Split the Differences, Pool the Rest: Provably Efficient Multi-Objective Imitation

Ziyad Sheebaelhamd[*]
University of Tübingen
ziyad.sheebaelhamd@uni-tuebingen.de

Luca Viano[*†]
ETH AI Center
luca.viano@ai.ethz.ch

Volkan Cevher
EPFL
volkan.cevher@epfl.ch

Claire Vernade
University of Technology Nuremberg
claire.vernade@utn.de

Abstract

This work investigates multi-objective imitation learning: the problem of recovering policies that lie on the Pareto front given demonstrations from multiple Pareto-optimal experts in a Multi-Objective Markov Decision Process (MOMDP). Standard imitation approaches are ill-equipped for this regime, as naively aggregating conflicting expert trajectories can result in dominated policies. To address this, we introduce Multi-Output Augmented Behavioral Cloning (MA-BC), an algorithm that systematically partitions divergent expert data while pooling state-action pairs where no behavior conflict is observed. Theoretically, we prove that MA-BC converges to Pareto-optimal policies at a faster statistical rate than any learner that considers each expert dataset independently. Furthermore, we establish a novel lower bound for multi-objective imitation learning, demonstrating that MA-BC is minimax optimal. Finally, we empirically validate our algorithm across diverse discrete environments and, guided by our theoretical insights, extend and evaluate MA-BC on a continuous Linear Quadratic Regulator (LQR) control task.

1 Introduction

Real-world datasets for complex sequential decision-making tasks, such as those used for LLM fine-tuning [44, 60, 8, 65] and robotic learning [33, 18], are rarely generated by a single, monolithic expert. Instead, they are typically sourced from diverse human or algorithmic operators, each acting under a distinct, often implicit, utility function over the underlying objectives. This inherent variation in operator preferences naturally yields highly multimodal datasets. The dominant paradigm addresses this by pooling these diverse trajectories into a single dataset and training expressive, multimodal models in an attempt to capture the aggregate variance of the constituent operators [32, 33, 57, 7, 23, 47]. However, the foundational theory of imitation learning in these conflicting, multi-objective settings remains largely underexplored.

In this work, we adopt an explicit treatment of this inherent multi-objectivity. We investigate the problem of imitation learning from multiple diverse experts, where each expert optimizes a distinct utility over the underlying vector rewards of a shared sequential decision task. We formalize this setting as a Multi-Objective Markov Decision Process (MOMDP) [59, 14, 39] and assume that the expert policies lie on the Pareto front.

A common practice, for example in LLM pretraining [36, 9] and robotics [19, 23, 7, 47], is to naively pool expert data to train a single joint model. This approach presents a challenge: we formally prove that such aggregation can yield dominated policies that fall entirely off the Pareto frontier (Theorem 2.2). As an example, consider an autonomous driving dataset collected from two experts:

[*]Equal contribution.
[†]Work done while at EPFL.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

one prioritizing speed, the other safety. A jointly trained model will learn a compromised policy that drives fast in some states and cautiously in others. Such a policy ultimately fails to optimize any coherent, monotonically increasing utility function of the returns and results in consistently suboptimal behavior.

An alternative approach, which has appeared in some form in [20, 21], is to reduce the problem to independent, single-expert imitation learning and train a separate model for each expert dataset. While this guarantees the recovered policies lie on the Pareto front, it entirely ignores the underlying structure of the MOMDP. Because the experts are constrained to the same Pareto frontier, their optimal policies inherently share substantial state-action overlap. Treating them as independent tasks discards valuable cross-dataset information, leading to severe sample inefficiency. Moreover, the only existing approach [11] that interpolates between merging all the data and splitting it still suffers from at least one of the two issues above. This raises the question:

Can data from multiple experts be used to efficiently learn a policy on the Pareto front?

We answer this question positively for discrete MOMDPs and introduce Multi-Output Augmented Behavioral Cloning (MA-BC), a Multi-Objective Imitation Learning (MOIL) algorithm designed to guarantee Pareto optimality while exploiting the shared structure on the frontier to achieve fast statistical rates. At its core, MA-BC operates on a simple yet powerful principle: by isolating the state-action pairs where experts disagree and aggregating the data where no conflict was observed, MA-BC efficiently reconstructs the optimal frontier policies that generated the dataset. We further propose a novel lower bound for Multi-Objective Imitation Learning, and show that MA-BC is minimax optimal. Motivated by our theory in the discrete domains, we propose an extension of MA-BC to continuous state and action spaces.

Contributions and outline In Section 2, we formalize the Multi-Objective Imitation Learning (MOIL) problem and contribute a structural result of independent interest: the "*Pareto Path Guarantee*", which proves that Pareto-optimal policies remain connected via single-state action changes. Building on this foundation, Section 3 introduces Multi-Output Augmented Behavioral Cloning (MA-BC), a novel algorithm that systematically partitions divergent expert data to maintain Pareto optimality. We provide theoretical guarantees proving that MA-BC converges to the Pareto front at a faster statistical rate than independent learners, and we establish a novel lower bound demonstrating its minimax optimality. Next, Section 4 extends our framework to accommodate continuous state-action spaces. Finally, Section 5 empirically validates MA-BC across diverse discrete environments and demonstrates its efficacy on a continuous 6-DOF quadcopter control task. For a broader discussion of the related literature on imitation learning, we refer the reader to Appendix A.

2 Preliminaries

In this section we establish the background for MOMDPs, focusing on the structural properties of Pareto front policies, and formalize the multi-objective imitation learning problem.

2.1 Multi-Objective Markov Decision Processes (MOMDPs)

To understand why naive imitation learning fails in multi-objective settings and why selective data partitioning succeeds, we must first establish the structural and geometric properties of MOMDPs.

Setup We define an MOMDP as a tuple $\mathcal{M} = (\mathcal{X}, \mathcal{A}, P, \gamma, \nu_0, r)$, where $\mathcal{X}$ and $\mathcal{A}$ are the discrete state and action spaces, P is the transition dynamics, $\gamma \in [0, 1)$ is the discount factor, and ν_0 is the initial state distribution [59, 14]. As in the single objective setting, we consider controlling the system with a stationary Markov policy $\pi : \mathcal{X} \to \Delta_{\mathcal{A}}$ which is a mapping from states to distributions over the action space. We denote the set of all stationary Markov policies as Π. Crucially, the reward function $r : \mathcal{X} \times \mathcal{A} \to [0, 1]^d$ outputs a d-dimensional vector representing potentially conflicting objectives. The expected vector-valued return of a stationary policy π, denoted as $J(\pi) \in \mathbb{R}^d$, is given by $J(\pi) = \mathbb{E}_\pi[\sum_{t=0}^{\infty} \gamma^t r(X_t, A_t)]$ where the infinite length sequence $(X_t, A_t)_{t=0}^{\infty}$ is generated by sampling $X_0 \sim \nu_0$, and then for any $t \geq 0$, $A_t \sim \pi(\cdot|X_t)$ and $X_{t+1} \sim P(\cdot|X_t, A_t)$. Moreover, we define the set of possible expected vector returns as $\mathbb{J} = \{J(\pi) : \pi : \mathcal{X} \to \Delta_{\mathcal{A}}\}$. An important concept for our analysis is the occupancy measure induced by policy π defined as

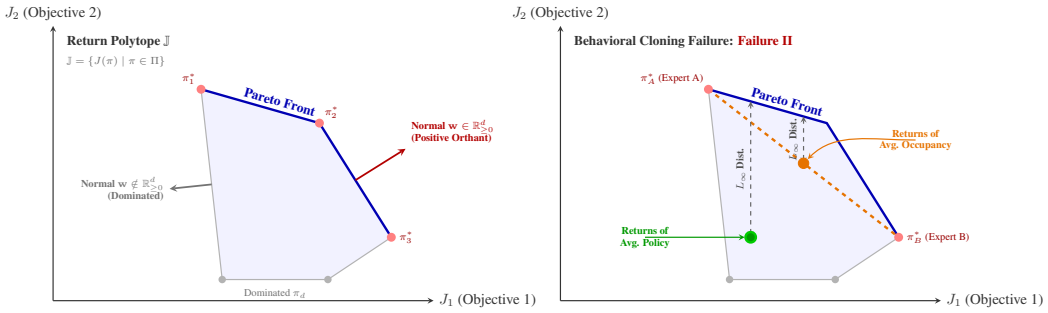


Figure 1: **(Left) Geometry of the MOMDP Return Space.** The space of expected returns $\mathbb{J}$ is a convex polytope [26]. The Pareto front (bold blue line) consists of boundary faces whose outward normal vectors $\mathbf{w}$ lie entirely within the positive orthant $\mathbb{R}^d_{\geq 0}$. Boundary faces with outward normal vectors containing negative components represent Pareto-dominated policies. All vertex policies are deterministic. **(Right) Naive Behavioral Cloning Failure.** Applying standard BC to a mixed dataset from multiple Pareto-optimal experts (π_A^*, π_B^*) converges to either the average occupancy measure (orange dot) or the average policy (green dot), depending on whether the objective is optimized at the trajectory or state level, respectively. Due to potentially nonlinear environment dynamics, the expected return of the average policy can "sag" significantly into the interior of the polytope. Nonetheless, both outcomes maintain a constant suboptimality gap as visualized by their L_∞ distance to the Pareto front (dashed black lines), highlighting **Failure II**.

$\mu^\pi(x,a) = (1-\gamma)\mathbb{E}_\pi\left[\sum_{t=0}^\infty \gamma^t \mathbb{1}_{\{X_t=x, A_t=a\}}\right]$ and the state occupancy measure defined as the marginal $\nu^\pi(x) = \sum_{a\in\mathcal{A}} \mu^\pi(x,a)$.

Solution concept In general MOMDPs, there rarely exists a single policy $\pi^\star$ such that $\forall i \in [d]$, $J_i(\pi^\star) = \max_{\pi\in\Pi} J_i(\pi)$. For this reason, we focus on learning *Pareto front policies* which are guaranteed to exist in any MOMDP. In particular, we define an approximate Pareto front policy as follows.

Definition 2.1. ***ε-approximate Pareto front policy (ε-PF-policy)*** *Let $\pi^\star$ be the output of a randomized algorithm. Then, $\pi^\star$ is an ε-PF-policy if*

$$\nexists \pi \in \Pi \text{ such that } (\forall i \in [d],\ J_i(\pi) - \mathbb{E}[J_i(\pi^\star)] \geq \varepsilon) \text{ and } (\exists j \in [d],\ J_j(\pi) - \mathbb{E}[J_j(\pi^\star)] > \varepsilon),$$

where the expectation is with respect to the algorithm's randomization. In words, a policy is an ε-PF-policy if there does not exist another policy in Π that improves all entries of $\mathbb{E}[J(\pi^\star)]$ by at least ε, and improves at least one entry by strictly more than ε. Moreover, if $\pi^\star$ is not random and is an ε-PF-policy with $\varepsilon = 0$, we call it a Pareto front policy, abbreviated as PF-Policy. In any MOMDP, the set of PF-policies is guaranteed to be non-empty [14, 59].

Structural properties In recent work [26], it has been shown that the return space achieved by the returns of stochastic stationary policies forms a convex polytope, where the vertices are deterministic policies. This implies that the Pareto front lies on the boundary of the polytope, and corresponds to edges and vertices that have a normal vector in the positive orthant. This can be visualized in Figure 1. Moreover, we define neighboring PF-policies as follows.

Definition 2.2 (Neighboring PF-policies)**.** *Two PF-policies π_A and π_B are said to be* ***neighboring*** *if and only if there exists a weight vector $w \in \mathbb{R}^d_+$ such that the face of optimal returns, i.e. $\arg\max_{J\in\mathbb{J}} w^T J$ coincides exactly with the line segment connecting $J(\pi_A)$ and $J(\pi_B)$.*

Next, we introduce a key structural result that motivates the design of our algorithm in Section 3: We show that any pair of neighboring PF-policies differing in k states can be bridged by a sequence of k intermediate PF-policies, each differing from the previous one in the action at a single state.

Theorem 2.1. *[Pareto Path Existence] Let π_A and π_B be neighboring PF-policies that differ in k states. Then, there exists a finite sequence of $k+1$ deterministic PF-policies $(\pi_0, \pi_1, \ldots, \pi_k)$ such that $\pi_0 = \pi_A$ and $\pi_k = \pi_B$ and consecutive policies in the sequence differ in exactly one state, i.e. $\sum_{x\in\mathcal{X}} \mathbb{1}\{\pi_{i-1}(x) \neq \pi_i(x)\} = 1$ for all $i \in \{1, \ldots, k\}$.*

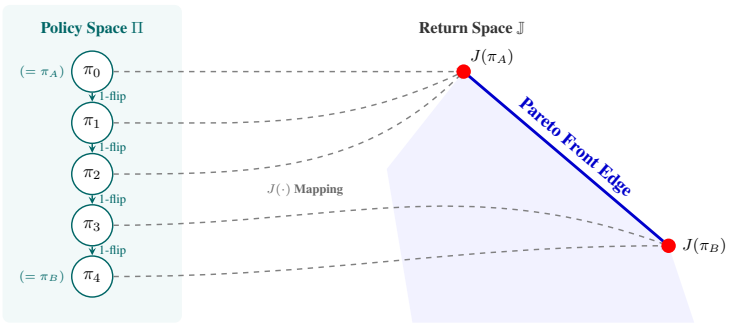


Figure 2: **Visualization of Theorem 2.1 (Pareto Path Existence).** Let two neighboring Pareto-optimal policies, π_A and π_B, differ from each other at exactly $k = 4$ states. As established by the theorem, we can find a sequence of intermediate policies that are a 1-state action flip from each other and are also guaranteed to be on the Pareto front.

We refer to the sequence $(\pi_0, \pi_1, \ldots, \pi_k)$ as a *Pareto path*. The existence of this path guarantees that the Pareto front is inherently populated by structurally similar policies, precluding scenarios where all optimal policies have completely disjoint behaviors. To the best of our knowledge, Theorem 2.1 is a new geometric characterization of MOMDPs of independent interest. Notably, it generalizes the findings of [25] by removing their requirement of a unique mapping between policies and returns. The proof is provided in Appendix B, where we contrast our result with [25] and additionally show that all stochastic mixtures of the policies along the segment $J(\pi_{i-1})$ to $J(\pi_i)$ are also Pareto optimal. This can be visualized in Figure 2.

We conclude that the Pareto front is densely populated by structurally similar policies. We will leverage this structural fact in developing our algorithm for imitation learning in MOMDPs that we introduce next.

2.2 Imitation Learning in MOMDPs

In imitation learning in MOMDPs, a learning algorithm aims to learn a set of ε-PF policies in an MOMDP with unknown transitions P, reward r and initial distribution ν_0. The learning signal is a collection of datasets $\{\mathcal{D}_\ell\}_{\ell=1}^{L}$. Each dataset contains state-action pairs sampled from the occupancy measure of an expert policy $\pi_\ell \in \Pi$, i.e. $\mathcal{D}_\ell = \{X_n^\ell, A_n^\ell\}_{n=1}^{N_\ell}$ and $X_n^\ell, A_n^\ell \sim \mu^{\pi_\ell}$ for each $n \in [N_\ell]$. Then, given that for each $\ell \in [L]$, π_ℓ is a PF-policy, we aim to find a learning algorithm that outputs a set of policies $\{\hat{\pi}_\ell\}_{\ell=1}^{L'}$ that are ε-PF.[3]

Assumption (Deterministic Experts) For ease of exposition, we will assume in the rest of the main text that the expert policies π_ℓ are deterministic. However, our framework naturally accommodates stochastic experts. We detail the extension to stochastic Pareto policies in Appendix C.

Reduction to single-expert imitation The problem can be reduced to single-expert imitation learning: it is indeed possible to learn each $\hat{\pi}_\ell$ by running an imitation learning algorithm for single-objective MDPs, such as behavioral cloning [34, 10, 38]. This approach completely neglects the information in the datasets $\{\mathcal{D}_j\}_{j\neq\ell}^{L}$, optimizing each policy independently as $\hat{\pi}_\ell = \operatorname{argmax}_{\pi\in\Pi} \sum_{X,A\in\mathcal{D}_\ell} \log \pi(A|X)$.

Invoking standard guarantees, such as [38, Theorem 1.2], we can conclude that $\hat{\pi}_\ell$ is an ε-PF policy if $N_\ell \geq \frac{|\mathcal{X}|}{(1-\gamma)^2\varepsilon}$. Moreover, the lower bounds for single-objective MDPs [38] imply that no better rate can be achieved by focusing only on $\mathcal{D}_\ell$ as input. Therefore, we have the following negative takeaway.

[3] Notice that we do not restrict the number of output policies to be equal to the number of experts. That is, we can have $L' \neq L$.

Failure I: Treating imitation learning in MOMDPs as imitation learning in MDPs requires each expert dataset to have size linear in the state space cardinality, i.e. $\forall \ell \in [L]$, $N_\ell = \Omega(|\mathcal{X}|)$.

Given that PF-policies agree on a large number of states (see the structural properties highlighted in Section 2.1) there is hope to require that only the size of the joint dataset $\sum_{\ell=1}^{L} N_\ell$ grows with $|\mathcal{X}|$ as opposed to the much stricter requirement of imposing that each single expert dataset does so.

Single output BC A tempting approach in this direction is not splitting the data at all, that is, to run behavioral cloning over the full dataset. This generates a policy $\pi_{\text{out}} = \text{argmax}_{\pi \in \Pi} \sum_{X,A \in \cup_{\ell=1}^{L} \mathcal{D}_\ell} \log \pi(A|X)$. However, this approach suffers from an even more severe failure mode. Indeed, the next theorem proves that single output BC can fail to learn an ε-PF policy even under infinite data.

