the use of only non-redundant policies by NRO and POMPS helps the agent to converge to the optimal policy faster.

B Preliminaries

We use in the paper classic causal inference results such as do-calculus, which we summarize here.

D-separation We start with the definition of d-separation \([42]\), without a particular consideration of deterministic relationships.

**Definition 6** (d-separation). Two sets of vertices \( X, Y \) are said to be d-separated by another set \( Z \) in a directed acyclic graph \( \mathcal{G} \), denoted by \( (X \perp \perp Y \mid Z)_{\mathcal{G}} \), if every path \( P \) from vertices in \( X \) to vertices in \( Y \) are blocked where blockage occurs when one of the following holds:
1. \( P \) contains at least one arrow-emitting node that is in \( Z \), or
2. \( P \) contains at least one collider that is outside \( Z \) and has no descendant in \( Z \).

**Do-calculus**  Do-calculus \([32]\) is an essential machinery to reason about the equivalence of conditional interventional probabilities induced by any model conforming to a given causal graph. Do-calculus consists of three rules where each rule ascertains that an equality between two probability distributions holds if a certain graphical test (separation) holds. The three rules are

- **R1 (Insertion/deletion of observations):** \( P_x(y|z, w) = P_x(y|w) \) if \( Z \perp Y \mid X, W \) in \( G_X \)
- **R2 (Action/observation exchange):** \( P_x(y|z, w) = P_{x,z}(y|w) \) if \( Z \perp Y \mid X, W \) in \( G_{XZ} \)
- **R3 (Insertion/deletion of actions):** \( P_x(y|w) = P_{x,y}(y|w) \) if \( Z \perp Y \mid X, W \) in \( G_{X/Z(W)} \)

where \( Z(W) \) is a subset of \( Z \) that is not an ancestor of \( W \) in \( G_X \). For convenience of some of the proofs, let \( \mathcal{H} = G \setminus X \) and \( Q = P_x \), and consider the following (rewritten) what rules:

- **R1 (Insertion/deletion of observations):** \( Q(y|z, w) = Q(y|w) \) if \( Z \perp Y \mid W \) in \( \mathcal{H} \)
- **R2 (Action/observation exchange):** \( Q(y|z, w) = Q_{x}(y|w) \) if \( Z \perp Y \mid W \) in \( \mathcal{H}_{Z} \)
- **R3 (Insertion/deletion of actions):** \( Q(y|w) = Q_{x}(y|w) \) if \( Z \perp Y \mid W \) in \( \mathcal{H}_{Z(W)} \)

This representation is motivated by C-factors and will help us to highlight the differences between distributions while abstracting away unnecessary details.

### C Mixed Policies

**Derivation of Expected Reward**  A derivation for Eq. (1) is shown below with abbreviations: MP, the definition of marginal probability; CR, the chain rule; R# : rule # of do-calculus; and def: by definition. Let \( \prec \) be a topological order compatible with \( G_x \), and let \( X_{\prec C} = \{ X \in X \mid X \prec C \} \). We may write capital subscripts of a value as lowercase, e.g., \( c_x \) instead of \( C_x \) for the value for \( C_X \). With \( X = X(\pi) \), \( C = C(\pi) \), and \( C^- = C \setminus X \), we start by writing the average reward giving by \( \pi \),

\[
\mu_{\pi} \triangleq \mathbb{E}_\pi [Y] \quad \text{def} \quad (18)
\]

\[
= \sum_{y, x, c^-} y P_x(y, x, c^-) \quad \text{MP} \quad (19)
\]

\[
= \sum_{y, x, c^-} y P_x(y|x, c^-) P_x(x, c^-) \quad \text{CR} \quad (20)
\]

\[
= \sum_{y, x, c^-} y P(y|x, c^-) P_x(x, c^-) \quad \text{R1} \quad (21)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) P_x(x, c^-) \quad \text{R2} \quad (22)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) \prod_{x \in X} P_{x}(x_{\prec x}, c_{\prec x}) \prod_{c \in C^{-}} P_{c}(c|c_{\prec C}) \quad \text{CR} \quad (23)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) \prod_{x \in X} P_{x}(x_{c_x}) \prod_{c \in C^{-}} P_{c}(c|c_{\prec C}) \quad \text{R1} \quad (24)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) \prod_{x \in X} \pi(x|c_x) \prod_{c \in C^{-}} P_{c}(c|c_{\prec C}) \quad \text{def} \quad (25)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) \prod_{x \in X} \pi(x|c_x) \prod_{c \in C^{-}} P_{x}(c|c_{\prec C}) \quad \text{R1} \quad (26)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) \prod_{x \in X} \pi(x|c_x) \prod_{c \in C^{-}} P_{x}(c|c_{\prec C}) \quad \text{R2} \quad (27)
\]

\[
= \sum_{y, x, c^-} y P_x(y|c^-) \prod_{x \in X} \pi(x|c_x) \prod_{c \in C^{-}} P_{x}(c|c_{\prec C}) \quad \text{R3} \quad (28)
\]

A-5
Algorithm 1 Separation of actions and contexts of an MPS

1: function sep-mps(S, G)
   input: a mixed policy scope S, a causal graph G
   output: an updated, action-context separated mixed policy scope S
2: for X ∈ topological-order(S(S); G_S) do
3:   Replace C_X in S by ((U_{X’εC_X∩X’} C_{X’}) ∪ (C_X \ X^*).)
4: return S

\[
= \sum_{y, x, c^-} yP_x(y|c^-) \prod_{x \in X} \pi(x|c_x)P_x(c^-) \quad \text{CR} \quad (29)
\]
\[
= \sum_{y, x, c^-} yP_x(y, c^-) \prod_{x \in X} \pi(x|c_x), \quad \text{CR} \quad (30)
\]

Note that Rule 1 of do-calculus applied to the regime nodes (Eq. (21)) is identical to Rule 2 applied to X in Eq. (22). The derivation for a subset of X and its contexts follows directly by treating uninteresting decision rules as natural mechanisms.

Note on Multi-Agent Systems Although the treatment given to mixed policies is framed with respect to a single agent, its implications to a multi-agent setting is apparent – each action variable can be considered as an agent where the absence of directed edges among them exhibits their autonomy. Further, from the multi-agent point of view, the current definition of mixed policy assumes that each agent has the same ability to sense contextual variables C'. More realistic multi-agent settings will allow for different sensing capabilities for agents. The results presented in this paper can be effortlessly generalized to this case, where each agent (or action) is associated with its own set of contextualizable variables. Another almost immediate extension is for multi-reward settings, e.g., where one attempts to optimize Y_1 and Y_2. Depending on the task, one may focus on a specific reward, or one can create a new aggregate reward Y = Y_1 + Y_2 to perform a task over the setting.

D Optimality and Deterministic Mixed Policy

Proposition 1. Given a mixed policy scope, there always exists a deterministic mixed policy, which is optimal with respect to the given scope.

Proof. Consider an arbitrary optimal policy \( \pi \sim S \) given an MPS S. Let X = X(\pi) and C^- = C(\pi) \setminus X. Given a topological order among X defined over G_S such that X_i \prec X_j if i < j, let Q' = P_{\pi \setminus (X_1)} where \( \pi \setminus X' \) denotes a policy \( \pi \) with actions X' \subseteq X removed. Then,

\[
\mu_\pi = \sum_{y, x_1, c_1} yQ_{x_1}^\prime(y, c_1)\pi(x_1|c_1) = \sum_{x_1, c_1} \pi(x_1|c_1) \sum_y yQ_{x_1}^\prime(y, c_1).
\]

If \( \pi_{X_1|C_1} \) is not deterministic with respect to c_1 where \( P_\pi(c_1) = Q'(c_1) > 0 \), there must be at least two values of x_1' and x_1'' such that

\[
\pi(x_1'|c_1) \sum_y yQ_{x_1}^\prime(y, c_1) = \pi(x_1''|c_1) \sum_y yQ_{x_1}^\prime(y, c_1).
\]

Otherwise, if one value is larger than the other, this contradicts the optimality since \( \pi_{X_1|C_1} \) can select the value that yields a larger value than the other. In case of \( Q'(c_1) = 0 \), the choice of x_1 becomes irrelevant. Hence, we can modify the strategy on a single action to be deterministic for a specific context. This argument can be sequentially applied to the rest of intervened variables. As a result, one can elicit a deterministic optimal mixed policy from a given optimal mixed policy. Therefore, there exists a deterministic mixed policy, which is optimal with respect to the given MPS. \( \square \)

Proposition 2 (Separation of Actions and Contexts). Given an MPS S, there always exists a deterministic mixed policy \( \pi \in \Pi \) such that X(\pi) and C(\pi) are disjoint and \( \mu_S = \mu_\pi \).
Proof. Let a mixed policy \( \rho \sim S \) be optimal with respect to \( S \). First, there exists an optimal deterministic mixed-policy \( \rho' \) equivalent to \( \rho \) with respect to the expected reward (Prop. 1). Since the graph \( G_{\rho'} \) is acyclic, there exists a topological order among \( X \). Consider \( X \in X \) such that \( C_{X'} \cap X = \emptyset \) for every \( X' \in X \cap X \). We can create a new function \( \pi_X \) based on \( \rho_X \) and \( \rho_{X'} \):

\[
\pi_X(\{x'\} \cup \{x\}) = \rho_X(\{x'\}) = c_{x'}, X' = \rho_{X'}(c_{x'}).)
\]

This can be iteratively applied following the topological order among \( X \) to obtain a new deterministic policy \( \pi \) such that \( X(\pi) \) and \( C(\pi) \) are disjoint without changing the expected reward (see Alg. 3).

### E Non-Redundant Mixed Policy

**Theorem 1.** Let \( S = \{(X, C_X)\}_{X \in X} \) be an MPS and let \( H = G_S \). \( S \) is non-redundant if and only if

(i) \( X \subseteq an(Y) \) and (ii) \((C \perp Y | C_X \{C\})\) in \( H \{X\} \), for every \( X \in X \) and \( C \in C_X \).

**Proof.** (Only if) (i) Let \( X \in X \\backslash an(Y) \) and \( Q' = P_{\pi'} \{X\} \) and \( H' = G_{\pi'} \{X\} \). First, \( X \in X \\backslash an(Y) \) since (i) for every \( X \in X \\backslash an(Y) \) not changing the descendants of \( X \). Then, \( Q'_X(y|c_x) = Q'(y|c_x) \) since \((X \perp Y | C_X)\) in \( H' \{c_x\} \) (i.e., Rule 3 of do-calculus). Further, \( Q'_X(c_x) = Q'(c_x) \) since, again, Rule 3 that \( X \perp C_X \) in \( H'_X \) as no \( C_X \) is a descendant of \( X \) and nothing is given. Then,

\[
\mu_{\pi} = \sum_{y, x, c_x} yQ'_X(y|c_x)\pi(x|c_x) \quad \text{def}
\]

\[
= \sum_{y, x, c_x} yQ'_X(y|c_x)Q'_X(c_x)\pi(x|c_x) \quad \text{CR}
\]

\[
= \sum_{y, x, c_x} yQ'(y|c_x)Q'(c_x)\pi(x|c_x) \quad \text{violation of (i)}
\]

\[
= \sum_{y, x, c_x} yQ'(y, c_x)\pi(x|c_x) \quad \text{CR}
\]

\[
= \sum_{y, c_x} yQ'(y, c_x)\sum_{x} \pi(x|c_x) \quad \text{algebra}
\]

\[
= \sum_{y} yQ'(y) \quad \text{MP}
\]

\[
= \mu_{\pi - \{X\}} \quad \text{def}.
\]