Theorem 2.2. *Let $\pi_{\text{out}} = \text{argmax}_{\pi \in \Pi} \sum_{X,A \in \cup_{\ell=1}^{L} \mathcal{D}_\ell} \log \pi(A|X)$. There exists an MOMDP and a policy $\pi^\star$ such that: $\forall i \in [d]$ it holds that*

$$J_i(\pi^\star) \geq J_i(\pi_{\text{out}}) + \frac{1}{3(1-\gamma)} .$$

The main idea to prove the negative result is to realize that given π, π' that are PF-policies, it is not true that the average policy, defined as $\frac{\pi(a|x)+\pi'(a|x)}{2}$ for all $x, a \in \mathcal{X} \times \mathcal{A}$, is a PF-policy. This fact is also depicted in Figure 1. We can therefore conclude the section with a second negative takeaway.

Failure II: Running imitation learning without splitting the dataset can fail to learn an ε-PF policy even with an infinite amount of data.

We now move to the next section to show that splitting the data *in the right manner* avoids both failure cases.

3 Multi-Output Augmented Behavioral Cloning (MA-BC)

Our new algorithm, Multi-Output Augmented Behavioral Cloning (MA-BC), is inspired by the structural result in Theorem 2.1 which establishes that PF-policies used by the experts are likely to agree on the same actions in several states. Our idea is then to use this fact to enable a large data reuse mechanism, while avoiding **Failure II**. MA-BC outputs $L' = L$ policies trained by Behavioral Cloning on a curated dataset. In particular, our algorithm first identifies the set of states in which the experts showed contradictory behavior (see Line 2 in Algorithm 1). More precisely, a state is included in $\mathcal{D}_{\text{div}}$ if it appeared in at least two datasets of the collection $\{\mathcal{D}_\ell\}_{\ell=1}^{L}$ but, in at least one case, the actions taken by two experts in that state are different. Then, in (Algorithm 1 Line 3) we define the set of common states as the set difference between the joint dataset $\cup_{\ell=1}^{L} \mathcal{D}_\ell$ and the diverging dataset $\mathcal{D}_{\text{div}}$. For this operation, we use the convention that removing a state from $\cup_{\ell=1}^{L} \mathcal{D}_\ell$ means removing all its occurrences together with all the actions taken at that state.

Finally, we learn L policies by running L instances of BC. In particular, to learn the output policy $\hat{\pi}_\ell$, we use the dataset $\mathcal{D}_{\text{common}} \cup \mathcal{D}_\ell$. That is, we use the union of: (i) the expert-specific dataset $\mathcal{D}_\ell$ and (ii) the common dataset $\mathcal{D}_{\text{common}}$.

The rationale behind the design of MA-BC Crucially, $\mathcal{D}_{\text{common}}$ also contains information collected by the other experts. In particular, the curated dataset includes states that have not been visited by the expert π_ℓ but by the other experts. This allows the *experts to help each other* and, in turn, to bypass **Failure I**. At the same time, **Failure II** is avoided by including only states in which no expert disagreement has been observed. The avoidance of both failures is made clear by the following theorem which is our main result.

Theorem 3.1. *Let $\{\pi_\ell\}_{\ell=1}^{L}$ be Pareto front policies and let $\mathcal{X}_{\text{common}} \subset \mathcal{X}$ be defined as $\mathcal{X}_{\text{common}} := \{x \in \mathcal{X} : \pi_1(x) = \pi_2(x) = \cdots = \pi_L(x)\}$ and let $\{\hat{\pi}_\ell\}_{\ell=1}^{L}$ be the output of MA-BC. Then, we have that for all $\ell \in [L]$, $\hat{\pi}_\ell$ is ε_ℓ-PF with*

Algorithm 1 MA-BC: Multi-Output Augmented Behavioral Cloning

Input: Single expert datasets $\{\mathcal{D}_\ell\}_{\ell=1}^L$.
Identify the diverging states

$$\mathcal{D}_{\text{div}} = \left\{ x \in \mathcal{X}, a \in \mathcal{A} \mid \exists \ell, \ell' \in [L], \ell \neq \ell', \exists\ n \in [N_\ell], n' \in [N_{\ell'}] \right. \\ \left. \text{s.t. } X_n^\ell = X_{n'}^{\ell'} = x \text{ AND } A_n^\ell \neq A_{n'}^{\ell'} \right\}$$

Define the common states

$$\mathcal{D}_{\text{common}} = (\cup_{\ell=1}^L \mathcal{D}_\ell) \setminus \mathcal{D}_{\text{div}}$$

% Run BC on the curated dataset
for $\ell = 1, \dots, L$ **do**

$$\hat{\pi}_\ell = \arg\max_{\pi \in \Pi} \sum_{X,A \in \mathcal{D}_{\text{common}} \cup \mathcal{D}_\ell} \log \pi(A|X).$$

end for
Output: $\{\hat{\pi}_\ell\}_{\ell=1}^L$.

$$\varepsilon_\ell = C_{\pi_\ell} \frac{|\mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N} + \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N_\ell},$$

where

$$C_{\pi_\ell} := \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} := \max_{x \in \mathcal{X}_{\text{common}} : \nu^{\pi_{\text{data}}}(x) > 0} \frac{\nu^{\pi_\ell}(x)}{\nu^{\pi_{\text{data}}}(x)},$$

and $\nu^{\pi_{\text{data}}}$ *is defined as*

$$\nu^{\pi_{\text{data}}}(x) = \frac{\sum_{j=1}^L N_j \nu^{\pi_j}(x)}{N}, \qquad \textit{for each } x \in \mathcal{X}.$$

We highlight that the concentrability parameter C_{π_ℓ} is restricted to the common state space $\mathcal{X}_{\text{common}}$. An immediate corollary reads as follows:

Corollary 3.1. *Fix expert sampling proportions* $\alpha_1, \dots, \alpha_L > 0$ *satisfying* $\sum_{\ell=1}^L \alpha_\ell = 1$*, and consider sample sizes* $N_\ell = \alpha_\ell N$*, so that the allocation remains fixed as* N *increases. For any* $\varepsilon > 0$*, let* $|\mathcal{X} \setminus \mathcal{X}_{\text{common}}| = K$ *and assume that* $|\mathcal{X}| \geq 2K$*. Let*

$$C_{\pi_{\text{data}}} := \max_{\ell \in [L]} \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}}, \qquad \nu^{\pi_{\text{data}}} = \sum_{\ell=1}^L \alpha_\ell \nu^{\pi_\ell}.$$

Then, the whole set $\{\hat{\pi}_\ell\}$ *output by MA-BC is* ε*-PF if*

$$N \geq \frac{2\, C_{\pi_{\text{data}}}\, |\mathcal{X}|}{e(1-\gamma)^2 \varepsilon} \qquad \textit{and} \qquad N_\ell \geq \frac{2K}{e(1-\gamma)^2 \varepsilon} \quad \textit{for all } \ell \in [L].$$

The first important remark is that the single dataset sizes $\{N_\ell\}_{\ell=1}^L$ are now required to scale linearly only with the number of diverging states K, which can be much smaller than $|\mathcal{X}|$. In this regime, our approach is expected to be much more efficient than the approach based on imitation learning in MDPs, which suffers from **Failure I**. Moreover, **Failure II** is avoided because Theorem 3.1 certifies that the output policies are PF-policies with infinite data, i.e. when $N \to \infty$ and $N_\ell \to \infty$.

On the role of the concentrability coefficient C_{π_ℓ} Interestingly, C_{π_ℓ} in Theorem 3.1 determines how much the experts' common dataset is informative for each individual learner. To illustrate this, consider a simple setting with $L = 2$. In this case, we have that $C_{\pi_1} = \max_{x \in \mathcal{X}_{\text{common}}} \frac{N \nu^{\pi_1}(x)}{N_1 \nu^{\pi_1}(x) + N_2 \nu^{\pi_2}(x)}$. In the first extreme scenario, assume the two experts are not informative for each other. In particular, let us consider the case in which there exists a state $x \in \mathcal{X}_{\text{common}}$ in which $\nu^{\pi_1}(x) > 0$ but $\nu^{\pi_2}(x) = 0$. Unfortunately, we have that $\left\| \frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} = \frac{N}{N_1}$ and plugging this into our general bound of Theorem 3.1 we have that the right hand side reduces to

$\frac{|\mathcal{X}|}{e(1-\gamma)^2 N_1}$, which is the same rate we could get using only the data from expert π_1 (i.e. we fall back to **Failure I**). In this situation, the data from the other expert do not help because C_{π_1} is as high as it can be. A more benign situation occurs when, for all $x \in \mathcal{X}_{\text{common}}$ we have that $\nu^{\pi_2}(x) \geq c_1 \nu^{\pi_1}(x)$ and $\nu^{\pi_1}(x) \geq c_2 \nu^{\pi_2}(x)$. Intuitively, this means that the two experts induce a similar state occupancy measure even though they choose different actions in some states. Then, we have that $C_{\pi_1} \leq \frac{N}{N_1 + c_1 N_2} \leq \frac{N}{c_1 N_1 + c_1 N_2} = \frac{1}{c_1}$, where we assumed that $c_1 \leq 1$. In this case, the right hand side in Theorem 3.1 becomes $\frac{|\mathcal{X}_{\text{common}}|}{e c_1 (1-\gamma)^2 N} + \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N_1}$, which is faster than what we would get using the dataset from a unique expert, especially when there are many common states. In this situation we ensure that the second term that scales as $1/N_1$ becomes unaffected by the number of states. At the same time, while the first term depends on the number of states, it exhibits a faster $\mathcal{O}(1/N)$ decay, scaling inversely with the size of the joint dataset. The inverse dependence on c_1 implies that the more similar the experts' state occupancy measures are, the faster the first term decays.

Extension to unsplit experts Our algorithm assumed access to the split datasets, where the expert identities are revealed. In Appendix D, we consider the setting where the dataset is not split, and the learner must first identify the latent experts' identities. We prove a similar bound to Theorem 3.1 in this setting (see Theorem D.1).

Is C_{π_ℓ} necessary? From the above discussion regarding the role of C_{π_ℓ}, it emerged that MA-BC is not always better than reducing the problem to imitation learning in MDPs. Indeed, when C_{π_ℓ} takes its largest possible value, each individual dataset $\mathcal{D}_\ell$ is required to grow linearly with $|\mathcal{X}|$, at least according to our analysis in Theorem 3.1. Therefore, our analysis states that MA-BC is beneficial only when C_{π_ℓ} is reasonably small. As we show in the next theorem, the dependence on C_{π_ℓ} is not an artifact of our analysis, but is rather unavoidable. That is, for any offline imitation learning algorithm, there exists a two-objective MDP in which learning an ε-PF policy requires at least $\Omega(K/\varepsilon)$ state-action pairs in both expert-specific datasets and $\Omega(C_{\pi_{\text{data}}} |\mathcal{X}| / \varepsilon)$ state-action pairs in the joint dataset.

Theorem 3.2. *Fix $\gamma \in [1/2, 1)$, $N_1, N_2 \geq 1$, $p_1, p_2 \in (0, 1/2]$, and an index $\ell \in \{1, 2\}$. For any offline expert-indexed multi-output imitation algorithm that receives*

$$\mathcal{D}_j = \left\{(X_n^j, A_n^j)\right\}_{n=1}^{N_j}, \qquad (X_n^j, A_n^j) \sim \mu^{\pi_j},$$

and outputs $\{\hat{\pi}_1, \hat{\pi}_2\}$, there exists a two-objective MOMDP and two deterministic PF-policies π_1, π_2 such that

$$\varepsilon_\ell \geq \Omega\left(\min\left\{\frac{p_\ell}{1-\gamma}, C_{\pi_\ell} \frac{|\mathcal{X}_{\text{common}}|}{N(1-\gamma)^2}\right\} + \min\left\{\frac{1-p_\ell}{1-\gamma}, \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{N_\ell (1-\gamma)^2}\right\}\right),$$

where ε_ℓ is any value for which $\hat{\pi}_\ell$ is an ε_ℓ-PF-policy and

$$C_{\pi_\ell} := \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}}.$$

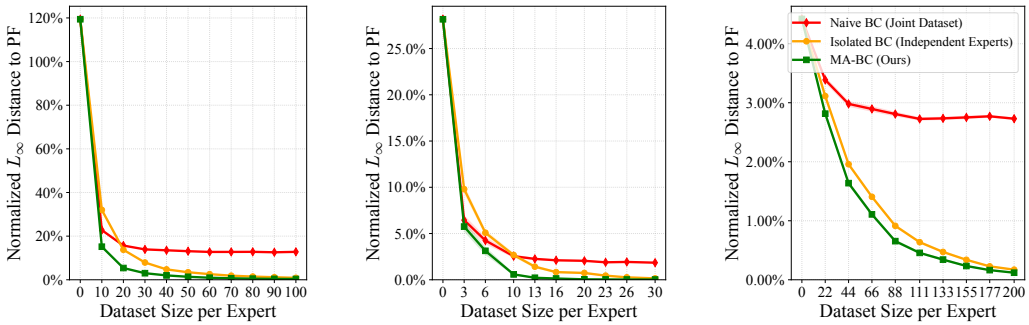
Moreover, for the constructed MOMDP,

$$C_{\pi_\ell} = \frac{N p_\ell}{N_1 p_1 + N_2 p_2}.$$

The universal constant hidden in $\Omega(\cdot)$ is independent of $N_1, N_2, \gamma, p_1, p_2$, and the state-space cardinalities.

Here, *expert-indexed* means that the ℓ-th output policy is tied to the ℓ-th demonstrated expert (in the construction of Appendix B.4, this is enforced by fixing the action at the root state). Without such a requirement, an algorithm could output the same PF-policy L times, and no bound involving N_ℓ could hold. Consequently, the multiplicative penalty of the concentrability coefficient C_{π_ℓ} is unavoidable. Specifically, it scales the $\mathcal{O}(1/N)$ term for the expert $\ell^\star = \arg\max_{\ell \in [L]} C_{\pi_\ell}$. Note that we exclude single-output imitation learning algorithms from the lower bound because for those, we have already proven a much stronger negative result in Theorem 2.2, which prevents convergence to PF-policies even under infinite data. In contrast, Theorem 3.2 states that the PF is approachable, but at a statistical rate that is no faster than the maximum of two terms, which matches the two terms in the rate obtained by our MA-BC (see Theorem 3.1).

Having developed a complete theoretical understanding of MA-BC in the tabular case, we present its extension to continuous state and action spaces in the next section.



(a) Deep Sea Treasure (b) Slippery Y-Maze (c) Resource Gathering

Figure 3: Sample efficiency and suboptimality gaps across all three discrete multi-objective environments. The y-axis represents the normalized L_∞ distance to the continuous Pareto frontier. Across all domains, Naive BC (Joint Dataset) fails to converge to an ε-PF policy (**Failure II**). While Isolated BC (Independent Experts) converges to the Pareto Front, it exhibits worse sample efficiency than MA-BC (**Failure I**). Solid lines denote the mean L_∞ distance to the Pareto front, while shaded regions represent the standard error of the mean over 100 trials.

4 Scaling MA-BC to Continuous State and Action Spaces

Our approach naturally generalizes to continuous state-action spaces by augmenting the dataset of each expert $\mathcal{D}_\ell$ with the following *compatible state action* pairs:

$$\mathcal{D}_\ell^+ = \left\{X', A' \in \cup_{\ell' \neq \ell}^{L} \mathcal{D}_{\ell'} \ : \ \exists X, A \in \mathcal{D}_\ell \text{ s.t. } \|X - X'\| \leq \delta \text{ AND } \|A - A'\| \leq L^{\pi_\ell} \|X - X'\|\right\}$$

where L^{π_ℓ} is the Lipschitz constant of the policy of the ℓ^{th} expert, i.e. for all $x, x' \in \mathcal{X}$ it holds that $\|\pi_\ell(x) - \pi_\ell(x')\| \leq L^{\pi_\ell} \|x - x'\|$. The quantity $L^{\pi_\ell} \|X - X'\|$ is the minimum meaningful action threshold. Reducing it further risks discarding bias-free samples because if the experts were in fact all equal, we could still expect actions at distance $L^{\pi_\ell}\delta$ in states at distance δ. Then, MA-BC outputs the set of policies $\{\hat{\pi}_\ell\}_{\ell=1}^{L}$ such that for each $\ell \in [L]$, $\hat{\pi}_\ell \in \operatorname{argmin}_{\pi:\mathcal{X}\to\mathcal{A}} \sum_{X,A \in \mathcal{D}_\ell \cup \mathcal{D}_\ell^+} \|A - \pi(X)\|^2$.

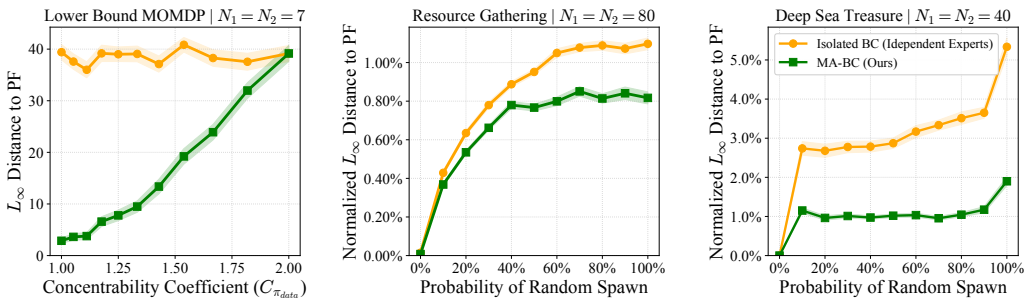
A bias-data tradeoff The parameter δ tunes a tradeoff between the bias introduced in the regression targets by augmenting the dataset with $\mathcal{D}_\ell^+$ and the size of $\mathcal{D}_\ell^+$ itself. In the extreme case where $\delta = 0$, no bias is introduced but we add some state action pairs in $\mathcal{D}_\ell^+$ only if the same state is visited by two different experts. However, this never happens in a continuous domain. On the contrary, as δ increases, we allow more and more bias but the size of $\mathcal{D}_\ell^+$ is also guaranteed not to decrease. Indeed, it is easy to notice that every state-action pair that is in the compatible set at level δ is also compatible for any $\delta' \geq \delta$. We highlight that the extension of MA-BC to continuous states and actions remains a heuristic, as we lack formal guarantees in this setting. However, we empirically investigate the tradeoffs in this setting in Section 5, highlighting many interesting phenomena.