(ii) Let \( Q = P_{\pi} \) and \( Q' = P_{\pi'} \{X\} \) for some \( \pi \sim S \) and \( C_X = C_X \{C\} \) where \( C \in C_X \) which violates (ii). Let \( H' = G_{\pi'} \{X\} \). Note that the test in \( H \{X\} \) is identical to the test in \( H' \{X\} \) as the only differences in \( H \) and \( H' \) are the parents of \( X \). Then,

\[
\mu_{\pi} = \sum_{y, x, c_x} yQ'_X(y|c_x)Q'_X(c_x)\pi(x|c_x) \quad \text{def, CR}
\]

\[
= \sum_{y, x, c_x} yQ'_X(y|c_x)Q'_X(c_x)\pi(x|c_x) \quad \text{violation of (ii)}
\]

\[
= \sum_{y, x, c_x} yQ'_X(y|c_x)Q'_X(c_x)Q'_X(c_x)\pi(x|c_x) \quad \text{CR}
\]

\[
= \sum_{y, x, c_x} yQ'_X(y, c_x)\sum_{c} Q'_X(c|c_x)\pi(x|c_x) \quad \text{CR, algebra}
\]

\[
= \sum_{y, x, c_x} yQ'_X(y, c_x) \sum_{c} Q_x(c|c_x)\pi(x|c_x) \quad \text{R1}
\]

\[
= \sum_{y, x, c_x} yQ'_X(y, c_x) \sum_{c} P_{\pi}(c|c_x)\pi(x|c_x) \quad \text{R3}
\]
We show, for an arbitrary MPS (e.g., \(Z\)) that are connected via bidirected paths with \(\{C_1, C_2, C_3\}\) having directed paths onto them. (d) bidirected paths between a subset of group \(\{C_1, C_2, C_3, C_4\}\) are shared; directed paths from \(C_7\) and \(C_6\) to \(C_4\) are shared; the bidirected path between \(C_1\) and \(Y\) intersects bidirected paths between \(C_1\) and \(\{C_2, C_3, C_4\}\); finally, the directed path from \(X\) to \(Y\) is also shared.

\[
\pi''(x|c_x) = \sum_{y, x, c_x} yQ_y(x, c_x) \pi''(x|c_x) = \mu_{\pi''},
\]

where \(\pi''(x|c_x) \equiv \sum_c P_{\pi}(c|x) \pi(x|c_x, c).\) Since \(\pi\) properly subsumes \(\pi''\), \(\pi\) is redundant.

Theorem 4. Given an MPS \(S\), \(\text{nr-mps}\) returns a unique, maximal non-redundant MPS of \(S\).

Proof. The algorithm refines a given mixed policy scope (MPS) \(S\) by iterating over \(X(S)\) and \(C(S)\). Let the graph be under examination (Line 4) and canceled out (i.e., bit parity) except the one involved between the group and \(Y\). But any \(\pi'\), whose \(X\) does not listen to \(C_X\), makes its expected reward 0.5. 

We now construct an SCM demonstrating non-redundancy. As illustrated in Fig. [12(d)], above mentioned paths can intersect with each other. We consider each bidirected and directed path maintains its own ‘channel’ where the path in \(H\) can be understood as a cable of multiple bits where non-end variables pass bits to the downstream. This principle is also applied to different groups since the paths connecting each group to \(Y\) can be shared. Let every parentless variable (including UCs and \(C_X'(\{C_X'\})\) behaves as a fair coin or a vector of independent fair coins if it involves in multiple paths (e.g., \(Z\) in Fig. [12(d)] has 6 bits for every pair among \(\{C_1, C_2, C_3, C_4\}\)). We design the function for every \(C \in C'_X\) and \(X\) to be the bit-parity of its parents (i.e., all channels incoming to \(C\)) and the mechanism for \(Y\) is similarly designed except that it takes its complement. Information of every fair coin is counted twice at \(X\) and canceled out (i.e., bit parity) except the one involved between the group and \(Y\) (e.g., one between \(C_1\) and \(X\) in every example in Fig. [12] which will be canceled out at \(Y\). Then, the expected reward for \(\pi\) becomes 1.0 as every bit-parity is counted twice and complemented at \(Y\). But any \(\pi'\), whose \(X\) does not listen to \(C_X\) as a whole, makes its expected reward 0.5. 

Proof. The algorithm refines a given mixed policy scope (MPS) \(S\) by iterating over \(X(S)\) and \(C(S)\). Let the graph be under examination (Line 4) and canceled out (i.e., bit parity) except the one involved between the group and \(Y\). But any \(\pi'\), whose \(X\) does not listen to \(C_X\) as a whole, makes its expected reward 0.5. 

We now construct an SCM demonstrating non-redundancy. As illustrated in Fig. [12(d)], above mentioned paths can intersect with each other. We consider each bidirected and directed path maintains its own ‘channel’ where the path in \(H\) can be understood as a cable of multiple bits where non-end variables pass bits to the downstream. This principle is also applied to different groups since the paths connecting each group to \(Y\) can be shared. Let every parentless variable (including UCs and \(C_X'(\{C_X'\})\) behaves as a fair coin or a vector of independent fair coins if it involves in multiple paths (e.g., \(Z\) in Fig. [12(d)] has 6 bits for every pair among \(\{C_1, C_2, C_3, C_4\}\)). We design the function for every \(C \in C'_X\) and \(X\) to be the bit-parity of its parents (i.e., all channels incoming to \(C\)) and the mechanism for \(Y\) is similarly designed except that it takes its complement. Information of every fair coin is counted twice at \(X\) and canceled out (i.e., bit parity) except the one involved between the group and \(Y\) (e.g., one between \(C_1\) and \(X\) in every example in Fig. [12] which will be canceled out at \(Y\). Then, the expected reward for \(\pi\) becomes 1.0 as every bit-parity is counted twice and complemented at \(Y\). But any \(\pi'\), whose \(X\) does not listen to \(C_X\) as a whole, makes its expected reward 0.5. 

Proof. The algorithm refines a given mixed policy scope (MPS) \(S\) by iterating over \(X(S)\) and \(C(S)\). Let the graph be under examination (Line 4) and canceled out (i.e., bit parity) except the one involved between the group and \(Y\) (e.g., one between \(C_1\) and \(X\) in every example in Fig. [12] which will be canceled out at \(Y\). Then, the expected reward for \(\pi\) becomes 1.0 as every bit-parity is counted twice and complemented at \(Y\). But any \(\pi'\), whose \(X\) does not listen to \(C_X\) as a whole, makes its expected reward 0.5.
Algorithm 2 Non-Redundant Mixed Policy Scope $S$

1: function NR-MPS($G$, $Y$, $S$)
   input: a mixed policy scope $S$, a causal graph $G$
   output: an updated, non-redundant mixed policy scope $S$
2: $S \leftarrow S \setminus (X \setminus \text{an}(Y) \cup \text{gS})$.
3: for $X \in \text{reverse-order}(X(S); G_S)$ do
4:   if $X \notin \text{an}(Y) \cup \text{gS}$ then
5:     $S \leftarrow S \setminus \{X\}$ and continue.
6:   for $C \in X_C$ do
7:     if $(C \perp Y | X_C \setminus \{C\})$ in $G_S \setminus \{X\}$ then
8:       $S \leftarrow (S \setminus \{X \}) \cup \{(X, C_X \setminus \{C\})\}$.
9: return $S$.

tests are irrelevant as they are all non-successors of $X$ while the ancestrality is only relevant to its successors.

Next consider examining $C \in X_C$ for some $X$ where now $S'$ is the MPS at Line 7. Consider a trail (d-connection path) $\rho$ between $C$ and $Y$ in $G_{S'}$. We restrict our attention to a collider-minimal, shortest path. Every collider $W$ in the path has a directed path towards $Y$ through $W \rightsquigarrow C_i \rightsquigarrow X \rightsquigarrow Y$ where $C_i \in X_C$ by the testing criteria and $W \rightsquigarrow C_i$ can be of zero length. Let $G_{\rho'} \subseteq G_{S'}$, be the path graph together with directed paths between colliders and conditionals. Let $S''$ be the MPS at the end of testing every $C_i \in X_C$. We show that $(C \perp Y | C_X')$ in $G_{S''} \setminus \{X\}$ holds true (with $C_X'$ as in $S''$). Let $C_i$ be the subset of $X_C$ that associates with the colliders in the path. Since every $C_i \in C_i$ will have a back-door path to $Y$ (by concatenating a directed path between $C_i$ and $W$ and the subpath of $\rho$ between $W$ and $Y$) given $C_i \setminus \{C\}$, $C_i$ is the subset of $C_X'$. Hence, the result follows (for an illustrative example, please see Fig. 13).

Now we investigate whether the path $\rho$ between $C$ and $Y$ is still valid in $G_{S'}$ given $C_X$. Specifically, we would like to ensure that the edges in the path are intact throughout the changes made by the algorithm. The removal of $X' \in X(S)_{\not\perp \! \! \perp}$ may affect the parents of $X'$. The removal of $C' \in X_C$ affects $C' \rightarrow X'$. That is, for both cases, we investigate whether $C' \rightarrow X'$ in $\rho$ is intact at the end of the algorithm. We first state two claims.

Claim 1. $X'$ has a directed path to $Y$ in $G_{\rho}$.

Claim 2. $C' \perp Y$ in $G_{\rho} \setminus \{X'\}$ demonstrates the existence of a collider-free d-connection $\phi$, which is disjoint with $C_X \setminus \{C'\}$.

The directed path between $X'$ and $Y$ will be valid in $G_{S'}$ if every $C'' \rightarrow X''$ appeared in the path is intact, which delegates its validity to the bottom-most $X'' \in X(S)_{\not\perp \! \! \perp}$. $C' \in X_{\not\perp \! \! \perp}$ will be dependent on $Y$ given $C_X \setminus \{C''\}$ as no $C_X \setminus \{C''\}$ exists in $G_{\rho}$. Hence, by tracing back the validity of each policy-induced edge in $G_{\rho}$, we conclude that every policy-induced edge is intact, and, thus, $G_{\rho} \subseteq G_{S'}$ and $C' \in C_X^{\perp \! \! \perp}$.

Now we prove the two claims in the above proof.

Claim 1. $X'$ has a directed path to $Y$ in $G_{\rho}$.

Proof. Let $\bullet$ be an unspecified edge mark representing either arrow or tail and a squiggly edge represents a path. An abstract representation for the path $\rho$ can be one of the following two forms $C \bullet \rightarrow X' \bullet \rightarrow Y$ or $C \bullet \rightarrow X' \leftarrow C' \bullet \rightarrow Y$ with $\times$ represents that the path may have colliders. $X'$ has a directed path to (i) $C$, (ii) $Y$, or (iii) some $C_a \in C_X \setminus \{C\}$ via a collider $W$ (which can be $X'$ itself) in the path. Then, a directed path can be of the form: (i) $X' \leftarrow C \rightarrow X \leftarrow Y$, (ii) $X' \leftarrow Y$, or (iii) $X' \leftarrow W \leftarrow C_a \rightarrow X \leftarrow Y$.

Claim 2. $C' \perp Y$ in $G_{\rho} \setminus \{X'\}$ demonstrates the existence of a collider-free d-connection $\phi$, which is disjoint with $C_X \setminus \{C'\}$.

A-9
Proof. Similar to the proof for the directed path between $X'$ and $Y$, our abstract representation informs us that we can consider two subpaths (a) $C \xrightarrow{\neq} C'$ or (b) $C' \xrightarrow{\neq} Y$ where both avoids passing through $X'$.

For (a), if the subpath does not contain a collider, (a1) $C' \xrightarrow{\neq} C \xrightarrow{} X \xrightarrow{} Y$ is a valid trail signaling $C' \not\perp Y$ in $G_\rho \{X'\}$. Otherwise, there exists a collider $W$ in the subpath, which uses a shortcut to $Y$ through $C_a \in C_X \{X'\}$: (a2) $C' \xrightarrow{} W \xrightarrow{} C_a \xrightarrow{} X \xrightarrow{} Y$.

In case of (a1), $C'' \in C_X \{C'\}$ cannot appear in between $C'$ and $C$ since otherwise it violates the fact that $\rho$ being the shortest—one can create $\rho'$ by replacing $C \ldots C'' \ldots C' \xrightarrow{} X$ by $C \ldots C'' \xrightarrow{} X$. In case of (a2), similarly, $C''$ cannot reside along $C' \xrightarrow{} W$. In addition, we prove that $C''$ does not exist in-between $W \xrightarrow{} C_a$ since otherwise there exists $\rho'$ which does not require $W$ in the path $\rho$ as a collider making use of $C'' \xrightarrow{} X'$, contradicting the collider-minimality of $\rho$. In either cases (a1) and (a2), $C \xrightarrow{} X \xrightarrow{} Y$ or $C_a \xrightarrow{} X \xrightarrow{} Y$ are $C''$-free otherwise it contradicts the topological order between $X'$ and $X$.

In case of (b), either there exists (b1) a path towards $Y$, $C' \xrightarrow{} Y$ without a collider, or (b2) through a collider as seen in the case of (a2), where the same proof is applicable. Given (b1), again, if $C''$ exists in the path, we can shorten $\rho$ by connecting $X' \leftarrow C''$ while replacing $X' \leftarrow C' \xrightarrow{} C''$, which violates $\rho$ being the shortest.