5 Experiments

We design a suite of experiments across four distinct domains. First, we evaluate our algorithm on three discrete MOMDP benchmarks: Deep Sea Treasure [63], Resource Gathering [5] and Slippery Y-Maze. These environments can be visualized in Figure 8. We further consider a continuous LQR drone navigation task [4]. All environments are described in detail in Appendix E. Across all domains, we consistently demonstrate that MA-BC converges to the Pareto front while achieving superior sample efficiency. We further provide a highly visual, pedagogical walkthrough of the Deep Sea Treasure environment in Appendix E.3. The implementation of all experiments is available at `https://github.com/ziyadsheeba/mabc.git`.

5.1 Sample Efficiency

We first evaluate the sample-efficiency trade-offs in multi-objective imitation across our discrete benchmarks. Our primary metric is the L_∞ distance to the continuous returns Pareto frontier,



(a) Lower Bound MOMDP (b) Resource Gathering (c) Deep Sea Treasure

Figure 4: Evaluation of dataset concentrability on MA-BC performance. **(Left)** In the Lower Bound MDP, as the concentrability factor decreases, the performance gap between MA-BC and Isolated BC widens. **(Middle and Right)** We vary the stochasticity of the initial state distribution in the Resource Gathering and the Deep Sea Treasure environments. As the initial state becomes more stochastic, more trajectories overlap between experts, thus providing MA-BC with an advantage. Solid lines denote the mean L_∞ distance to the Pareto front, while shaded regions represent the standard error of the mean over 100 trials.

computed as $\min_{J^* \in \text{PF}} \|J(\hat{\pi}) - J^*\|_\infty$, where $\hat{\pi}$ is the estimated policy from the data. We apply coordinate-wise normalization to the returns to appropriately adjust for the different scales of each objective. We compute this distance exactly as described in Appendix E.2. For the multi-output algorithms (Isolated BC and MA-BC), we average the distance across the experts and consolidate the results in Figure 3. In each of the environments, we consider data coming from two extreme experts, each optimizing for only one dimension of the reward vector over a uniform initial state distribution. For the Resource Gathering environment, we choose the two experts to be the Gold-optimizing and the Gem-optimizing agents.

Across all environments, the Naive BC approach (which jointly trains on all aggregated data) completely fails to converge to an ε-PF policy as the dataset size increases, which validates Theorem 2.2, highlighting **Failure II**. The Isolated BC baseline (training independent experts) successfully converges to the Pareto Front. However, it suffers from sample inefficiency as it ignores the underlying structure of the MOMDP (**Failure I**). In stark contrast, MA-BC converges to the Pareto front at a much faster statistical rate by leveraging shared structural knowledge. This confirms the theoretical guarantees of Theorem 3.1.

5.2 The Impact of Concentrability

Theorem 3.2 suggests that the performance difference between MA-BC and Isolated BC is governed by the concentrability parameter $C_{\pi_{\text{data}}}$. Intuitively, this means that the more experts' trajectories overlap, the more sample-efficient MA-BC becomes. To evaluate this relationship directly, our lower bound MOMDP allows us to explicitly compute $C_{\pi_{\text{data}}}$. As shown in Figure 4a, as $C_{\pi_{\text{data}}}$ decreases, MA-BC actively capitalizes on the shared structure. The resulting empirical gap precisely mirrors our theoretical expectations, validating Theorem 3.2. In order to capture this effect in the Resource Gathering and Deep Sea Treasure environments, we vary the stochasticity of the initial state distribution by controlling the probability of spawning uniformly at random versus starting from a fixed initial state. As illustrated in Figures 4b and 4c, the performance gap widens as the initial state distribution becomes more stochastic. This increased stochasticity causes the experts' trajectories to overlap more frequently, effectively simulating a smaller $C_{\pi_{\text{data}}}$.

5.3 Scaling to Continuous State-Action Spaces (LQR Drone Control)

Finally, we demonstrate that MA-BC scales to continuous domains using a linearized 6-DOF quadcopter navigation task [4]. The multi-objective conflict lies between speed (Agile Expert) and energy conservation (Economic Expert). We estimate the underlying LQR controllers via Ridge Regression ($\lambda = 0.001$) on trajectories sampled from these Pareto-extreme experts (see Appendix E.1.4). Using our continuous MA-BC extension (Section 4), we apply a spatial tolerance δ to pool compatible state-action pairs across datasets. We apply an empirical min/max normalization of the states and

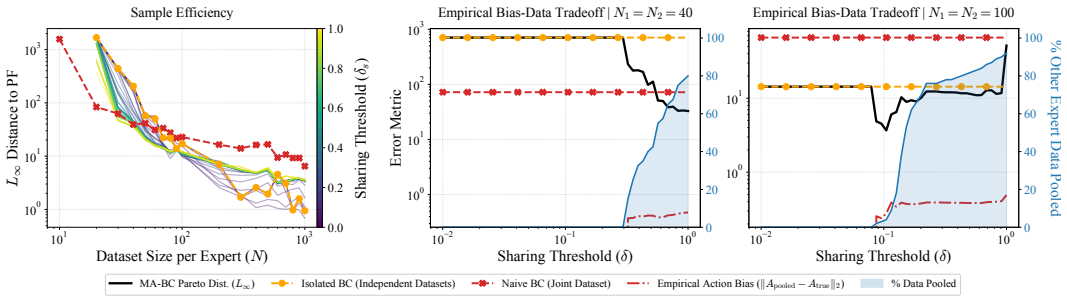


Figure 5: Continuous state-action imitation results for the LQR drone control task. The plots report the performance on recovering the Economic expert for the multi-output learners (i.e., MA-BC and Isolated BC). **(Left)** Sample efficiency across varying δ thresholds for MA-BC. **(Middle & Right)** The bias-data tradeoff at $N = 40$ and $N = 100$, demonstrating how the optimal pooling threshold δ shrinks as the baseline data increases.

actions to facilitate tuning the parameter δ. We compute the Lipschitz constants using the true experts' LQR matrices. Figure 5 evaluates the recovery of the Economic Expert.

As shown in the left panel of Figure 5, large δ (yellow lines) drastically improve performance in the extreme low-data regime by aggressively pooling data, whereas small δ (purple lines) achieve the lowest asymptotic error given sufficient data. This empirical bias-data tradeoff is explicitly quantified in the middle and right panels of Figure 5. As δ increases, MA-BC pools a larger percentage of the other expert's data (blue shaded area), initially reducing the Pareto distance (solid black line). However, excessive δ injects severe empirical action bias (dashed red line), causing the error to eventually rebound. The empirical action bias is computed as $\|A_{\text{pooled}} - A_{\text{true}}\|_2$, i.e., the action error introduced by pooling the data. Crucially, comparing the $N = 40$ and $N = 100$ scenarios reveals a data-dependent "sweet spot": as the available data grows, the learner requires less pooling to reduce variance and becomes highly sensitive to bias, causing the optimal δ threshold to shrink and shift leftwards.

6 Conclusions

In this paper, we studied offline imitation learning in MOMDPs for the first time. We identified a key structural property of MOMDPs, the Pareto path existence, which inspired the design of our new algorithm MA-BC. It learns to imitate Pareto Front experts when they are similar while retaining the single expert guarantees when the experts are dissimilar. Moreover, we have proven the optimality of MA-BC across the class of offline imitation algorithms. Several exciting directions remain open. First, it would be interesting to provide convergence guarantees for our continuous control extension of MA-BC. The smoothness of the Pareto front identified in [17] offers a natural starting point for this analysis. Furthermore, our new structural result in Theorem 2.1 enables the development of traversal algorithms for Multi-Objective RL that apply to a larger class of MOMDPs compared to the state-of-the-art in [25]. Finally, a promising architectural direction involves developing neural network structures that can implicitly learn to pool shared state-action representations while maintaining specialized outputs for divergent expert behaviors.

Acknowledgments and Disclosure of Funding

We thank Amin Charusaie and Adrian Müller for the valuable discussions throughout the project. C.V. and Z.S. are funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under project 468806714 of the Emmy Noether Programme and under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645. C.V. also gratefully acknowledges funding from the European Union (ERC, ConSequentIAL, 101165883). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. C.V. and Z.S. also thank the international Max Planck Research School for Intelligent Systems (IMPRS-IS) for their support. L.V. was supported by the Swiss Data Science Center under fellowship number P22_03. Research by V.C. was sponsored by the Army Research Office under Grant Number W911NF-24-1-0048 and funded by the Swiss National Science Foundation (SNSF) under grant number 2000-1-240094.

References

[1] P. Abbeel, D. Dolgov, A. Y. Ng, and S. Thrun. Apprenticeship learning for motion planning with application to parking lot navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008.

[2] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2004.

[3] Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The twelfth international conference on learning representations*, 2024.

[4] Shibbir Ahmed, Baijing Qiu, Chun-Wei Kong, Huang Xin, Fiaz Ahmad, and Jinlong Lin. A data-driven dynamic obstacle avoidance method for liquid-carrying plant protection uavs. *Agronomy*, 12(4), 2022.

[5] Leon Barrett and Srini Narayanan. Learning all optimal policies with multiple criteria. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 41–47, New York, NY, USA, 2008. Association for Computing Machinery.

[6] Abdeslam Boularias, Jens Kober, and Jan Peters. Relative entropy inverse reinforcement learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 182–189. JMLR Workshop and Conference Proceedings, 2011.

[7] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion, 2024.

[8] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, et al. Ultrafeedback: Boosting language models with scaled ai feedback. *arXiv preprint arXiv:2310.01377*, 2023.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[10] Dylan J Foster, Adam Block, and Dipendra Misra. Is behavior cloning all you need? understanding horizon in imitation learning. *arXiv preprint arXiv:2407.15007*, 2024.

[11] Tim Franzmeyer, Edith Elkind, Philip Torr, Jakob Foerster, and Joao Henriques. Select to perfect: Imitating desired behavior from large multi-agent data. *arXiv preprint arXiv:2405.03735*, 2024.

[12] Till Freihaut, Luca Viano, Volkan Cevher, Matthieu Geist, and Giorgia Ramponi. Learning equilibria from data: Provably efficient multi-agent imitation learning, 2025.

[13] Till Freihaut, Luca Viano, Emanuele Nevali, Volkan Cevher, Matthieu Geist, and Giorgia Ramponi. Rate optimal learning of equilibria from data. *arXiv preprint arXiv:2510.09325*, 2025.

[14] Nagata Furukawa. Characterization of optimal policies in vector-valued markovian decision processes. *Mathematics of operations research*, 5(2):271–279, 1980.

[15] J. Ho and S. Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[16] J. Ho, J. K. Gupta, and S. Ermon. Model-free imitation learning with policy optimization. In *International Conference on Machine Learning (ICML)*, 2016.

[17] Ali Jadbabaie, Devavrat Shah, and Sean R Sinclair. Multi-objective lqr with linear scalarization. *arXiv preprint arXiv:2408.04488*, 2024.

[18] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[19] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *8th Annual Conference on Robot Learning*, 2024.

[20] Woo Kyung Kim, Minjong Yoo, and Honguk Woo. Pareto inverse reinforcement learning for diverse expert policy generation. *arXiv preprint arXiv:2408.12110*, 2024.

[21] Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. Pretraining language models with human preferences. In *International conference on machine learning*, pages 17506–17533. PMLR, 2023.

[22] Hoang M Le, Yisong Yue, Peter Carr, and Patrick Lucey. Coordinated multi-agent imitation learning. In *International Conference on Machine Learning*, pages 1995–2003. PMLR, 2017.

[23] Seungjae Lee, Yibin Wang, Haritheja Etukuru, H. Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior generation with latent actions. In *Multi-modal Foundation Model meets Embodied AI Workshop @ ICML2024*, 2024.

[24] Shiqi Lei, Kanghoon Lee, Linjing Li, and Jinkyoo Park. Learning strategy representation for imitation learning in multi-agent games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 18163–18171, 2025.

[25] Yining Li, Peizhong Ju, and Ness B Shroff. How to find the exact pareto front for multi-objective mdps? *arXiv preprint arXiv:2410.15557*, 2024.

[26] Haoye Lu, Daniel Herman, and Yaoliang Yu. Multi-objective reinforcement learning: Convexity, stationarity and pareto optimality. In *The Eleventh International Conference on Learning Representations*, 2023.

[27] H. Molter. *Classic graph problems made temporal – a parameterized complexity analysis*. Foundations of Computing. Universitätsverlag der TU Berlin, 2020.

[28] Antoine Moulin, Gergely Neu, and Luca Viano. Inverse q-learning done right: Offline imitation learning in q^π-realizable mdps, 2025.

[29] Antoine Moulin, Gergely Neu, and Luca Viano. Optimistically optimistic exploration for provably efficient infinite-horizon reinforcement and imitation learning. *arXiv preprint arXiv:2502.13900*, 2025.

[30] A. Y. Ng and S. J. Russell. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2000.

[31] T Osa, J Pajarinen, G Neumann, JA Bagnell, P Abbeel, and J Peters. An algorithmic perspective on imitation learning. *Foundations and Trends in Robotics*, 2018.

[32] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022.

[33] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[34] D. A. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural Computation*, 3(1):88–97, 1991.

[35] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., USA, 1st edition, 1994.

[36] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[37] Nived Rajaraman, Yanjun Han, Lin Yang, Jingbo Liu, Jiantao Jiao, and Kannan Ramchandran. On the value of interaction and function approximation in imitation learning. *Advances in Neural Information Processing Systems*, 34:1325–1336, 2021.

[38] Nived Rajaraman, Lin Yang, Jiantao Jiao, and Kannan Ramchandran. Toward the fundamental limits of imitation learning. *Advances in Neural Information Processing Systems*, 33:2914–2924, 2020.

[39] Diederik M Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48:67–113, 2013.

[40] Diederik Marijn Roijers. *Multi-Objective Decision-Theoretic Planning*. PhD thesis, University of Amsterdam, 2016.

[41] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

[42] Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

[43] Stuart Russell. Learning agents for uncertain environments (extended abstract). In *Annual Conference on Computational Learning Theory (COLT)*, 1998.

[44] Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. Social bias frames: Reasoning about social and power implications of language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5477–5490, Online, July 2020. Association for Computational Linguistics.

[45] Lior Shani, Tom Zahavy, and Shie Mannor. Online apprenticeship learning. *arXiv:2102.06924*, 2021.

[46] Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953.

[47] Ziyad Sheebaelhamd, Michael Tschannen, Michael Muehlebach, and Claire Vernade. Quantization-free autoregressive action transformer. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

[48] Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. Multi-agent generative adversarial imitation learning. *Advances in neural information processing systems*, 31, 2018.

[49] Gokul Swamy, Sanjiban Choudhury, J Andrew Bagnell, and Steven Wu. Of moments and matching: A game-theoretic framework for closing the imitation gap. In *International Conference on Machine Learning*, pages 10022–10032. PMLR, 2021.

[50] U. Syed, M. Bowling, and R.E. Schapire. Apprenticeship learning using linear programming. In *International Conference on Machine Learning (ICML)*, 2008.

[51] U. Syed and R. E. Schapire. A game-theoretic approach to apprenticeship learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2007.

[52] Jingwu Tang, Gokul Swamy, Fei Fang, and Zhiwei Steven Wu. Multi-agent imitation learning: Value is easy, regret is hard. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 27790–27816. Curran Associates, Inc., 2024.

[53] Peter Vamplew, Richard Dazeley, Adam Berry, Rustam Issabekov, and Evan Dekker. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Machine Learning*, 84:51–80, 2011.

[54] Luca Viano, Till Freihaut, Emanuele Nevali, Volkan Cevher, Matthieu Geist, and Giorgia Ramponi. Multi-agent imitation learning with function approximation: Linear markov games and beyond. *arXiv preprint arXiv:2602.22810*, 2026.

[55] Luca Viano, Angeliki Kamoutsi, Gergely Neu, Igor Krawczuk, and Volkan Cevher. Proximal point imitation learning. *Advances in Neural Information Processing Systems*, 35:24309–24326, 2022.

[56] Luca Viano, Stratis Skoulakis, and Volkan Cevher. Imitation learning in discounted linear MDPs without exploration assumptions. In *Forty-first International Conference on Machine Learning*, 2024.

[57] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023.

[58] Joe Watson, Sandy H. Huang, and Nicolas Heess. Coherent soft imitation learning, 2023.

[59] D.J. White. Multi-objective infinite-horizon discounted markov decision processes. *Journal of mathematical analysis and applications*, 89(2):639–647, 1982.

[60] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics, 2018.

[61] Markus Wulfmeier, Michael Bloesch, Nino Vieillard, Arun Ahuja, Jörg Bornschein, Sandy Huang, Artem Sokolov, Matt Barnes, Guillaume Desjardins, Alex Bewley, et al. Imitating language via scalable inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 37:90714–90735, 2024.

[62] Tian Xu, Ziniu Li, Yang Yu, and Zhi-Quan Luo. Provably efficient adversarial imitation learning with unknown transitions. *arXiv preprint arXiv:2306.06563*, 2023.

[63] Runzhe Yang, Xingyuan Sun, and Karthik Narasimhan. A generalized algorithm for multi-objective reinforcement learning and policy adaptation. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[64] Lantao Yu, Jiaming Song, and Stefano Ermon. Multi-agent adversarial inverse reinforcement learning. In *International conference on machine learning*, pages 7194–7201. PMLR, 2019.

[65] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.

[66] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *National Conference on Artificial Intelligence (AAAI)*, 2008.