Hence, the result follows.

E.1 Non-Redundancy under Optimality

Derivations for the Redundancy of Examples We demonstrate the redundancy for Figs. 5b to 5d.

We may employ “≤” to highlight that fixing operation can improve the expected reward although,
given the optimality of the left hand side, it becomes ‘=’. We present a derivation for Fig. 5b showing that $C_3$ is non-informative.

$$
\mu_\pi = \sum_{y, x, c} yQ(y|x, c)Q(x, c) \quad \text{(MP, CR)} \quad (31)
$$

$$
= \sum_{y, x, c} yQ(y|x, c_1, c_2)Q(x, c) \quad \text{(R1)} \quad (32)
$$

$$
= \sum_{y, x, c} yQ'_x(y|c_1, c_2)Q(x, c) \quad \text{(R1, R2)} \quad (33)
$$

$$
= \sum_{y, x, c} yQ'_x(y|c_1, c_2)Q(x_1|c_3, c_1)Q(x_2|c_3, c_2)Q(c) \quad \text{(CR, R1)} \quad (34)
$$

$$
= \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)Q(c_1, c_2)\sum_{c_3} Q(x_1|c_3, c_1)Q(x_2|c_3, c_2)Q(c_3) \quad \text{(MP, CR)} \quad (35)
$$

$$
= \sum_{c_3} Q(c_3) \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)Q(c_1, c_2)Q(x_1|c_3, c_1)Q(x_2|c_3, c_2) \quad \text{algebra} \quad (36)
$$

$$
\leq \sum_{c_3} Q(c_3) \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)Q(c_1, c_2)Q(x_1|c_3, c_1)Q(x_2|c_3, c_2) \quad \text{def} \quad (37)
$$

$$
= \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)Q(c_1, c_2)Q(x_1|c'_3, c_1)Q(x_2|c'_3, c_2) \quad \text{algebra} \quad (38)
$$

$$
\leq \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)Q(c_1, c_2)\pi'(x_1|c_1)\pi'(x_2|c_2) \quad \text{def} \quad (39)
$$

$$
= \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)Q'_x(c_1, c_2)\pi'(x_1|c_1)\pi'(x_2|c_2) \quad \text{(R1, R3)} \quad (40)
$$

$$
= \sum_{y, x, c, c_1, c_2} yQ'_x(y|c_1, c_2)\pi'(x_1|c_1)\pi'(x_2|c_2) \quad \text{CR} \quad (41)
$$

where $c'_3 \in X_{C_3}$ is the value maximizing the inner sum.

The derivation for Fig. 5c is given that $C_2$ is non-informative.

$$
\mu_\pi = \sum_{y, x, c} yQ(y|x, c)Q(x, c) \quad \text{def} \quad (42)
$$

$$
= \sum_{y, x, c} yQ(y|x, c_1)Q(x, c) \quad \text{(R1)} \quad (43)
$$

$$
= \sum_{y, x, c} yQ'_x(y|c_1)Q(x, c) \quad \text{(R1, R2)} \quad (44)
$$

$$
= \sum_{y, x, c} yQ'_x(y|c_1)Q(c_1)Q(c_2|c_1)Q(x_1|c)Q(x_2|c) \quad \text{(CR)} \quad (45)
$$

$$
= \sum_{y, x, c} yQ'_x(y|c_1)Q'_x(c_1)Q(c_2|c_1)Q(x_1|c)Q(x_2|c) \quad \text{(R1, R2)} \quad (46)
$$

$$
= \sum_{y, x, c} yQ'_x(y, c_1)Q(c_2|c_1)Q(x_1|c)Q(x_2|c) \quad \text{CR} \quad (47)
$$

$$
= \sum_{y, x, c, c_1} yQ'_x(y, c_1)\sum_{c_2} Q(c_2|c_1)Q(x_1|c)Q(x_2|c) \quad \text{algebra} \quad (48)
$$

$$
\leq \sum_{c_3} \sum_{y, x, c, c_1} Q(c_2|c_1)Q(x_1|c)Q(x_2|c) \quad \text{algebra} \quad (49)
$$

$$
\leq \sum_{c_3} \sum_{y, x, c, c_1} Q(c_2|c_1)Q(x_1|c_1, c'_2(c_1))Q(x_2|c_1, c'_2(c_1)) \quad \text{def} \quad (50)
$$

$$
= \sum_{c_3} \sum_{y, x, c, c_1} yQ'_x(y, c_1)Q(x_1|c_1, c'_2(c_1))Q(x_2|c_1, c'_2(c_1)) \quad \text{algebra} \quad (51)
$$

A-11
= \sum_{y,x,c_1} yQ'_x(y,c_1)\pi'(x_1|c_1)\pi'(x_2|c_1)

The derivation for Fig. 5d is as follows. Let \( Q' = P_{\pi \setminus \{X_1,X_2\}} \).

\[
\mu_{\pi} = \sum_{y,x_2,c_2} yQ(y|x_{12},c_2)Q(x_{12},c_2) \quad \text{MP,CR (53)}
\]

\[
= \sum_{y,x_{12},c_2} yQ'_x(y,c_2)Q(x_{12},c_2) \quad \text{R1,R2 (54)}
\]

\[
= \sum_{y,x_{12},c_2} yQ'_x(y,c_2)\sum_{x_3} Q(x,c_2) \quad \text{MP (55)}
\]

\[
= \sum_{y,x_{12},c_2} yQ'_x(y,c_2)\sum_{x_3} Q(x_3|c_2)Q(x_1|x_3,c_2)Q(x_2|x_3,c_2) \quad \text{CR (56)}
\]

\[
= \sum_{y,x_{12},c_2} yQ'_x(y,c_2)\sum_{x_3} Q(x_3|c_2)Q(x_1,x_3,c_2)Q(x_2|x_3,c_2) \quad \text{algebra,R3 (57)}
\]

\[
= \sum_{y,x_{12},c_2} \sum_{x_2} Q(x_3|c_2) yQ'_x(y,c_2)Q(x_1|x_3,c_2)Q(x_2|x_3,c_2) \quad \text{algebra (59)}
\]

\[
\leq \sum_{x_2} \sum_{y,x_{12},c_2} Q(x_3|c_2) yQ'_x(y,c_2)Q(x_1|x_3^*(c_2),c_2)Q(x_2|x_3^*(c_2),c_2) \quad \text{def (60)}
\]

\[
= \sum_{y,x_{12},c_2} yQ'_x(y,c_2)Q(x_1|x_3^*(c_2),c_2)Q(x_2|x_3^*(c_2),c_2) \quad \text{algebra (61)}
\]

\[
= \sum_{y,x_{12},c_2} yQ'_x(y,c_2)\pi'(x_1|c_2)\pi'(x_2|c_2) \quad \text{def (62)}
\]