Contents of Appendix

A Related Work **17**

B Omitted Proofs **19**

B.1 Proof of Theorem 2.1 19

B.2 Proof of Theorem 2.2 19

B.3 Proof of Theorem 3.1 20

B.4 Proof of Theorem 3.2 22

C MA-BC with Stochastic Experts **28**

D MA-BC with Unsplit Expert Data **29**

D.1 Upper Bound for Unsplit MA-BC 30

E Experimental Details **33**

E.1 Environment Details 33

E.2 Exact L_∞ Distance to the Pareto Front via Linear Programming 35

E.3 A Pedagogical Deep Dive into the Deep Sea Treasure Environment 36

A Related Work

Our work builds on the vast single-objective MDP imitation learning literature and on the classic and more recent characterizations of the Pareto Front in MOMDPs. Imitation learning has been a flourishing field in the last three decades, motivated first by robotics [6] and autonomous driving [34] and more recently by language modeling [61, 3]. Common approaches include reductions to supervised learning [34, 42], RL in the loop [30, 2, 1, 51, 50], online learning [45, 56, 29, 49, 28], regularized RL [66, 16, 15, 55, 58] and reward free RL [62].

Our work relies on Behavioral Cloning [34], which is the first imitation learning algorithm. While originally used as a heuristic and long thought to be suboptimal compared to interactive algorithms [42, 41, 37], it has been proven to be optimal across offline algorithms [38], while [10] establishes that it is actually suboptimal compared to interactive algorithms only when the expert is highly non-stationary.

Secondly, we build on the foundations of multi-objective reinforcement learning, such as early characterizations and existence results of Pareto Front policies [14, 59] and on the recent result by [25], which established that, when each policy attains a distinct return, neighboring policies on the Pareto front differ in exactly one state. This result inspired our data-sharing idea in the context of imitation learning. Moreover, the Pareto path existence is a direct generalization of their results that allows multiple policies to attain the same returns. In that case, we establish that each vertex on the Pareto front can correspond to multiple policies and that neighboring vertices can always be mapped to policies differing in at most 1 state. Our result is also related to the recent line of literature [52, 12, 13, 54] which also studied imitation from multiple experts, who are, however, assumed to be in an equilibrium profile rather than in a Pareto Front configuration. Next, we review the most closely related works in each area mentioned above.

Classical Imitation Learning Imitation Learning has been shown to be successful in autonomous driving applications in [34], which proposed to model the problem as supervised learning, thus inventing behavioral cloning, the algorithm on which we build. Follow-up works focused on the problem of recovering a reward function in addition to reconstructing the expert policy [30, 43, 2, 1, 66], proving better error propagation analyses via expert interaction [42, 41] or with environment access [66, 6, 16, 15]. We refer to the monograph [31] for a complete overview of the classics.

Recent theoretical developments The first solid sample complexity investigation in imitation learning is by [38], which also introduced the lower bound construction on which we build for ours to show that behavioral cloning is actually optimal across offline algorithms. Beyond the tabular setting, [10] proved that behavioral cloning is suboptimal compared to interactive algorithms only when the expert policy is non-stationary, or without parameter sharing in their terminology. This result shows that the benefit of interaction is more limited than previously thought. On a similar line, we show that MA-BC is optimal across offline algorithms that reconstruct the Pareto Front experts.

Imitation learning in multi-agent systems As previously mentioned, approaches that merge all the data as in LLM pretraining [36, 9] or that treat each expert individually [20] suffer from **Failure II** or **Failure I**, respectively. Moreover, the only existing approach [11] that interpolates between merging all the data and splitting them is still limited by treating all the data collected by an expert as equal. They are either all included or excluded. Therefore, in a simple example with two experts, one of the two failure modes would still apply. Decentralized imitation learning for learning team behaviors has been studied in [22, 48]. Moreover, other recent works studied imitation learning in multi-agent systems [64, 52, 12, 13, 24, 54]; however, taking the different perspective of considering that the expert follows a Nash (or Correlated in [52]) equilibrium in a Markov Game [46]. These two settings are quite distinct. In their setting each agent receives a scalar reward that depends on other players' strategies, that is the players are coupled. In contrast, in our setting, each agent receives a reward vector which is independent of which other agents are providing demonstrations. The setting of considering both vectorized rewards and coupled agents is perhaps an interesting future direction.

Structural properties of MOMDPs The existence of Pareto front policies in any MOMDP has been established by [14, 59] in the discounted setting. Moreover, [59] also suggested an algorithmic approach to reconstruct the state value functions of the Pareto front policies with a value-iteration style approach. Moreover, convexity of Pareto front has been established in [26]. These papers leave aside the question of how similar the Pareto front policies can be, which is instead settled by [25] where the authors prove that if each policy has a unique return, then neighboring policies on the

Pareto front are guaranteed to differ at only one state. Therefore, there exist very similar Pareto front policies. In our current work, we establish a more general result saying that even when the same value vector can be attained by multiple policies, we can always find, for each pair of neighboring return vectors, corresponding policies that differ in at most one state. This structural result also inspired the current approach to reconstruct the full Pareto front in a known MOMDP, which is known as Pareto Front Traversal. The basic idea is to start from a Pareto Front deterministic policy computed via scalarization and some single objective RL algorithm and then flip the policy at one state and check that the new policy (as well as any convex combination with the old one) remains on the Pareto Front. We notice that in the imitation learning context, such an approach is not viable because reward knowledge is needed for the algorithmic routine checking that the new policy is a PF policy.

B Omitted Proofs

We start by presenting the proofs omitted from the main text.

B.1 Proof of Theorem 2.1

Theorem 2.1. *[Pareto Path Existence] Let π_A and π_B be neighboring PF-policies that differ in k states. Then, there exists a finite sequence of $k+1$ deterministic PF-policies $(\pi_0, \pi_1, \ldots, \pi_k)$ such that $\pi_0 = \pi_A$ and $\pi_k = \pi_B$ and consecutive policies in the sequence differ in exactly one state, i.e. $\sum_{x \in \mathcal{X}} \mathbb{1}\{\pi_{i-1}(x) \neq \pi_i(x)\} = 1$ for all $i \in \{1, \ldots, k\}$.*

Proof. Let w be the vector for which policies π_A and π_B are neighboring PF-policies and let $E(J(\pi_A), J(\pi_B)) = \arg\max_{J \in \mathbb{J}} w^T J$ be the edge connecting the expected returns vector of policies π_A and π_B in the return polytope. Then, consider the single-objective MDP defined by the scalarized reward function $r_w(x, a) = w^\top r(x, a)$ and let us denote by $\Pi^\star_w$ the set of optimal policies for the scalar reward r_w. By the Bellman Optimality Equation, a deterministic policy π belongs to Π^*_w if it selects an optimal action at every state $x \in \mathcal{X}$: $\pi(x) \in \arg\max_{a \in \mathcal{A}} Q^*_w(x, a)$.

Since both π_A and π_B are in Π^*_w, both $\pi_A(x)$ and $\pi_B(x)$ are optimal actions for all x. By the Markov property, any deterministic policy constructed by independently selecting either $\pi_A(x)$ or $\pi_B(x)$ at each state will also select optimal actions, ensuring it is also in Π^*_w.

Suppose π_A and π_B differ at exactly k states: $\{x_1, \ldots, x_k\}$. We construct the sequence $(\pi_0, \ldots, \pi_k)$ by initializing $\pi_0 = \pi_A$, and iteratively defining π_i by flipping the action at state x_i from $\pi_A(x_i)$ to $\pi_B(x_i)$. By construction, $\pi_k = \pi_B$, and each adjacent pair differs by exactly one state, that is

$$\sum_{x \in \mathcal{X}} \mathbb{1}\{\pi_{i-1}(x) \neq \pi_i(x)\} = 1.$$

Furthermore, since every π_i only uses actions from either π_A or π_B, $\pi_i \in \Pi^*_w$ for all i. Thus, the return $J(\pi_i)$ remains on the optimal face $E(J(\pi_A), J(\pi_B))$ for the entire sequence. Since this face is the segment between two PF returns, none of its points is dominated (a point dominating a point of the segment would also maximize $w^\top J$, hence lie on the segment, and two points of a segment whose endpoints are mutually non-dominated cannot dominate each other), so every π_i is a PF-policy. □

Note the contrast between Theorem 2.1 and the result presented in [25]: Unlike in [25, Theorem 1], we do not assume that neighbors on the Pareto front differ at a single state, since this statement is not true in general (see counter example in Appendix E.3). The breaking of [25, Theorem 1] happens in highly structured environments typical of reinforcement learning benchmarks (e.g., grid-worlds with sparse, integer rewards), which can cause one of the following two phenomena to occur:

1. **Aliasing:** Multiple distinct deterministic policies collapse into the exact same geometric vertex of the return polytope.
2. **Edge-bound Policies:** Deterministic policies may map to the flat edge connecting two vertices, rather than forming new extreme points.

Because of these phenomena, the strict "distance-1" property between adjacent vertices breaks down. Our result in Theorem 2.1 holds in general, and is agnostic to the existence of *aliased* or *edge-bound* policies.

B.2 Proof of Theorem 2.2

Here, we present the proof of the failure case for the single-output BC.

Theorem 2.2. *Let $\pi_{\text{out}} = \text{argmax}_{\pi \in \Pi} \sum_{X, A \in \cup_{\ell=1}^{L} \mathcal{D}_\ell} \log \pi(A|X)$. There exists an MOMDP and a policy $\pi^\star$ such that: $\forall i \in [d]$ it holds that*

$$J_i(\pi^\star) \geq J_i(\pi_{\text{out}}) + \frac{1}{3(1-\gamma)}.$$

Proof. It is enough to consider a 2-objective MOMDP with a single (absorbing) state and 3 actions $\mathcal{A} = \{a_1, a_2, a_3\}$. We consider the following rewards,

$$r(a_1) = [1, 0] \quad r(a_2) = [0, 1] \quad r(a_3) = [0.5 + \alpha, 0.5 + \alpha].$$

Now we consider as experts the following policies

$$\pi_1(a_1) = 1 \quad \pi_2(a_2) = 1,$$

and we consider a dataset containing half of the samples collected from π_1 and half collected from π_2, which are both Pareto front policies (i.e., $N_1 = N_2$). Since there is a single state, the maximum likelihood policy is the empirical action frequency, so for any dataset size we get the policy $\pi_{\text{out}}(a_1) = 0.5$, $\pi_{\text{out}}(a_2) = 0.5$. Notice now that

$$J_1(\pi_{\text{out}}) = \frac{0.5}{1-\gamma} \quad J_2(\pi_{\text{out}}) = \frac{0.5}{1-\gamma}.$$

However, consider $\pi^\star$ as the policy that deterministically plays a_3 we obtain

$$J_1(\pi^\star) = \frac{0.5+\alpha}{1-\gamma} \quad J_2(\pi^\star) = \frac{0.5+\alpha}{1-\gamma},$$

for some α such that $0 < \alpha < 0.5$. Therefore, the policy π_{out} is not on the Pareto front, and its suboptimality gap $\alpha/(1-\gamma)$ grows with α. Then, the theorem statement follows from choosing $\alpha = 1/3$. □

B.3 Proof of Theorem 3.1

Next, we present the proof of our main result.

Theorem 3.1. *Let $\{\pi_\ell\}_{\ell=1}^L$ be Pareto front policies and let $\mathcal{X}_{\text{common}} \subset \mathcal{X}$ be defined as $\mathcal{X}_{\text{common}} := \{x \in \mathcal{X} : \pi_1(x) = \pi_2(x) = \cdots = \pi_L(x)\}$ and let $\{\hat{\pi}_\ell\}_{\ell=1}^L$ be the output of MA-BC. Then, we have that for all $\ell \in [L]$, $\hat{\pi}_\ell$ is ε_ℓ-PF with*

$$\varepsilon_\ell = C_{\pi_\ell} \frac{|\mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N} + \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N_\ell},$$

where

$$C_{\pi_\ell} := \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} := \max_{x \in \mathcal{X}_{\text{common}} : \nu^{\pi_{\text{data}}}(x) > 0} \frac{\nu^{\pi_\ell}(x)}{\nu^{\pi_{\text{data}}}(x)},$$

and $\nu^{\pi_{\text{data}}}$ is defined as

$$\nu^{\pi_{\text{data}}}(x) = \frac{\sum_{j=1}^L N_j \nu^{\pi_j}(x)}{N}, \qquad \text{for each } x \in \mathcal{X}.$$

Proof. First of all, we notice that the output of Algorithm 1 can be rewritten as (see [38])

$$\hat{\pi}_\ell(x) = \begin{cases} \pi_\ell(x) & \text{if} \quad x \in \mathcal{D}_\ell \\ \pi_j(x) & \text{if} \quad x \in \mathcal{D}_{\text{common}} \setminus \mathcal{D}_\ell, \text{ where } j \text{ is any expert with } x \in \mathcal{D}_j \\ 1/|\mathcal{A}| & \text{otherwise.} \end{cases}$$

In the second case, $\pi_j(x) = \pi_\ell(x)$ whenever $x \in \mathcal{X}_{\text{common}}$; for $x \notin \mathcal{X}_{\text{common}}$ the prediction may be wrong, which is accounted for below by charging an error whenever $x \notin \mathcal{D}_\ell$. The choice of playing actions uniformly at random in unobserved states is arbitrary but that is the choice that allows us to get the best upper bounds as shown in the following. From this point onwards, without loss of generality let us consider $\ell = 1$. Let us introduce the notation

$$\|J(\pi_1) - J(\hat{\pi}_1)\|_\infty = \max_{i \in [d]} |J_i(\pi_1) - J_i(\hat{\pi}_1)|.$$

Then, we can proceed as follows

$$\begin{aligned}
(1-\gamma)^2 \|J(\pi_1) - J(\hat{\pi}_1)\|_\infty &\leq \sum_{x \in \mathcal{X}} \nu^{\pi_1}(x) \sum_{a \in \mathcal{A}} \pi_1(a|x) \mathbb{1}\{a \neq \hat{\pi}_1(x)\} \\
&= \sum_{x \in \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x) \sum_{a \in \mathcal{A}} \pi_1(a|x) \mathbb{1}\{a \neq \hat{\pi}_1(x)\} \\
&\quad + \sum_{x \in \mathcal{X} \setminus \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x) \sum_{a \in \mathcal{A}} \pi_1(a|x) \mathbb{1}\{a \neq \hat{\pi}_1(x)\} \qquad (1)
\end{aligned}$$

Now, we control each term individually. Starting from the first one:

$$\begin{aligned}
&\sum_{x\in\mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)\sum_{a\in\mathcal{A}}\pi_1(a|x)\mathbb{1}\left\{a\neq\hat{\pi}_1(x)\right\}\\
&\qquad=\sum_{x\in\mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)\sum_{a\in\mathcal{A}}\pi_{\text{data}}(a|x)\mathbb{1}\left\{a\neq\hat{\pi}_1(x)\right\}\\
&\qquad\leq\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty}\sum_{x\in\mathcal{X}_{\text{common}}} \nu^{\pi_{\text{data}}}(x)\sum_{a\in\mathcal{A}}\pi_{\text{data}}(a|x)\mathbb{1}\left\{a\neq\hat{\pi}_1(x)\right\},
\end{aligned}$$

where for the first equality in the block above we have used that all experts agree on $\mathcal{X}_{\text{common}}$, so that we can define $\pi_{\text{data}}(x) := \pi_1(x) = \cdots = \pi_L(x)$ for all $x \in \mathcal{X}_{\text{common}}$. Then, taking expectation over the randomness involved in the collection of $\mathcal{D}_{\text{common}}$, and using that for $x \in \mathcal{X}_{\text{common}}$ the prediction of $\hat{\pi}_1$ is correct as soon as x is sampled once, i.e., as soon as it appears in $\mathcal{D}_{\text{common}}$, no matter which expert sampled the action at that state, we have that

$$\begin{aligned}
&\mathbb{E}\left[\sum_{x\in\mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)\sum_{a\in\mathcal{A}}\pi_1(a|x)\mathbb{1}\left\{a\neq\hat{\pi}_1(x)\right\}\right]\\
&\leq\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty,\mathcal{X}_{\text{common}}}\sum_{x\in\mathcal{X}_{\text{common}}} \nu^{\pi_{\text{data}}}(x)\mathbb{P}\left[x\notin\mathcal{D}_{\text{common}}\right]\\
&\leq\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty,\mathcal{X}_{\text{common}}}\sum_{x\in\mathcal{X}_{\text{common}}} \nu^{\pi_{\text{data}}}(x)(1-\nu^{\pi_{\text{data}}}(x))^N\\
&\leq\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty,\mathcal{X}_{\text{common}}}\sum_{x\in\mathcal{X}_{\text{common}}} \frac{1}{eN}\\
&\leq\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty,\mathcal{X}_{\text{common}}}\frac{|\mathcal{X}_{\text{common}}|}{eN}.
\end{aligned}$$

The second step uses $\mathbb{P}\left[x\notin\mathcal{D}_{\text{common}}\right]=\prod_{j=1}^{L}(1-\nu^{\pi_j}(x))^{N_j}\leq(1-\nu^{\pi_{\text{data}}}(x))^N$, which follows from Jensen's inequality applied to the concave function $\log(1-\cdot)$ with weights N_j/N (the two sides need not be equal). The third step uses $p(1-p)^N\leq pe^{-pN}\leq 1/(eN)$. This concludes the bound on the first term of (1). Now let us look at the second term.

$$\begin{aligned}
\mathbb{E}\left[\sum_{x\in\mathcal{X}\setminus\mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)\sum_{a\in\mathcal{A}}\pi_1(a|x)\mathbb{1}\left\{a\neq\hat{\pi}_1(x)\right\}\right]&\leq\mathbb{E}\left[\sum_{x\in\mathcal{X}\setminus\mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)\mathbb{P}\left[x\notin\mathcal{D}_1\right]\right]\\
&=\sum_{x\in\mathcal{X}\setminus\mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)(1-\nu^{\pi_1}(x))^{N_1}\\
&\leq\frac{|\mathcal{X}\setminus\mathcal{X}_{\text{common}}|}{eN_1}.
\end{aligned}$$

Therefore, in conclusion, we get the bound

$$(1-\gamma)^2\mathbb{E}\left[\|J(\pi_1)-J(\hat{\pi}_1)\|_{\infty}\right]\leq\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty,\mathcal{X}_{\text{common}}}\frac{|\mathcal{X}_{\text{common}}|}{eN}+\frac{|\mathcal{X}\setminus\mathcal{X}_{\text{common}}|}{eN_1}.$$

Because the L_∞ norm bounds the maximum deviation across all dimensions (and $|J_i(\pi_1)-\mathbb{E}[J_i(\hat{\pi}_1)]|\leq\mathbb{E}\,|J_i(\pi_1)-J_i(\hat{\pi}_1)|$ by Jensen's inequality), our BC analysis guarantees that for all $i\in[d]$:

$$(1-\gamma)^2\mathbb{E}[J_i(\hat{\pi}_1)]+\left\|\frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}}\right\|_{\infty,\mathcal{X}_{\text{common}}}\frac{|\mathcal{X}_{\text{common}}|}{eN}+\frac{|\mathcal{X}\setminus\mathcal{X}_{\text{common}}|}{eN_1}\geq(1-\gamma)^2J_i(\pi_1).$$

Now, using the fact that π_1 is on the Pareto front, we know there does not exist any policy π^* that dominates it. Therefore, substituting our bound into the definition of the Pareto front, it holds that $\nexists\pi^*$ such that both of the following conditions are true simultaneously:

$$\forall i \in [d],\ \ J_i(\pi^*) \geq \mathbb{E}[J_i(\hat{\pi}_1)] + \left\| \frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} \frac{|\mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N} + \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N_1}$$

$$\exists j \in [d],\ \ J_j(\pi^*) > \mathbb{E}[J_j(\hat{\pi}_1)] + \left\| \frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} \frac{|\mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N} + \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N_1}.$$

□

B.4 Proof of Theorem 3.2

Before presenting the proof we formalize the algorithmic class under which the lower bound holds.