**Proofs and Additional Characterizations**

**Theorem 2.** Given an MPS \( S \), which satisfies non-redundancy (Thm. [7]), let \( \pi \sim S \) be a deterministic policy optimal with respect to \( S \). Let \( X' \subseteq X(S) \), actions of interest, \( C' \subseteq C_X \setminus X' \), non-action contexts to retain, where \( Z = C_X \setminus X' \cap C' \) is contexts to examine its redundancy relative to \( X' \). Let \( W' = X' \cup C' \), \( \mathcal{H} = \mathcal{G}_{Z}(\{Y\} \cup X' \cup C_X) \), and \( C'_X = C_X \cap W' \). Let \( \mathcal{T} \) be an acyclic graph over \( W' \cup Z \) such that \( Z = Z_1 W_1 Z_{<Z} H_{<Z} Z \) as its parents in \( \mathcal{T} \) where \( H' = \mathcal{H} \) without bidirected edges. If an order \( \prec \) exists over \( W' \cup Z = X' \cup C_X \), while being compatible with \( H \) respectively over \( W' \) and \( X' \cup Z \), such that the following three conditions hold,

1. \( \{Y \perp \pi_{X'} | [W']\}_{H'} \),
2. \( \{C \perp \pi_{X' \setminus C} | [W'_C]\}_{H'} \) for every \( C \in C' \), and
3. \( \{\text{sub}_< C'_X \} = C_X \) where \( \text{sub}(T) = T \cup \{V \in \mathcal{T} \mid \text{pa}(V) \subseteq \mathcal{T}\} \) for every \( X \in X' \),

then \( S' = (S,X') \cup \{\{X,C_X \cap W'\}\}_{X \in X'} \) satisfies \( \mu_S = \mu_S' \).

**Proof.** Let \( Q = P_{\pi} \) and \( Q' = P_{\pi_{X'}}. \) Given the order \( \prec \), the following equalities hold:

\[
Q(x|W'_C \cup Z_{<x}) = Q(x|W'_C \cup Z_{<x} \cap C_{x'}, \cup Z_{<x} \cap C_x) = Q(x|c'_{x'}, z_{<x} \cup C_x \setminus c_x') \quad \text{given to infer/remove}
\]

Without loss of generality, let \( U \) be unobserved confounders in \( H \). Let \( C# \) be the condition #.

\[
\mu_{\pi} = \sum_{y,c',x'} yQ(y|c',x')Q(c',x') \quad \text{MP,CR (63)}
\]

\[
= \sum_{y,c',x'} yQ'_x(y|c')Q(c',x') \quad \text{C1 (65)}
\]
Figure 14: A more involved example for redundancies in a mixed policy scope

\[
\begin{align*}
&= \sum_{y',c'} y Q'_x(y|c') \sum_z Q(c', z, x') & \text{MP (66)} \\
&= \sum_{y',c'} y Q'_x(y|c') \sum_{z} \prod_{c' \in c'} Q(c|(w', z)_{<c}) \prod_{v \in X' \cup Z} Q(v|(w', z)_{<v}) & \text{CR (67)} \\
&= \sum_{y',c'} y Q'_x(y|c') \prod_{c' \in c'} Q(c'| c'_{<c}) \sum_z \prod_{v \in X' \cup Z} Q(v|(w', z)_{<v}) & \text{C2 (68)} \\
&= \sum_{y',c'} y Q'_x(y|c') \prod_{c' \in c'} Q(c'| c'_{<c}) \sum_z \prod_{v \in X' \cup Z} Q(v|(w', z)_{<v}) & \text{R3 (69)} \\
&= \sum_{y',c'} y Q'_x(y|c') \sum_{z} \prod_{u \in u} Q(z|(w', z)_{<z}) \prod_{x \in X'} Q(x|(w', z)_{<z}, u) & \text{MP (71)} \\
&= \sum_{y',c'} y Q'_x(y|c') \sum_{z} \prod_{u \in u} Q(z|v_z, u) \prod_{x \in X'} Q(x|c'_x, c_x|c'_x) & \text{R1 (72)} \\
&= \sum_{c',x'} \sum_{z} \prod_{x' \in x'} Q(z|v_z, u) \sum_{y} \prod_{x \in X'} Q(x|c'_x, c_x|c'_x) & \text{algebra (73)} \\
&\leq \sum_{c',x',u'} \sum_{z} \prod_{x' \in x'} Q(z|v_z^{(s)}, u^*) \sum_{y} \prod_{x \in X'} Q(x|c'_x, c_x|c'_x) & \text{def (74)} \\
&= \sum_{c',x'} \sum_{x' \in x'} Q(x|c'_x, c_x|c'_x) & \text{algebra (75)} \\
&= \sum_{c',x'} \sum_{x' \in x'} \pi'(x|c'_x) & \text{def (76)}
\end{align*}
\]

where \(v_z^{(s)} = (z^* \cap V_z) \cup (v_z \setminus Z)\). Here, neither \(U\) nor its value \(u^*\) needs to be specified but the mere existence suffices for the derivation. 