Expert-indexed output algorithms. An expert-indexed multi-output imitation-learning algorithm returns one policy $\hat{\pi}_\ell$ associated with each demonstrated expert π_ℓ. In the construction below, this association is enforced by requiring $\hat{\pi}_\ell$ to use the same root action as π_ℓ. The corresponding root action is revealed to the learner for free, and therefore does not contribute to the statistical difficulty of the instance.

Theorem 3.2. *Fix $\gamma \in [1/2, 1)$, $N_1, N_2 \geq 1$, $p_1, p_2 \in (0, 1/2]$, and an index $\ell \in \{1, 2\}$. For any offline expert-indexed multi-output imitation algorithm that receives*

$$\mathcal{D}_j = \left\{(X_n^j, A_n^j)\right\}_{n=1}^{N_j}, \qquad (X_n^j, A_n^j) \sim \mu^{\pi_j},$$

and outputs $\{\hat{\pi}_1, \hat{\pi}_2\}$, there exists a two-objective MOMDP and two deterministic PF-policies π_1, π_2 such that

$$\varepsilon_\ell \geq \Omega\left(\min\left\{\frac{p_\ell}{1-\gamma},\ C_{\pi_\ell}\frac{|\mathcal{X}_{\text{common}}|}{N(1-\gamma)^2}\right\} + \min\left\{\frac{1-p_\ell}{1-\gamma},\ \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{N_\ell(1-\gamma)^2}\right\}\right),$$

where ε_ℓ is any value for which $\hat{\pi}_\ell$ is an ε_ℓ-PF-policy and

$$C_{\pi_\ell} := \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}}.$$

Moreover, for the constructed MOMDP,

$$C_{\pi_\ell} = \frac{N p_\ell}{N_1 p_1 + N_2 p_2}.$$

The universal constant hidden in $\Omega(\cdot)$ is independent of $N_1, N_2, \gamma, p_1, p_2$, and the state-space cardinalities.

Proof intuition. We adapt the hard-instance idea of [38] using three recurrent reset blocks: one common block and two private blocks, as illustrated in Figure 6a. Expert ℓ enters the common block with probability p_ℓ and its own private block with probability $1 - p_\ell$. Only its own demonstrations reveal the correct actions in its private block, whereas demonstrations from both experts reveal the correct actions in the common block.

Because the observations are sampled from normalized discounted occupancy measures, the expected numbers of samples in these blocks are $\gamma N_\ell(1-p_\ell)$ and γM_{common}, respectively, where $M_{\text{common}} := N_1 p_1 + N_2 p_2$. The effect of an error in the common block on expert ℓ's return is also weighted by p_ℓ. The exact common-state concentrability coefficient is therefore

$$C_{\pi_\ell} = \frac{N p_\ell}{M_{\text{common}}}.$$

The recurrent reset makes an unobserved action a repeated failure hazard over the discounted horizon. In the regime before the return loss saturates, the resulting common and private contributions scale as

$$\frac{C_{\pi_\ell}\,|\mathcal{X}_{\text{common}}|}{N(1-\gamma)^2} \qquad \text{and} \qquad \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{N_\ell(1-\gamma)^2}.$$

The formal argument below retains the saturation terms in Theorem 3.2 and proves a lower bound for each fixed expert-indexed output. Two experts and two objectives suffice because this is a special case of the general problem class.

Construction of the lower-bound MOMDP. Fix $p_1, p_2 \in (0, 1/2]$ and define

$$M_{\text{common}} := N_1 p_1 + N_2 p_2.$$

Let $\beta := 1/4$. The state space is

$$\mathcal{X} = \{x_{\text{root}}, x_{\text{end}}\} \cup \mathcal{X}^{\text{c}} \cup \mathcal{X}^1_{\text{div}} \cup \mathcal{X}^2_{\text{div}},$$

where $\mathcal{X}^{\text{c}}$ is the common block and $\mathcal{X}^{\ell}_{\text{div}}$ is the private block associated with expert ℓ. Each block consists of a collection of hard states and one anchor state. The correct action at the anchor state is fixed and revealed to the learner; all statistical uncertainty is concentrated on the hard states.

The action set is $\mathcal{A} = \{a_1, a_2\}$. At the root state, the reward is zero and the transition kernel satisfies

$$P(\cdot \mid x_{\text{root}}, a_1) = p_1 \rho_{\text{c}} + (1 - p_1)\rho_1, \tag{2}$$

$$P(\cdot \mid x_{\text{root}}, a_2) = p_2 \rho_{\text{c}} + (1 - p_2)\rho_2, \tag{3}$$

where ρ_{c}, ρ_1, and ρ_2 are distributions supported on $\mathcal{X}^{\text{c}}$, $\mathcal{X}^1_{\text{div}}$, and $\mathcal{X}^2_{\text{div}}$, respectively.

Every hard state x is assigned a correct action $a^\star(x) \in \{a_1, a_2\}$. Under the Yao prior, these actions are independent and uniformly distributed. For a state x in a block $B \in \{\text{c}, 1, 2\}$, define

$$r(x, a^\star(x)) = u_B, \qquad P(\cdot \mid x, a^\star(x)) = \rho_B, \tag{4}$$

$$r(x, a) = \mathbf{0}, \quad a \neq a^\star(x), \qquad P(x_{\text{end}} \mid x, a) = 1. \tag{5}$$

The same transitions hold at the anchor state, except that its correct action is known to the learner. The reward vectors are

$$u_{\text{c}} = \begin{bmatrix} \beta \\ \beta \end{bmatrix}, \qquad u_1 = \begin{bmatrix} 1 \\ \beta \end{bmatrix}, \qquad u_2 = \begin{bmatrix} \beta \\ 1 \end{bmatrix}.$$

Finally, x_{end} is absorbing and has zero reward.

The two expert policies satisfy

$$\pi_1(x_{\text{root}}) = a_1, \qquad \pi_2(x_{\text{root}}) = a_2.$$

Both experts take $a^\star(x)$ on the common block. Expert ℓ takes $a^\star(x)$ on its own private block $\mathcal{X}^{\ell}_{\text{div}}$. On the private block that it never visits, each expert is defined to take the action opposite to $a^\star(x)$. The two policies take the same arbitrary action at x_{end}. Consequently,

$$\mathcal{X}_{\text{common}} = \mathcal{X}^{\text{c}} \cup \{x_{\text{end}}\},$$

whereas x_{root} and the two private blocks belong to $\mathcal{X} \setminus \mathcal{X}_{\text{common}}$.

The essential feature of the construction is the reset mechanism: after every correct action, the process returns to a fresh state drawn from the same block distribution. Thus, uncertainty about an unobserved action creates a failure hazard throughout the effective discounted horizon rather than at a single time step.

Lemma B.1 (Value gap in a recurrent reset block)**.** *Consider a block B with reset distribution ρ_B and reward vector u_B. Let $\pi^\star$ take $a^\star(x)$ at every $x \in B$. For an arbitrary policy $\widehat{\pi}$, define its failure probability under a fresh draw from the block as*

$$q_B(\widehat{\pi}) := \sum_{x \in B} \rho_B(x) \left(1 - \widehat{\pi}(a^\star(x) \mid x)\right).$$

Then, for every objective $i \in [2]$,

$$V_i^{\pi^\star}(B) = \frac{u_{B,i}}{1 - \gamma}, \tag{6}$$

$$V_i^{\widehat{\pi}}(B) = \frac{(1 - q_B(\widehat{\pi}))u_{B,i}}{1 - \gamma(1 - q_B(\widehat{\pi}))}, \tag{7}$$

and hence

$$V_i^{\pi^\star}(B) - V_i^{\widehat{\pi}}(B) = \frac{u_{B,i} q_B(\widehat{\pi})}{(1 - \gamma)\left(1 - \gamma + \gamma q_B(\widehat{\pi})\right)}. \tag{8}$$

In particular, if $\gamma q_B(\widehat{\pi}) \leq c_0(1 - \gamma)$, then

$$V_i^{\pi^\star}(B) - V_i^{\widehat{\pi}}(B) \geq \frac{u_{B,i}}{1 + c_0} \frac{q_B(\widehat{\pi})}{(1 - \gamma)^2}.$$

Proof. After every successful action, the next state is an independent draw from ρ_B. Therefore, at every live time step, the policy succeeds with probability $1 - q_B(\widehat{\pi})$, receives reward $u_{B,i}$, and otherwise moves to x_{end}. It follows that

$$V_i^{\widehat{\pi}}(B) = (1 - q_B(\widehat{\pi}))\left(u_{B,i} + \gamma V_i^{\widehat{\pi}}(B)\right).$$

Solving this equation gives the claimed expression for $V_i^{\widehat{\pi}}(B)$. Subtracting it from $u_{B,i}/(1-\gamma)$ gives Equation (8).

Finally, if $\gamma q_B(\widehat{\pi}) \le c_0(1-\gamma)$, then

$$1 - \gamma + \gamma q_B(\widehat{\pi}) \le (1 + c_0)(1 - \gamma),$$

which proves the final claim. □

Proof of Theorem 3.2. Fix an arbitrary index $\ell \in \{1, 2\}$. The hard instance may depend on this fixed index, as is standard for a per-output minimax lower bound.

Step 1: the expert policies lie on the Pareto front. Consider first a policy that chooses a_1 at x_{root}. Replacing its actions on $\mathcal{X}^{\mathrm{c}} \cup \mathcal{X}^1_{\mathrm{div}}$ by the correct actions $a^\star(x)$ weakly improves both return coordinates and strictly improves both whenever the original policy has a positive failure probability on a reachable block. Therefore, among policies choosing a_1 at the root, π_1 is the unique nondominated policy on the reachable states. The analogous statement holds for π_2 among policies choosing a_2 at the root.

The expert returns are

$$J(\pi_1) = \frac{\gamma}{1-\gamma}\left(p_1 u_{\mathrm{c}} + (1 - p_1)u_1\right), \tag{9}$$

$$J(\pi_2) = \frac{\gamma}{1-\gamma}\left(p_2 u_{\mathrm{c}} + (1 - p_2)u_2\right). \tag{10}$$

Since $\beta = 1/4$ and $p_1, p_2 \le 1/2$, the first coordinate of $J(\pi_1)$ is strictly larger than the first coordinate of $J(\pi_2)$, whereas the second coordinate of $J(\pi_2)$ is strictly larger than the second coordinate of $J(\pi_1)$. Hence neither expert dominates the other.

More generally, take any policy and preserve its randomization at the root while replacing all of its reachable block actions by the corresponding correct actions. The resulting policy has a return in $\mathrm{conv}\{J(\pi_1), J(\pi_2)\}$ and weakly dominates the original policy. Consequently, π_1 and π_2 are deterministic PF-policies.

Step 2: occupancy measures and common-state concentrability. Under expert j, the normalized discounted occupancy of $x \in \mathcal{X}^{\mathrm{c}}$ is

$$\nu^{\pi_j}(x) = (1-\gamma)\sum_{t=1}^{\infty} \gamma^t p_j \rho_{\mathrm{c}}(x) \tag{11}$$

$$= \gamma p_j \rho_{\mathrm{c}}(x). \tag{12}$$

It follows that

$$\nu^{\pi_{\mathrm{data}}}(x) = \gamma \rho_{\mathrm{c}}(x)\frac{N_1 p_1 + N_2 p_2}{N}.$$

Therefore, for every common state with positive pooled occupancy,

$$\frac{\nu^{\pi_\ell}(x)}{\nu^{\pi_{\mathrm{data}}}(x)} = \frac{N p_\ell}{N_1 p_1 + N_2 p_2}.$$

The ratio is constant over the common block, and hence

$$C_{\pi_\ell} = \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\mathrm{data}}}} \right\|_{\infty, \mathcal{X}_{\mathrm{common}}} = \frac{N p_\ell}{N_1 p_1 + N_2 p_2}. \tag{13}$$

Notice that on a private state $x \in \mathcal{X}^\ell_{\mathrm{div}}$, the corresponding full-space occupancy ratio is N/N_ℓ. This is why the pooled term uses a concentrability coefficient restricted to $\mathcal{X}_{\mathrm{common}}$, whereas the private term is controlled directly by N_ℓ.

Step 3: choice of the rare-state probabilities. Let n_{c} denote the number of hard states in $\mathcal{X}^{\mathrm{c}}$ and let n_{d} denote the number of hard states in each private block. Since each block contains only one additional anchor state,

$$n_{\mathrm{c}} = \Theta\left(|\mathcal{X}_{\mathrm{common}}|\right), \qquad n_{\mathrm{d}} = \Theta\left(|\mathcal{X} \setminus \mathcal{X}_{\mathrm{common}}|\right),$$

where the hidden constants are universal.

For a sufficiently small universal constant $c_0 \in (0, 1/16]$, define

$$\eta_{\mathrm{c}} := c_0 \min\left\{\frac{1}{\gamma M_{\mathrm{common}}}, \frac{1-\gamma}{\gamma n_{\mathrm{c}}}\right\}, \tag{14}$$

$$\eta_{\ell} := c_0 \min\left\{\frac{1}{\gamma(1-p_{\ell})N_{\ell}}, \frac{1-\gamma}{\gamma n_{\mathrm{d}}}\right\}. \tag{15}$$

Assign probability η_{c} to each hard state under ρ_{c} and probability η_{ℓ} to each hard state in $\mathcal{X}^{\ell}_{\mathrm{div}}$ under ρ_{ℓ}. Place all remaining probability on the corresponding anchor state. The analogous distribution on the other private block can be defined symmetrically.

These are valid probability distributions because the total probability assigned to the hard states is at most $c_0(1-\gamma)/\gamma < 1$. Moreover,

$$\gamma n_{\mathrm{c}} \eta_{\mathrm{c}} \le c_0(1-\gamma), \qquad \gamma n_{\mathrm{d}} \eta_{\ell} \le c_0(1-\gamma). \tag{16}$$

Step 4: unobserved states imply unavoidable action error. For each block B, let $\overline{q}_B(\widehat{\pi}_{\ell})$ denote the contribution of only the hard states to $q_B(\widehat{\pi}_{\ell})$. Then

$$q_B(\widehat{\pi}_{\ell}) \ge \overline{q}_B(\widehat{\pi}_{\ell}),$$

and, by Equation (16),

$$\gamma \overline{q}_B(\widehat{\pi}_{\ell}) \le c_0(1-\gamma),$$

regardless of the learner's action at the anchor state.

Fix a hard state $x \in \mathcal{X}^{\mathrm{c}}$. Its normalized occupancy under expert j is $\gamma p_j \eta_{\mathrm{c}}$. Since $\mathcal{D}_j$ contains N_j independent samples from μ^{π_j}, the probability that x is absent from all datasets is

$$\mathbb{P}\left[x \notin \bigcup_{j=1}^{2} \mathcal{D}_j\right] = \prod_{j=1}^{2} \left(1 - \gamma p_j \eta_{\mathrm{c}}\right)^{N_j} \tag{17}$$

$$\ge 1 - \gamma \eta_{\mathrm{c}} \left(N_1 p_1 + N_2 p_2\right) \tag{18}$$

$$\ge 1 - c_0. \tag{19}$$

The first inequality follows by upper-bounding the probability that the state appears at least once by the sum of its appearance probabilities.

Conditional on x being absent, its hidden action remains uniform on $\{a_1, a_2\}$ under the Yao prior. Therefore, every learner satisfies

$$\mathbb{E}\left[1 - \widehat{\pi}_{\ell}(a^{\star}(x) \mid x) \,\middle|\, x \notin \bigcup_{j=1}^{2} \mathcal{D}_j\right] \ge \frac{1}{2}.$$

Summing over the hard common states gives

$$\mathbb{E}\left[\overline{q}_{\mathrm{c}}(\widehat{\pi}_{\ell})\right] \ge \frac{1-c_0}{2} n_{\mathrm{c}} \eta_{\mathrm{c}}. \tag{20}$$

Similarly, a hard state $x \in \mathcal{X}^{\ell}_{\mathrm{div}}$ is visited only by expert ℓ and has occupancy $\gamma(1-p_{\ell})\eta_{\ell}$. Hence

$$\mathbb{P}\left[x \notin \mathcal{D}_{\ell}\right] = \left(1 - \gamma(1-p_{\ell})\eta_{\ell}\right)^{N_{\ell}} \tag{21}$$

$$\ge 1 - c_0. \tag{22}$$

The demonstrations of the other expert contain no information about the hidden actions in $\mathcal{X}^{\ell}_{\mathrm{div}}$. Consequently,

$$\mathbb{E}\left[\overline{q}_{\ell}(\widehat{\pi}_{\ell})\right] \ge \frac{1-c_0}{2} n_{\mathrm{d}} \eta_{\ell}. \tag{23}$$

Step 5: conversion of action error into return loss. The expert-indexing condition fixes the root action of $\widehat{\pi}_\ell$ to be a_ℓ. Define

$$f_B(q) := \frac{u_{B,i} q}{(1-\gamma)(1-\gamma+\gamma q)}.$$

This function is nondecreasing in q. Since $q_B(\widehat{\pi}_\ell) \geq \overline{q}_B(\widehat{\pi}_\ell)$, the exact expression in Equation (8) gives

$$V_i^{\pi^\star}(B) - V_i^{\widehat{\pi}_\ell}(B) \geq f_B\left(\overline{q}_B(\widehat{\pi}_\ell)\right).$$

The small-hazard condition in Equation (16) can therefore be applied to $\overline{q}_B(\widehat{\pi}_\ell)$. For every objective $i \in [2]$, we obtain

$$J_i(\pi_\ell) - \mathbb{E}\left[J_i(\widehat{\pi}_\ell)\right] \geq \frac{\gamma}{1+c_0} \frac{1}{(1-\gamma)^2} \Big(p_\ell u_{\mathrm{c},i} \mathbb{E}\left[\overline{q}_{\mathrm{c}}(\widehat{\pi}_\ell)\right] + (1-p_\ell) u_{\ell,i} \mathbb{E}\left[\overline{q}_\ell(\widehat{\pi}_\ell)\right] \Big) \tag{24}$$

$$\geq c_1 \frac{\gamma}{(1-\gamma)^2} \left(p_\ell n_{\mathrm{c}} \eta_{\mathrm{c}} + (1-p_\ell) n_{\mathrm{d}} \eta_\ell\right), \tag{25}$$

where $c_1 > 0$ is a universal constant, since every coordinate of u_{c}, u_1, and u_2 is at least $\beta = 1/4$.

Substituting the definitions of η_{c} and η_ℓ and using $\gamma \geq 1/2$ gives

$$J_i(\pi_\ell) - \mathbb{E}\left[J_i(\widehat{\pi}_\ell)\right] \geq \Omega\Bigg(\min\left\{ \frac{p_\ell}{1-\gamma}, \frac{p_\ell n_{\mathrm{c}}}{M_{\mathrm{common}}(1-\gamma)^2} \right\} + \min\left\{ \frac{1-p_\ell}{1-\gamma}, \frac{n_{\mathrm{d}}}{N_\ell (1-\gamma)^2} \right\} \Bigg). \tag{26}$$

By Equation (13),

$$\frac{p_\ell}{M_{\mathrm{common}}} = \frac{C_{\pi_\ell}}{N}.$$

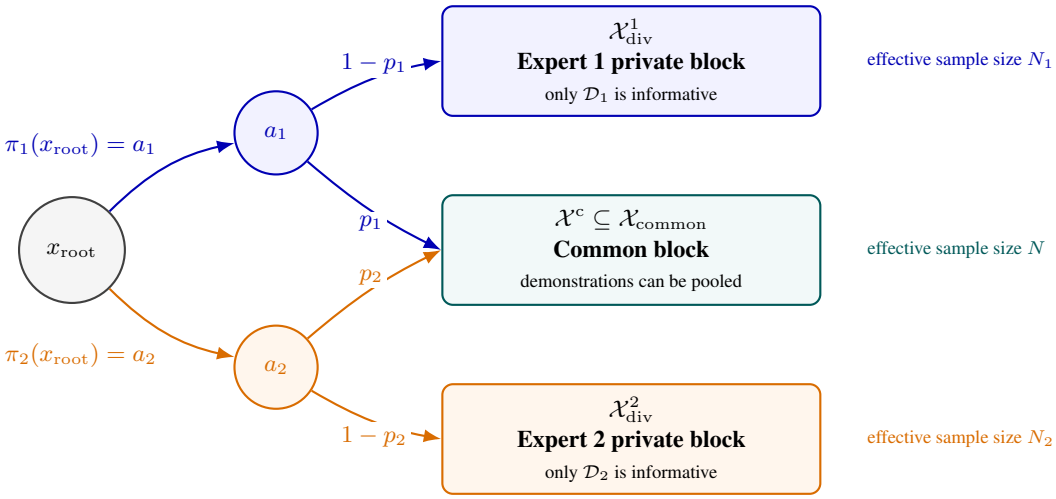
Replacing n_{c} and n_{d} by the corresponding state-set cardinalities up to universal constants yields

$$J_i(\pi_\ell) - \mathbb{E}\left[J_i(\widehat{\pi}_\ell)\right] \geq \Omega\left(\min\left\{ \frac{p_\ell}{1-\gamma}, C_{\pi_\ell} \frac{|\mathcal{X}_{\mathrm{common}}|}{N(1-\gamma)^2} \right\} + \min\left\{ \frac{1-p_\ell}{1-\gamma}, \frac{|\mathcal{X} \setminus \mathcal{X}_{\mathrm{common}}|}{N_\ell (1-\gamma)^2} \right\} \right).$$

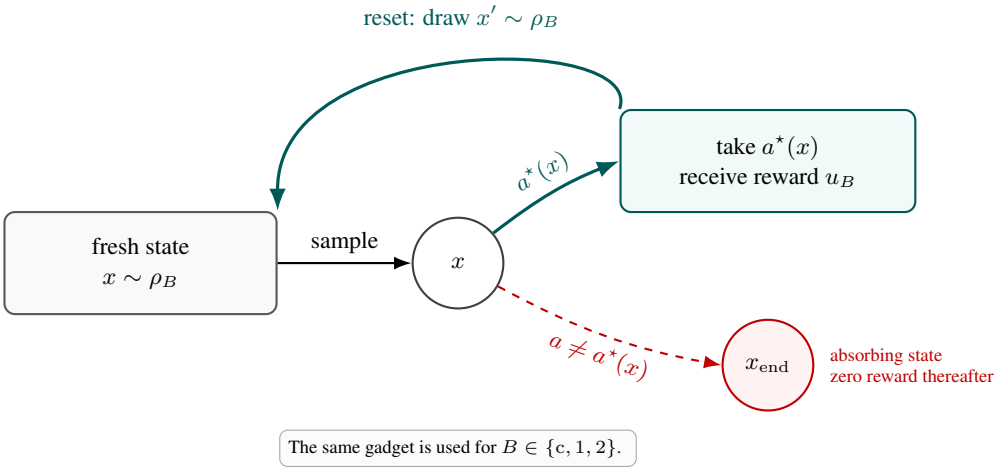
Step 6: conversion to a deterministic hard instance and Pareto-front error. The preceding argument gives a Bayes lower bound under the independent uniform prior over the hidden actions. By Yao's minimax principle, there exists a deterministic realization of the hidden actions for which the same lower bound, up to a universal constant, holds for the fixed output index ℓ.

Moreover, Equation (25) holds for both objectives. Thus, the expert policy π_ℓ improves every coordinate of $\mathbb{E}[J(\widehat{\pi}_\ell)]$ by the displayed amount. According to the definition of an ε_ℓ-PF-policy, $\widehat{\pi}_\ell$ cannot be an ε_ℓ-PF-policy for any smaller value of ε_ℓ. This proves Theorem 3.2. □

Why the reset mechanism recovers the matching rate. In the previous one-shot construction, an error at a hard state could occur only once, which lost one factor of $(1-\gamma)^{-1}$. In the construction above, every correct action returns the process to a fresh state from the same hard distribution. The learner therefore faces an approximately constant failure hazard at every step of the effective discounted horizon. The denominator in Equation (8) consequently produces the additional factor of $(1-\gamma)^{-1}$, yielding the matching $(1-\gamma)^{-2}$ dependence in both the common-state and expert-specific terms.



(a) Root routing and the common–divergent decomposition.



(b) The hard construction inside each block from [38].

Figure 6: **Recurrent hard-state construction for the matching lower bound. (a)** Expert π_ℓ enters the common block with probability p_ℓ and its private block with probability $1 - p_\ell$. The common block is informed by all N samples, whereas the private block of expert ℓ is informed only by its own N_ℓ samples. **(b)** The hard-state gadget is adapted from the standard imitation-learning lower-bound construction of [38]. Inside each block, taking the correct action yields reward u_B and resets the process to a fresh state drawn from the same block distribution ρ_B, while an incorrect action transitions to the absorbing zero-reward state x_{end}. The recurrent reset is the modification that produces the matching $(1-\gamma)^{-2}$ dependence.

C MA-BC with Stochastic Experts

In the main text, we assumed deterministic experts for ease of exposition. Here, we formally discuss how MA-BC handles stochastic experts.

Because the return space $\mathbb{J}$ of an MOMDP forms a convex polytope [26], any stochastic expert residing on the Pareto front can be viewed as a convex mixture of deterministic vertex policies. Crucially, a single stochastic expert on the Pareto front maximizes a specific scalarized reward function defined by some weight vector $w \in \mathbb{R}^d_+$. If this expert selects among multiple actions at a given state, it implies that all such actions are optimal with respect to w.

When the expert identities are known (the split data setting of Algorithm 1), intra-expert stochasticity poses no issue. Because all actions played by a single stochastic expert at a given state are optimal under the same scalarization w, the Markov property ensures that any policy independently selecting among these actions at each state remains optimal for w (as leveraged in the proof of Theorem 2.1). Consequently, mixing or averaging these actions at the state level does not cause the policy's expected return to fall off the Pareto front. Standard MA-BC (Algorithm 1) can safely pool these shared stochastic actions alongside other experts' non-conflicting data, perfectly recovering a policy on the same optimal face of the Pareto front.

We note that the stochastic expert setting can also be analyzed directly. Using standard behavioral cloning analyses based on the logarithmic loss [10], one could establish performance bounds for MA-BC (similar to Theorem 3.1) that accommodate stochastic expert policies by bounding the statistical discrepancy (e.g., via Kullback-Leibler divergence or Total Variation distance) between the learned and expert action distributions. However, we opt to maintain the deterministic expert assumption in our main theoretical presentation for simplicity.

D MA-BC with Unsplit Expert Data

In the standard setting of Multi-Output Augmented Behavioral Cloning (MA-BC) presented in Section 3, we assume the dataset $\mathcal{D}$ is already partitioned by expert identity (i.e., we know which trajectories belong to which expert). We now generalize our analysis to the fully unsplit data setting. We consider a mixed dataset generated by L latent deterministic expert policies that differ at a set of distinct states, denoted by the divergence set $\mathcal{X}_{\text{div}} = \mathcal{X} \setminus \mathcal{X}_{\text{common}}$. We assume the expert identities are hidden and $\mathcal{X}_{\text{div}}$ is initially unknown. Because MA-BC relies fundamentally on state-level data partitioning, we can approach this problem by directly identifying where the experts disagree. We identify the divergent states $\mathcal{X}_{\text{div}}$ by scanning the mixed dataset for states where multiple distinct actions are observed. Once $\mathcal{X}_{\text{div}}$ is identified, we can safely isolate the common data $\mathcal{D}_{\text{common}}$ where all experts agree.

However, within the diverging states, we face the *Binding Problem*: we know *where* the experts disagree, but we do not know *which* expert takes *which* set of actions. To resolve this, we must construct a consistent set of divergent actions for each expert. We achieve this by leveraging the transitivity of trajectories. If a single trajectory visits x_1 and takes action a_1, and later visits x_2 taking action a_2, we can bind the state-action pairs (x_1, a_1) and (x_2, a_2) to the same latent expert.

We formalize this using a **Consistency Graph**, visualized in Figure 7. Each unique divergent state-action pair (x, a) observed in $\mathcal{X}_{\text{div}}$ is a node. An edge connects (x_i, a_i) and (x_j, a_j) if there exists at least one trajectory $\tau \in \mathcal{D}$ that visits both states and executes those specific actions. The connected components of this graph naturally cluster the divergent state-actions into consistent, expert-specific sets $\mathcal{D}_{\text{div},\ell}$. This state-level partitioning algorithm is formalized in Algorithm 2.

Assumption (Pairwise Distinct Actions) To guarantee that the Consistency Graph perfectly isolates L latent experts when $L \geq 3$, we assume that all experts take mutually exclusive actions at the divergent states. Formally, for any divergent state $x \in \mathcal{X}_{\text{div}}$ and any two distinct experts $\ell, \ell' \in [L]$, it holds that $\pi_\ell(x) \neq \pi_{\ell'}(x)$. This structural assumption prevents a scenario where two distinct experts share a single node (x, a_{shared}) in the graph, which would otherwise merge their components and reintroduce **Failure II**. For the two-expert case ($L = 2$), this assumption is satisfied by definition, as a state is only flagged as divergent if the two experts disagree. We highlight that this assumption can be removed at the cost of building a consistency graph where each node is the sequence of visited divergent states in a single trajectory. Two nodes would then be connected if they share at least one divergent state and they do not contradict at any other divergent state. However, instead of graph partitioning, one would need to extract Maximal Cliques [27] in the graph, which is an NP-hard problem. Thus, we restrict our setting to pair-wise distinct actions, and leave the extension to arbitrary experts for future work.

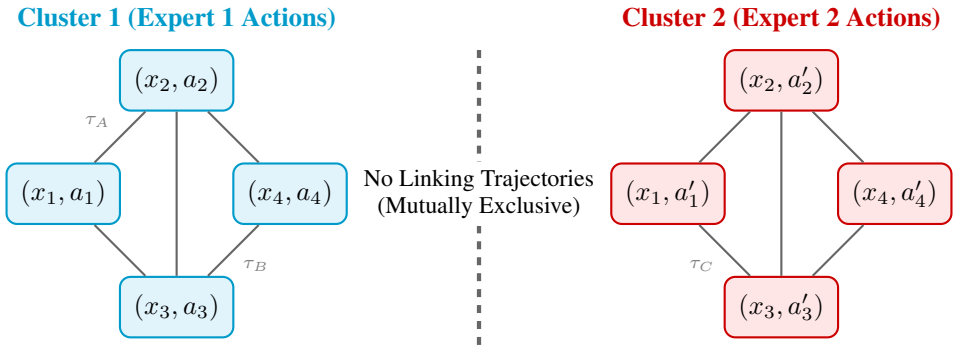


Figure 7: **Action Consistency Graph on $\mathcal{X}_{\text{div}}$.** The nodes represent the conflicting state-action pairs observed at the divergent states. Edges connect actions that were observed co-occurring within the same trajectory. The graph partitions the conflicting actions into consistent, expert-specific datasets ($\mathcal{D}_{\text{div},\ell}$), successfully binding the latent experts' actions on the divergent set of states.

Algorithm 2 State-Level MA-BC for Unsplit Expert Data

Require: Mixed Dataset $\mathcal{D} = \{\tau_i\}_{i=1}^N$
1. Identify Diverging & Common States:
$\mathcal{X}_{\text{div}} \leftarrow \{x \in \mathcal{X} \mid \exists (x, a), (x, a') \in \mathcal{D} \text{ such that } a \neq a'\}$
$\mathcal{D}_{\text{common}} \leftarrow \{(x, a) \in \mathcal{D} \mid x \notin \mathcal{X}_{\text{div}}\}$
$\mathcal{D}_{\text{div}} \leftarrow \{(x, a) \in \mathcal{D} \mid x \in \mathcal{X}_{\text{div}}\}$
2. Build Action Consistency Graph:
Construct graph G where nodes are the unique state-action pairs $v = (x, a) \in \mathcal{D}_{\text{div}}$.
Add an edge (v_i, v_j) if $\exists \tau \in \mathcal{D}$ that visits both v_i and v_j.
3. Partition Divergent Actions:
Find the connected components $C_1, \ldots, C_L$ of G.
Let $\mathcal{D}_{\text{div},\ell} \leftarrow \{(x, a) \in \mathcal{D}_{\text{div}} \mid (x, a) \in C_\ell\}$
4. Train Experts (MA-BC):
for $\ell = 1, \ldots, L$ **do**
    $\hat{\pi}_\ell = \arg\max_{\pi \in \Pi} \sum_{X,A \in \mathcal{D}_{\text{common}} \cup \mathcal{D}_{\text{div},\ell}} \log \pi(A|X)$
end for
Output: $\{\hat{\pi}_\ell\}_{\ell=1}^L$

D.1 Upper Bound for Unsplit MA-BC

We now extend Theorem 3.1 to the unsplit setting. We assume that each of the N independent observations is generated by first selecting latent expert ℓ with probability $\alpha_\ell > 0$, where $\sum_{\ell=1}^L \alpha_\ell = 1$, and then sampling a state-action pair from μ^{π_ℓ}. Thus, the state marginal is $\nu^{\pi_{\text{data}}} = \sum_{\ell=1}^L \alpha_\ell \nu^{\pi_\ell}$, and $N_\ell := N\alpha_\ell$ is the expected size of the latent subset $\mathcal{D}_\ell$. Furthermore, we assume the auxiliary mechanism connecting divergent states observes links as independent events during the data collection process.