An example Fig. [14] is given accompanied with a derivation below. Consider an order of \(C_1, C_4, C_2, C_5, X_1, X_3, C_3, C_6, X_2, X_4\). Let \(U_{25}, U_{36}\) be the two UCs between \(C_2\) and \(C_5\) and \(C_3\) and \(C_6\).

\[
\begin{align*}
\mu_{\pi} &= \sum_{y,x,c_{14}} y Q(y|x, c_{14}) Q(x, c_{14}) \\
&= \sum_{y,x,c_{14}} y Q'_x(y|c_{14}) Q(x, c_{14}) \\
&= \sum_{y,x,c_{14}} y Q'_x(y|c_{14}) \sum_{c_{2356}} Q(c_{2356}, x, c_{14}) \\
&= \sum_{y,x,c_{14}} y Q'_x(y|c_{14}) \sum_{c_{2356}} Q(c_{14}) Q(c_{2}|c_{1}) Q(c_{5}|c_{124}) Q(x|c_{12})
\end{align*}
\]
Proof. Let UCs between $X$ and $C_1$ and $C_4$ be $U_1$ and $U_4$ where each one is two-bit fair coins. Let $X$ be binary variables. Let $C_1$ and $C_4$ copy $U_1$ and $U_4$, and $Y$ take one minus the bit parity of four bits of $X$ and the four bits of $\{U_1, U_4\}$. Hence, only when $X_1$, $X_2$, $X_3$, and $X_4$ pass the matching information, the expected reward becomes 1. Otherwise, the expected reward falls down to 0.5.

Even when $C_2, C_3, C_5, C_6$ are all confounded in Fig. 15, we can similarly elicit the same result as UCs affecting those removables are marginally fixable.

Proposition 6. The first condition for Thm. 2 can be rewritten as $(Y \perp \pi_X^Y | [W]^\ast)_H = (Y \perp \pi_X^Y, Z_{\downarrow Y} | [W]^\ast)_H$ with $Y$ being the last in the order $\prec$.

Proof. By the first expression where $X^\prime \subseteq [W]^\ast$, there exists no trail from $\pi_X \in \pi_X^Y$ to $Y$. The trail we sought in the first independence must pass through $C_X \setminus C_{\downarrow Y}$ (i.e., some $Z \in Z$) since both (i) the only child of $\pi_X$, $X$, is blocked and (ii) the parents of $X$ other than $Z$ are blocked. If $Z$ and $Y$ are conditionally dependent in the second expression, then by augmenting $Z \rightarrow X \leftarrow \pi_X$, we can show that $\pi_X$ is also dependent to $Y$ in the first independence, which is contradictory. Hence, the equality between the first two independence statements holds. By the assumption for Thm. 2 requiring Thm. 1 all actions and contexts are ancestors of $Y$. Hence, $X^\prime \prec Y$, $\pi_X \prec Y$, $Z \prec Y$, and $W^\prime \prec Y$. 

Please see Fig. 15 how values can be properly fixed. Then,

$$\sum_{y, x, c_{14}} y Q^2(y, c_{14}) \pi'_{\{x_1 | c_{14}\}} \pi'_{\{x_2 | c_{14}\}} \pi'_{\{x_3 | c_{14}\}} \pi'_{\{x_4 | c_{14}\}}$$

Remark 1. $S' = \{X_1, \{C_1\}, X_2, \{C_1\}, X_3, \{C_4\}, X_4, \{C_4\}\}$ is non-redundant under optimality.
Algorithm 3 A polynomial-time algorithm for an admissible order for Thm. 2

1: function admissible-order(G, S, X', C', Z)
2: Initialize a directed graph G’ with G(S) ⊂ X ∪ X’ without UCs.
3: repeat
4: Let ≺∼ G’, an arbitrary topological order from G’.
5: for C ∈ C’ in the order ≺, for X ∈ X’ do
6: if (π X ⊥⊥ C X’ ⊥⊥ C) in G then G’ ← G’ ∪ {C → X}; break
7: return none if G’ is cyclic.
8: until G’ is not changed
9: Add to G’ the directed edges in G(S) (X’ ∪ Z) and the latest ≺ as a directed path.
10: G’ ← G’ ∪ {C → Z | (π X ⊥⊥ C X’ ⊥⊥ C) in G(S), C ∈ C’, Z ∈ Z’} where ≺∼ G’.
11: return ≺∼ G’ if G’ is acyclic else none.

Corollary 1. The first and second conditions for Thm. 2 can be concisely written as

\[ (V \perp \pi_{X’,V}, Z_{\prec V} \mid [W_{\prec V}])_{\mathcal{H}} \text{ for every } V \in C’ \cup \{Y\} \]

Proposition 7. admissible-order returns an order over W’ ∪ Z if one exists that satisfies the condition 2 of Thm. 2.

Proof. First, no X ∈ X’ should be an ancestor of C ∈ C’ in H since, otherwise, the condition 2 is failed as the order should be topological with respect to W’ in H as specified by Thm. 2. Hence, we focus on the case where no X is an ancestor of C ∈ C’. Since there is no descendant of C in W’ <C, any conditional dependence between C and X ∈ X’ <C can be explained by a d-connection path going up and down repeatedly where no C X ⊂ C’ < W’ <C blocks the path. Further, a path that goes up and down more than once implies that other C’ or X’ in X’ <C opens the colliders in the path. In other words, either C and X’ or C’ and X are conditionally dependent with a shorter path. Hence, any order that satisfies condition 2 needs to address a d-connection path goes up and down once only once between C and X. Given an order, if C and X are dependent via such an up-down trail, the order is invalid because only to make them independent is avoiding X being before C in the order, which we will show.

While keeping X before C in an order, consider changing the conditionals for the test by reordering some of the other contexts or actions after C forward. If some of other action X’ is placed in the path to block, this will be an ancestor of C in H, which violates the aforementioned condition. If they are in the down part of the path, it must not be after C, which contradicts the order X’ ≺ C. Similarly, if C’ is located in the up part of the path, C’ is dependent to X, again, failing to meet condition 2. Also C’ cannot be in the down part of the path for the same reason. Hence, only resolution for X being dependent to C through a single up-down path is placing X after C. The algorithm checks in order examining the existence of a single up-down path, and imposes such constraint, addressing independence between π_{X’,C} and C’ ∈ C’.