For the unsplit algorithm to successfully match the statistical rate of standard MA-BC, the consistency graph must perfectly bind the divergent state-actions to their respective experts. This identifiability relies on observing "bridge" samples that link the diverging states. Let the diverging states be $\mathcal{X}_{\text{div}} = \{x_1, \ldots, x_K\}$. It suffices to observe the set of pairwise links $\{(x_k, x_{k+1})\}_{k=1}^{K-1}$, where a link implies an independent sampling event captures an association between x_k and x_{k+1}.

We seek to explicitly bound the probability that the Consistency Graph fails to fully connect the divergent actions of a target expert. Let the set of diverging states be ordered as an arbitrary sequence $\mathcal{X}_{\text{div}} = \{x_1, x_2, \ldots, x_K\}$. To uniquely identify an expert's actions across all K states, it is sufficient to observe a "chain" of connected components. Specifically, we require the presence of $K - 1$ pairwise links: $(x_1, x_2), (x_2, x_3), \ldots, (x_{K-1}, x_K)$. A link (x_i, x_{i+1}) is considered *present* in the dataset $\mathcal{D}$ if there exists at least one independent sampling event $e \in \mathcal{D}$ that bridges both states. Let E_i be the event that a *single* sampling event from the mixed distribution bridges both x_i and x_{i+1}. We define p_{link} as the worst-case probability of any such link occurring in a single independent sampling event:

$$\mathbb{P}(E_i) \geq \min_{j \in \{1,\ldots,K-1\}} \mathbb{P}_{e \sim \nu^{\pi_{\text{data}}}}(x_j \text{ and } x_{j+1} \text{ linked in } e) := p_{\text{link}}$$

Let F_i be the event that the link (x_i, x_{i+1}) is completely missing from the entire dataset of N independent samples. Because the N samples are drawn independently and identically, the probability that all N samples fail to contain this link is:

$$\begin{aligned} \mathbb{P}(F_i) &= (1 - \mathbb{P}(E_i))^N \\ &\leq (1 - p_{\text{link}})^N. \end{aligned}$$

The global identification process fails if **at least one** of the $K - 1$ necessary links is missing from the dataset. Let F_{global} be this global failure event: $F_{\text{global}} = \bigcup_{i=1}^{K-1} F_i$. By the Union Bound, we bound the global failure probability as follows:

$$\mathbb{P}\left(\bigcup_{i=1}^{K-1} F_i\right) \leq \sum_{i=1}^{K-1} \mathbb{P}(F_i)$$
$$\leq \sum_{i=1}^{K-1} (1-p_{\text{link}})^N$$
$$= (K-1)(1-p_{\text{link}})^N.$$

Thus, the probability that the consistency graph is "broken" (failing to fully connect the expert's divergent actions) is bounded from above by:

$$\mathbb{P}[\text{Consistency Graph Broken}] \leq (K-1)(1-p_{\text{link}})^N$$

We will leverage this failure probability in the following theorem to demonstrate that Unsplit MA-BC achieves the exact same suboptimality bound as Theorem 3.1, offset only by an exponentially fast-decaying identification penalty, dependent on p_{link}.

Theorem D.1. *Let $\{\pi_\ell\}_{\ell=1}^L$ be Pareto front policies and let $\{\hat{\pi}_\ell\}_{\ell=1}^L$ be the output of Algorithm 2 on an unsplit dataset of N independent samples. Let N_ℓ be the expected number of independent state-action samples generated by expert ℓ, and let $K = |\mathcal{X} \setminus \mathcal{X}_{\text{common}}|$. Then, for all $\ell \in [L]$, we have:*

$$\nexists \pi^\star \mid \forall i \in [d] \quad J_i(\pi^\star) - \mathbb{E}\left[J_i(\hat{\pi}_\ell)\right] \geq \left\| \frac{\nu^{\pi_\ell}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} \frac{|\mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N}$$
$$+ \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{e(1-\gamma)^2 N_\ell} + \frac{\nu_{\text{div}}^{\text{total}}(K-1)(1-p_{\text{link}})^N}{(1-\gamma)^2}$$

where $\nu_{\text{div}}^{\text{total}} = \sum_{x \in \mathcal{X} \setminus \mathcal{X}_{\text{common}}} \nu^{\pi_\ell}(x)$ is the occupancy mass on the divergent states and p_{link} is defined as above.

Proof. Without loss of generality, we consider $\ell = 1$. Following the exact decomposition from the proof of Theorem 3.1, we bound the L_∞ error of policy $\hat{\pi}_1$ across the state space:

$$(1-\gamma)^2 \left\| J(\pi_1) - J(\hat{\pi}_1) \right\|_\infty \leq \underbrace{\sum_{x \in \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x) \sum_{a \in \mathcal{A}} \pi_1(a|x) \mathbb{1}\left\{a \neq \hat{\pi}_1(x)\right\}}_{\text{Term I: Shared States}}$$
$$+ \underbrace{\sum_{x \in \mathcal{X} \setminus \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x) \sum_{a \in \mathcal{A}} \pi_1(a|x) \mathbb{1}\left\{a \neq \hat{\pi}_1(x)\right\}}_{\text{Term II: Divergent States}}.$$

Because the algorithm seamlessly isolates $\mathcal{D}_{\text{common}}$ (where no conflict exists), Term I is entirely unaffected by the latent expert identities. Applying the change of measure exactly as in Theorem 3.1, the error on the shared states is bounded purely by the missing mass on the total dataset N:

$$\mathbb{E}[\text{Term I}] \leq \left\| \frac{\nu^{\pi_1}}{\nu^{\pi_{\text{data}}}} \right\|_{\infty, \mathcal{X}_{\text{common}}} \frac{|\mathcal{X}_{\text{common}}|}{eN}.$$

For Term II (the divergent states), the learner $\hat{\pi}_1$ can fail for two reasons: (1) *Coverage Error*: the specific state x was never visited by expert 1 in the independently sampled dataset, or (2) *Identification Error*: the state was visited, but the Consistency Graph was broken, causing the state-action pair to be discarded or misassigned. We upper-bound the probability of failure by the sum of the coverage and identification failure probabilities:

$$
\begin{aligned}
\mathbb{E}[\text{Term II}] &\leq \sum_{x \in \mathcal{X} \setminus \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x) \left(\mathbb{P}[x \notin \mathcal{D}_1] + \mathbb{P}[\text{Consistency Graph Broken}] \right) \\
&\leq \sum_{x \in \mathcal{X} \setminus \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x) e^{-N_1 \nu^{\pi_1}(x)} + \sum_{x \in \mathcal{X} \setminus \mathcal{X}_{\text{common}}} \nu^{\pi_1}(x)(K-1)(1-p_{\text{link}})^N \\
&\leq \frac{|\mathcal{X} \setminus \mathcal{X}_{\text{common}}|}{eN_1} + \nu_{\text{div}}^{\text{total}}(K-1)(1-p_{\text{link}})^N,
\end{aligned}
$$

The first inequality is the union bound. For the second, we use the graph bound and

$$
\mathbb{P}[x \notin \mathcal{D}_1] = (1 - \alpha_1 \nu^{\pi_1}(x))^N \leq e^{-N_1 \nu^{\pi_1}(x)}.
$$

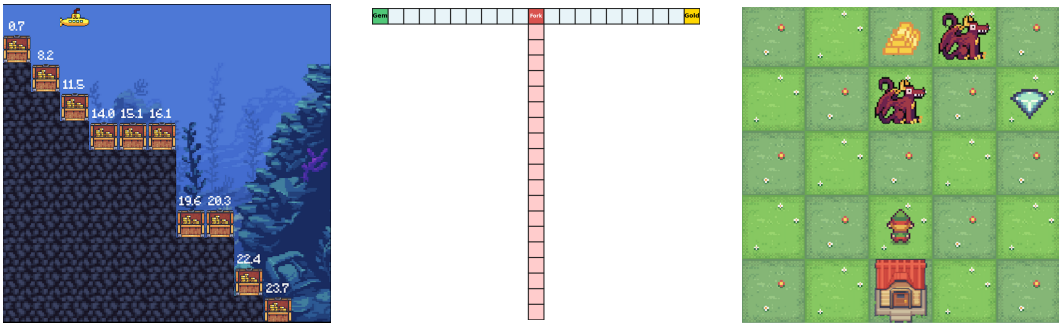
The final step uses $ve^{-N_1 v} \leq 1/(eN_1)$ for $v \geq 0$. Summing the terms and dividing by $(1-\gamma)^2$ establishes the expected L_∞ bound on the return. This confirms that Unsplit MA-BC achieves the optimal statistical rate $\mathcal{O}(1/N_\ell + 1/N)$ up to an exponentially vanishing identification penalty. $\square$

It remains open whether the dependence on p_{link} is necessary, and we highlight that providing a lower bound in the latent experts setting remains as an exciting future direction.

E Experimental Details

E.1 Environment Details

In this section, we provide a detailed description of the state spaces, action spaces, transition dynamics, and multi-objective reward structures for all the domains used in our empirical evaluation in Section 5. All experiments were run on a desktop CPU with a negligible runtime.



(a) Deep Sea Treasure (b) Slippery Y-Maze (c) Resource Gathering

Figure 8: Overview of the discrete multi-objective environments used for empirical evaluation. **(a)** Deep Sea Treasure requires balancing the treasure value against a time penalty. **(b)** The Slippery Y-Maze introduces a stochastic corridor where the agent must navigate, and choose between the gem and the gold. **(c)** In the Resource Gathering environment, the agent receives a 3-D reward signal regarding collecting the gold, the gem and avoiding the enemies.

E.1.1 Deep Sea Treasure

Deep Sea Treasure [63, 53] is a standard, widely used benchmark in multi-objective reinforcement learning. The environment consists of a discrete grid-world where a submarine agent navigates to find treasures located on the sea floor.

- **State Space:** The discrete (x, y) coordinates of the submarine.
- **Action Space:** Four discrete directional actions: Up, Down, Left, and Right.
- **Reward Vector:** A 2-dimensional vector $r(x, a) = [r_{\text{treasure}}, r_{\text{time}}]^T$. The agent receives a time penalty of $r_{\text{time}} = -1$ for every step taken. The treasure reward r_{treasure} is strictly positive only when the agent reaches a terminal treasure state. Treasures located further from the start yield exponentially higher values, creating a strictly convex Pareto front that explicitly trades off time efficiency against total treasure value.

E.1.2 Slippery Y-Maze

The Slippery Y-Maze is a discrete grid-world environment designed to introduce severe stochastic bottlenecks and spatial aliasing, testing the robustness of multi-objective imitation learning algorithms under risk.

State Space The state space $\mathcal{X}$ consists of a 2D coordinate grid shaped as a 'Y', plus a dedicated absorbing terminal state x_{term}. The grid is divided into three distinct regions: the stem, the left branch (Gem path), and the right branch (Gold path). We use the parameters ($L_{\text{stem}} = 20$, $L_{\text{branch}} = 10$) in our experiments. The agent spawns according to a uniform initial state distribution over all non-terminal states.

Action Space The agent has four discrete directional actions $\mathcal{A} = \{\text{Up}, \text{Down}, \text{Left}, \text{Right}\}$. We denote the deterministic coordinate shift associated with an action a as Δ_a (e.g., $\Delta_{\text{Up}} = (0, 1)$).

Transition Dynamics The environment introduces a stochastic bottleneck in the stem to punish dithering. While the agent is in the stem (S_{stem}), every action carries a 10% ambient risk of "falling off" or dying, which transitions the agent immediately to x_{term} with zero reward. Once the agent reaches the branches, the ambient risk drops to zero and transitions become deterministic.

Formally, the transition probabilities $P(x' \mid x, a)$ are defined as follows. Let $x_{\text{next}} = x + \Delta_a$. If $x_{\text{next}} \notin \mathcal{X}$, the agent collides with a wall and remains in its current state (i.e., $x_{\text{next}} = x$).

- **In the Stem ($x \in S_{\text{stem}}$):**

$$P(x_{\text{term}} \mid x, a) = 0.10$$
$$P(x_{\text{next}} \mid x, a) = 0.90$$

- **In the Branches ($x \in S_{\text{left}} \cup S_{\text{right}}$):**

$$P(x_{\text{next}} \mid x, a) = 1.0$$

When the agent steps off the ends of the branches (coordinates $(10, 19)$ or $(-10, 19)$), it transitions deterministically to x_{term} and the episode ends.

Reward Vector The reward function $r(x, a, x') \in \mathbb{R}^2$ represents a trade-off between two objectives: [Gold, Gem]. The environment yields zero reward everywhere except on the transitions exiting the extremities of the two branches:

$$r((10, 19), a, x_{\text{term}}) = [1.0, 0.0]^T \quad \text{(Gold Expert Target)}$$
$$r((-10, 19), a, x_{\text{term}}) = [0.0, 1.0]^T \quad \text{(Gem Expert Target)}$$

All other transitions yield $[0.0, 0.0]^T$. The discount factor is set to $\gamma = 0.9999$ to simulate an effectively undiscounted shortest-path problem.

Expert Policies By finding the exact continuous Pareto frontier using Optimistic Linear Support (OLS), we extract two deterministic target experts at the extremes of the frontier:

1. **Gold Expert (π_A^*):** Always routes right at the central fork $(0, 19)$ to secure the $[1.0, 0.0]^T$ reward.
2. **Gem Expert (π_B^*):** Always routes left at the central fork $(0, 19)$ to secure the $[0.0, 1.0]^T$ reward.

Crucially, because both experts want to minimize time spent in the dangerous stem region (to avoid the 10% ambient death probability), they share the exact same optimal policy for all states in S_{stem} (always moving Up). The only point of divergence occurs at the fork $(0, 19)$. If an imitation learner naively pools these experts (Naive BC), the conflicting actions at the fork will cause the agent to dither, repeatedly triggering the 10% death penalty and falling entirely off the Pareto frontier.

E.1.3 Resource Gathering

Resource Gathering tests the scalability of MA-BC by extending the conflict into a 3-dimensional objective space, moving beyond simple 2D trade-offs.

- **State Space:** A discrete grid containing multiple spawn points for two distinct types of resources (Gold and Gems) as well as roaming enemies.
- **Action Space:** Four discrete directional actions.
- **Reward Vector:** A 3-dimensional vector $r(x, a) = [r_{\text{gold}}, r_{\text{gem}}, r_{\text{enemy}}]^T$. The agent receives $+1$ on the respective channel for collecting a Gold or Gem, and a severe -1 penalty on the third channel if it encounters an enemy. The Pareto-optimal experts in our dataset demonstrate distinct, specialized behaviors: one strictly prioritizes gathering Gold, one exclusively hunts Gems, and one exhibits risk-averse behavior by prioritizing enemy avoidance over resource collection.

E.1.4 LQR Drone Control

In our continuous-control experiments, we model a linearized 6 Degrees of Freedom (6-DOF) quadcopter. The continuous-time linear time-invariant (LTI) system is defined as $\dot{x} = A_c x + B_c u$.

State and Action Space The state vector $x \in \mathbb{R}^{12}$ consists of the Euler angles (roll, pitch, yaw), the 3D position, the body angular rates, and the linear velocities:

$$x = [\varphi, \theta, \psi, X, Y, Z, p, q, r, u, v, w]^T$$

The action vector $u \in \mathbb{R}^4$ consists of the vertical thrust and the torques applied around the three body axes:

$$u = [F, \tau_x, \tau_y, \tau_z]^T$$

Continuous Dynamics The continuous dynamics matrices $A_c \in \mathbb{R}^{12\times 12}$ and $B_c \in \mathbb{R}^{12\times 4}$ encode the kinematic mappings and gravity couplings (where $g = 9.8$ m/s^2). The non-zero derivatives of the state variables are defined as:

$$\begin{aligned}
&\dot{\varphi} = p, \quad \dot{\theta} = q, \quad \dot{\psi} = r \\
&\dot{X} = u, \quad \dot{Y} = v, \quad \dot{Z} = w \\
&\dot{u} = -g\theta, \quad \dot{v} = g\varphi \\
&\dot{p} = 10\tau_x, \quad \dot{q} = 10\tau_y, \quad \dot{r} = 6.6667\tau_z, \quad \dot{w} = 5F
\end{aligned}$$

To formulate the discrete-time MDP $x_{t+1} = Ax_t + Bu_t$, we discretize the continuous dynamics using Euler integration with a time step of $\Delta t = 0.05$ seconds:

$$A = I + A_c\Delta t, \quad B = B_c\Delta t$$

LQR Objective Matrices The reward function is defined by the standard LQR quadratic penalty $r(x, u) = -(x^T Qx + u^T Ru)$. We define a standard, uniform control effort penalty $R = I_{4\times 4}$.

To create the multi-objective conflict, we define two base state-penalty matrices. The first heavily penalizes positional tracking error, while the second relaxes positional tracking to simulate an economic, energy-saving mode. We define Q_{tracking} and Q_{eco} as diagonal matrices:

$$\begin{aligned}
Q_{\text{tracking}} &= \text{diag}(10, 10, 10, 100, 100, 100, 1, 1, 1, 1, 1, 1) \\
Q_{\text{eco}} &= \text{diag}(0.1, 0.1, 0.1, 1, 1, 1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1)
\end{aligned}$$

Expert Generation To test the agents under a controlled degree of objective conflict, we introduce a divergence parameter $\alpha \in [0, 1]$. When $\alpha = 0$, the experts are identical; when $\alpha = 1$, they are maximally divergent. In our experiments, we use $\alpha = 0.5$. The specific reward matrices for the two experts are generated via convex combinations:

$$\begin{aligned}
Q_1 \text{ (Agile Expert)} &= (0.5 + 0.5\alpha)\, Q_{\text{tracking}} + (0.5 - 0.5\alpha)\, Q_{\text{eco}} \\
Q_2 \text{ (Eco Expert)} &= (0.5 - 0.5\alpha)\, Q_{\text{tracking}} + (0.5 + 0.5\alpha)\, Q_{\text{eco}}
\end{aligned}$$

Given A, B, R, and the respective Q_i, the true expert controllers K_1 and K_2 are obtained by solving the Discrete Algebraic Riccati Equation (DARE).

E.2 Exact L_∞ Distance to the Pareto Front via Linear Programming

In our empirical evaluations, we measure the suboptimality of a learned policy π by computing its normalized L_∞ distance to the continuous Pareto frontier. Because the exact continuous frontier is defined by the convex hull of the discrete, deterministic Pareto-optimal vertices, we cannot simply measure the distance to the nearest discrete vertex. Doing so would unfairly penalize policies that correctly learn an optimal stochastic mixture of experts along a continuous edge.

To compute the exact distance to the continuous frontier, we formulate the problem as a Linear Program (LP). Let $\mathcal{V} = \{v_1, v_2, \ldots, v_M\}$ be the set of expected return vectors for the M deterministic Pareto-optimal policies (we write M to avoid a clash with the dataset size N). Let $J(\pi) \in \mathbb{R}^d$ be the expected return vector of the evaluated policy.

To adjust for the different scales of the objectives, we compute the objective ranges $R_i = J_i^{\max} - J_i^{\min}$, where $J^{\max}$ and $J^{\min}$ are the coordinate-wise maximum and minimum of the Pareto vertices, respectively. The vertices on the Pareto front can be computed using the Optimistic Linear Support algorithm [40].

We seek to find the maximum normalized gap $\delta \geq 0$ such that there exists a valid point on the continuous Pareto frontier (a convex combination of the vertices $\mathcal{V}$) that dominates the learned policy $J(\pi)$ by at least δ across all d normalized objectives.

This is computed via the following Linear Program:

$$\begin{aligned} \max_{\delta,\alpha} \quad & \delta \\ \text{subject to} \quad & \sum_{j=1}^{M} \alpha_j v_{j,i} \geq J(\pi)_i + \delta \cdot R_i, \quad \forall i \in [d] \\ & \sum_{j=1}^{M} \alpha_j = 1 \\ & \alpha_j \geq 0, \quad \forall j \in \{1, \ldots, M\} \end{aligned}$$

where $\alpha \in \mathbb{R}^M$ represents the mixture weights over the discrete Pareto vertices. By maximizing δ, we find the exact worst-case normalized suboptimality gap between $J(\pi)$ and the true continuous boundary of the return polytope $\mathbb{J}$.

E.3 A Pedagogical Deep Dive into the Deep Sea Treasure Environment

In this section, we present a detailed analysis of the Deep Sea Treasure (DST) environment (Figure 8a), consolidating our empirical and theoretical findings into a cohesive, pedagogical narrative.

The DST agent receives a 2D reward signal: the value of the collected treasure and a constant step penalty for the time spent navigating. To establish the ground truth, we compute the exact Pareto vertices using Optimistic Linear Support (OLS) [40]. Because we have access to the environment's transition dynamics, we evaluate the scalarized objectives precisely via Value Iteration [35]. Assuming a uniform initial state distribution over all non-terminal states and a discount factor of $\gamma = 0.999$, we reconstruct the complete Pareto front, visualized in Figure 9.

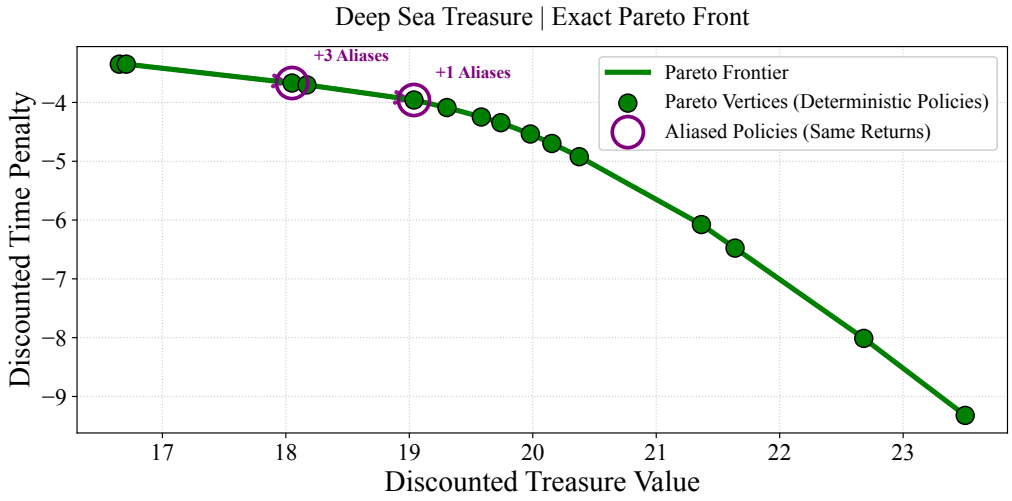


Figure 9: **The DST Return Polytope.** The exact continuous Pareto frontier is bounded by deterministic optimal policies (green vertices). Aliased policies (where multiple distinct optimal policies yield the exact same expected return) are highlighted with purple circles.

Pareto front structure First, we empirically validate the structural properties introduced in Section 2, namely, that the Pareto front is densely populated by structurally similar policies. We observe a compelling geometric phenomenon: almost all Pareto-optimal policies differ from their immediate neighbors at exactly one state. The only exceptions are two specific vertices (highlighted with purple circles in Figure 9). This observation demonstrates that the structural guarantee in [25, Theorem 1] does not strictly hold in highly structured environments like DST, where neighboring policies on the Pareto front may differ by more than one state. This is not a flaw in their proof, but rather a consequence of their measure-theoretic assumption that each return vector is uniquely attained by a single policy (thus confining structured environments to a measure-zero set of MOMDPs).

Upon further investigation, we found that the vertices violating the 1-flip rule are heavily *aliased*, meaning multiple distinct deterministic policies achieve the exact same expected return vector. This insight directly motivated our generalized "Pareto Path" guarantee (Theorem 2.1), which ensures 1-flip connectivity even in the presence of severe aliasing. Notably, policy-based search methods like Pareto Traversal [25, Algorithm 1] become computationally prohibitive here, as tracking all aliased combinations induces an exponential branching factor. This justifies our use of a value-based scalarization algorithm (OLS), which remains entirely agnostic to policy aliasing.

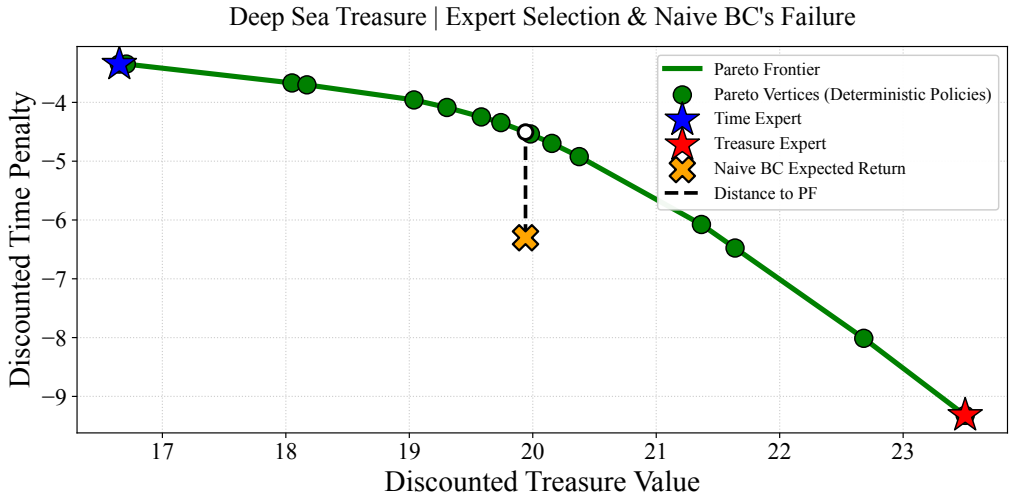


Figure 10: **The Geometric Failure of Naive BC (Failure II)**. We collect data from two extreme experts, highlighted using the blue and red stars. Naive BC aggregates all of this data into a single joint dataset. Because the experts fundamentally disagree at bottleneck states, the resulting Maximum Likelihood Estimate policy diffuses its probability mass destructively, causing its expected return to sag deeply into the suboptimal interior of the polytope.

Imitation learning (Failure II) We now turn our attention to the imitation learning setting. We select two extreme experts: a time-optimizing expert (blue star) and a treasure-optimizing expert (red star), as visualized in Figure 10. We sample trajectories from both experts and learn a single joint policy using naive Behavioral Cloning. As predicted by **Failure II** and demonstrated in Section 5, this pooled policy completely fails to reach the Pareto front.

To understand why, we compare this dominated policy to the closest mathematically optimal policy on the continuous Pareto frontier (Figure 11). We observe a stark contrast in their stochastic behaviors. The naively learned policy is overly stochastic and uniformly randomizes its actions across all divergent states, failing to optimize any coherent utility. In contrast, the true nearest Pareto-optimal policy is also stochastic, but its stochasticity is tightly concentrated at a *single* critical state. It carefully randomizes between navigating to one treasure versus the immediately adjacent one. This demonstrates that while stochastic mixing is necessary to traverse the continuous edges of the Pareto front, naive BC diffuses this entropy destructively, whereas the true optimal mixture focuses it.

Imitation learning (Failure I) Finally, we analyze the mechanics of MA-BC compared to Isolated BC to understand how it overcomes **Failure I**. We collect an extremely small dataset of five trajectories from each expert and train both algorithms. The resulting policies are visualized in Figure 12.

In the right panel of Figure 12, we observe that MA-BC successfully pools a massive portion of data from the treasure-optimizing expert to assist the time-minimizing learner in unvisited regions. However, this reveals a nuance: while much of the pooled data is consistent with a time-minimizing behavior, some is not. Specifically, the actions above cells (16, 19, 20, 22) direct the agent away from the nearest treasure, effectively increasing the time penalty. This illustrates that MA-BC can inadvertently inject bias into the target expert's dataset in the ultra-low data regime. The critical mechanism of MA-BC, however, is that this bias is inherently *self-correcting*. When we scale the dataset to 50 trajectories per expert (Figure 13), the bias almost vanishes. As the time-minimizing

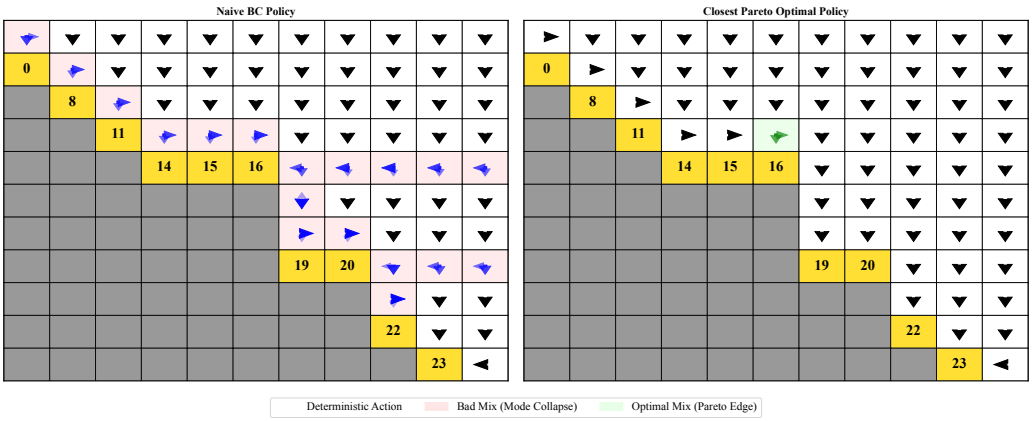


Figure 11: **Failure II: Naive BC vs. Closest Optimal Mixture. (Left)** The policy learned by Naive BC is overly stochastic at conflicting states (red cells). **(Right)** The closest theoretically optimal policy on the Pareto front. This policy achieves continuous returns by randomizing at only a single critical state, properly trading off between two adjacent treasures without diffusing entropy throughout the trajectory.

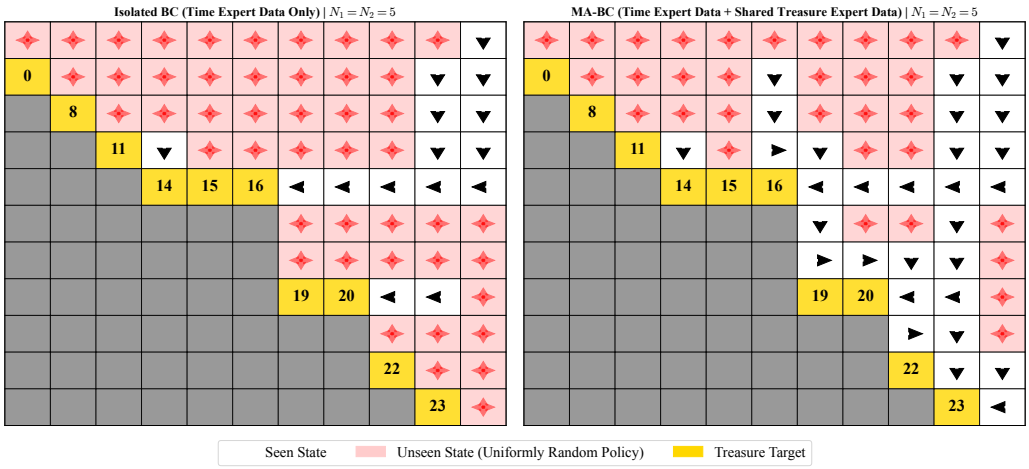


Figure 12: **Failure I: The Failure of Isolated BC** ($N_1 = N_2 = 5$). We depict the time-optimizing policy learned via Isolated BC (Left) versus MA-BC (Right) using only 5 trajectories. MA-BC aggressively pools non-divergent actions from the treasure-optimizing expert to fill in unvisited states. While largely helpful, MA-BC introduces temporary bias in the low-data regime (e.g., the actions above cells 16, 19, 20, and 22 incorrectly route the agent away from the nearest treasure).

expert explores more of the state space, it eventually generates trajectories that conflict with the previously pooled data at cells (16, 19, 20). MA-BC detects this conflict, immediately isolates those states, and reverts to the true expert's actions. This highlights the core operating principle of MA-BC: data is safely pooled until a contradiction is observed. This self-correcting property ensures that MA-BC remains a consistent estimator of the underlying expert policy asymptotically.

This phenomenon also provides vital context for our continuous state-action extension (Section 4). In continuous spaces, the spatial tolerance threshold δ introduces a constant bias that does *not* neatly self-correct, because exact state collisions almost surely have zero probability. This fundamentally explains the empirical bias-data tradeoff observed in Figure 5, dictating that as the baseline sample size increases, the continuous learner requires progressively smaller δ values to prioritize bias reduction over variance reduction.

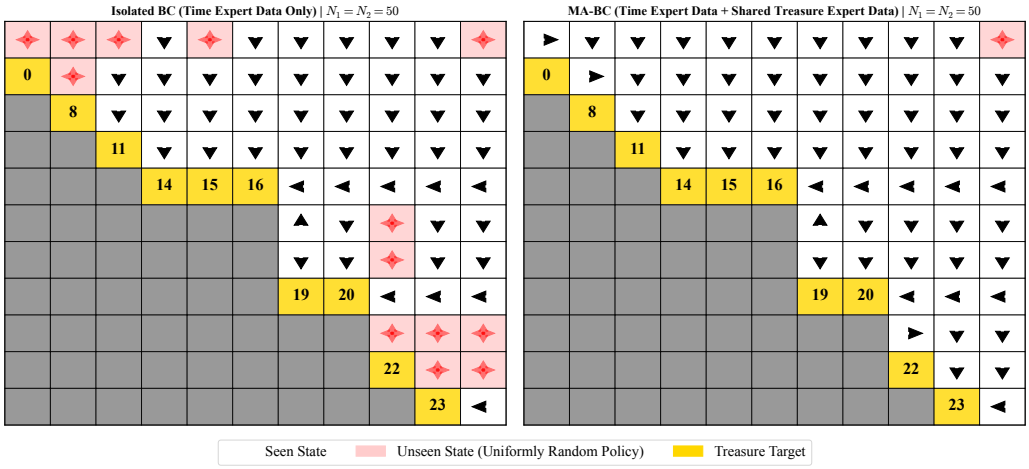


Figure 13: **MA-BC Self-Correction** ($N_1 = N_2 = 50$). As the sample size increases to 50 trajectories per expert, the bias previously introduced by MA-BC (Figure 12) is successfully self-corrected. The actions above cells 16, 19, and 20 (Right) have been overwritten to match the true time-minimizing behavior. As the target expert explores more states, conflicts are triggered, naturally partitioning the dataset and eliminating the bias.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims are directly supported by the theoretical proofs in Section 3 and the empirical results in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the reliance on the concentrability coefficient in Section 3 and note the heuristic nature of the continuous extension in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All assumptions are explicitly stated, and complete proofs for all theorems are provided in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Detailed descriptions of the environments, evaluation metrics, and algorithms are provided in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The complete source code to reproduce all experiments and plots is provided in the supplemental material.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Full experimental details are in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All empirical plots in Section 5 include shaded regions representing the standard error of the mean across multiple independent trials.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details regarding the computational resources and runtimes are included in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: The research strictly conforms to the NeurIPS Code of Ethics.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: This is foundational theoretical work in reinforcement and imitation learning and does not have immediate negative societal impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper focuses on algorithmic theory and simulated control tasks, and does not release high-risk models or scraped datasets.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Existing baseline environments and mathematical frameworks are properly cited in the main text and appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce novel, large-scale datasets or assets requiring separate documentation.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The research relies entirely on mathematical proofs and simulated computational environments; no human subjects were involved.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The research does not involve human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were not used as a component of the core methodology.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.

- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.