For Z ∈ Z <C and C ∈ C’, if Z is descendant of C in H, then there is no way to block (no conditional W’ <C is a descendant of C). Simply Z should not be before C in ≺. Then, we can adopt the same argument that other Z, C’, X’ cannot be moved forward to block the path. Hence, the task becomes resolving the order for each pair of C and Z. If there exists any order without conflict, the order satisfies the condition 2. \hfill \Box

F A Partial Order over Mixed Policies and Possible-Optimality

Theorem 3. Given an MPS S, let S’ ≠ S be an MPS with X(S) ⊂ X(S’) and X’ = \(X(S') \setminus X(S)\) \cup \{X ∈ X(S) | C_X ⊄ C_X\}. Let \(\mathcal{H}'\) be the union of induced graphs, G_S ∪ G_{S’}, where the unobserved confounders adjacent to X’ are made explicit and let S” = \{(X, pa(X)_{\mathcal{H}'})\}_{X \in X’}. If \(\mathcal{H}'\) is acyclic and \(\mu_{S’} = \mu_{S’}\) can be elicited by Thm. 2 then, \(\mu_S ≤ \mu_{S’}\).

Proof. The superimposition of the two induced graphs which resulting in an directed acyclic graph corresponds to an MPS S’ induced graph, that is at least as good as S and S’ with respect to their optimal expected rewards. By construction of \(\mathcal{H}’\), G_X ⊆ pa(X)_{\mathcal{H}’} and C_X ⊆ pa(X)_{\mathcal{H}’} for A-15
We first justify removing incoming edges onto \( X \). Then, \( X \) and \( C' \) match to the arguments for Thm. 2 with \( \mathcal{H}' \) and \( \mathcal{S}' \) as \( \mathcal{H} \) and \( \mathcal{S} \). Therefore, the result follows from \( \mu_{\mathcal{S}} \leq \mu_{\mathcal{S}'} = \mu_{\mathcal{S}'} \).

**Proposition 5.** Given \( \langle \mathcal{G}, Y, X^*, C^* \rangle \), let \( X' = X \cup \{ Y \} \cup X \cap (C^* \cup \{ Y \}) = \emptyset \). \( \mathcal{G}' \) is the graph obtained by removing all edges of \( X \) to \( Y \) and \( C^* \cup \{ Y \} \) from \( \mathcal{G} \). Then, \( \mathcal{G}' \) is the POMPS for \( \langle \mathcal{G}, Y, X', C' \rangle \). The POMPS for \( \langle \mathcal{G}, Y, X^*, C^* \rangle \) are the same as those for \( \langle \mathcal{G}_{\bar{X}''} \backslash (V \backslash Z), Y, X', C^* \backslash X'' \backslash Z \rangle \).

**Proof.** Consider a POMPS \( \mathcal{S} \) for \( \langle \mathcal{G}, Y, X^*, C^* \rangle \). Let \( \mathcal{H} = \mathcal{G}_{\bar{S}} \). By definition, \( \mathcal{X}(\mathcal{S}) \subseteq \text{an}(Y)_{\mathcal{H}} \) and, hence, \( C \subseteq \mathcal{C}_X \subseteq \text{an}(Y)_{\mathcal{H}} \) for every \( X \in \mathcal{X}(\mathcal{S}) \).

First, we justify the reduction of \( X^* \). If \( X \in X^* \) has no \( C^* \) or \( Y \) as descendants (inclusive) in \( \mathcal{G} \), it is impossible for \( X \) to become the ancestor of \( Y \) as none of its descendant (including itself) can become a context for \( Y \) in \( \mathcal{G} \). Therefore, \( X^* \) can be first refined to those among \( \mathcal{X}(\mathcal{S}) \cap \mathcal{C}(\mathcal{S}) \). Now consider the case where \( X \) is also contextualizable, i.e., \( X \in X^* \cap C^* \). If \( X \in \mathcal{C}(\mathcal{S}) \), that is, it is used as a context for another actions, then it won't be an action due to the separation (Prop. 2). Thus, we further exclude actionable variables that is not an ancestor of contextualizable variables or \( Y \). Hence, \( X' = \{ X \in X^* \mid \text{de}(X)_{\mathcal{G}} \cap (C^* \cup \{ Y \}) = \emptyset \} \) is the subset of \( X^* \) that can become an action in POMPS.

Next, we explain the reduction of \( C^* \) focusing on removing \( Z \). Any \( C \subseteq C^* \) that is a descendant of \( Y \) without any \( X^* \) present in the directed path from \( Y \) to \( C \) cannot become \( \text{an}(Y)_{\mathcal{H}} \). Hence, \( C^* \) are not contextualizable. With \( X'' = \text{de}(Y)_{\mathcal{G}_{\bar{X}''}} \cap \mathcal{X}' \), we can elicit

\[
\text{de}(Y)_{\mathcal{G}_{\bar{X}''}} \cap C^* = \text{de}(Y)_{\mathcal{G}_{\bar{X}''}} \cap C^* = \text{de}(Y)_{\mathcal{G}_{\bar{X}''}} \cap C^*.
\]

Hence, \( C^* \) can be reduced to \( C^* \backslash Z \) where \( Z = \text{de}(Y)_{\mathcal{G}_{\bar{X}''}} \).

Finally, we examine simplifying the given graph together with excluding \( X'' \) from contextualizables. We first justify removing incoming edges onto \( X'' \). Consider an action \( X \in \mathcal{X}(\mathcal{S}) \) that is a descendant of \( Y \) in \( \mathcal{G} \). Then, \( X \) must be either in \( X'' \) or the descendants of \( X'' \) (excluding \( X \) itself). Since there is no directed path from \( Y \) to \( X \) in \( \mathcal{H} \), every directed path between \( Y \) and \( X \) in \( \mathcal{G} \) is invalid in \( \mathcal{H} \) because some of \( X' \in \mathcal{X}(\mathcal{S}) \) breaks the ancestrality. If \( X' \in X'' \), then done as none of its parents will be used as a context. Otherwise, \( X' \) will be a descendant of \( X'' \), which makes committing \( X'' \) as redundant actions irrelevant to the reward (rule 3 of do-calculus). For either cases, treating \( X'' \) as pre-selected interventions that can be either the part of \( \mathcal{X}(\mathcal{S}) \) or redundant, \( X'' \) cannot become contexts per (Prop. 2). Hence, it justifies both the removal of the incoming edges onto \( X'' \) in \( \mathcal{G} \) and removing \( X'' \) from contextualizables as well. Consequently, the given setting is reduced to \( \langle \mathcal{G}_{\bar{X}''} \backslash (V \backslash Z), Y, X', C^* \backslash X'' \backslash Z \rangle \) projecting out \( Z \), which is neither an action, context, nor reward, from \( \mathcal{G}_{\bar{X}''} \).