Provable Guarantees for Generative Behavior Cloning: Bridging Low-Level Stability and High-Level Behavior

Adam Block
MIT
Cambridge, MA
ablock@mit.edu

Ali Jadbabaie
MIT
Cambridge, MA
jadbabai@mit.edu

Daniel Pfrommer
MIT
Cambridge, MA
dpfrom@mit.edu

Max Simchowitz
MIT
Cambridge, MA
msimchow@csail.mit.edu

Russ Tedrake
MIT
Cambridge, MA
russt@mit.edu

Abstract

We propose a theoretical framework for studying behavior cloning of complex expert demonstrations using generative modeling. Our framework invokes low-level controllers - either learned or implicit in position-command control - to stabilize imitation around expert demonstrations. We show that with (a) a suitable low-level stability guarantee and (b) a powerful enough generative model as our imitation learner, pure supervised behavior cloning can generate trajectories matching the per-time step distribution of essentially arbitrary expert trajectories in an optimal transport cost. Our analysis relies on a stochastic continuity property of the learned policy we call "total variation continuity" (TVC). We then show that TVC can be ensured with minimal degradation of accuracy by combining a popular data-augmentation regimen with a novel algorithmic trick: adding augmentation noise at execution time. We instantiate our guarantees for policies parameterized by diffusion models and prove that if the learner accurately estimates the score of the (noise-augmented) expert policy, then the distribution of imitator trajectories is close to the demonstrator distribution in a natural optimal transport distance. Our analysis constructs intricate couplings between noise-augmented trajectories, a technique that may be of independent interest. We conclude by empirically validating our algorithmic recommendations, and discussing implications for future research directions for better behavior cloning with generative modeling.

1 Introduction

Training dynamic agents from datasets of expert examples, known as *imitation learning*, promises to take advantage of the plentiful demonstrations available in the modern data environment, in an analogous manner to the recent successes of language models conducting unsupervised learning on enormous corpora of text [67, 70]. Imitation learning is especially exciting in robotics, where mass stores of pre-recorded demonstrations on Youtube [1] or cheaply collected simulated trajectories [42, 20] can be converted into learned robotic policies.

For imitation learning to be a viable path toward generalist robotic behavior, it needs to be able to both represent and *execute* the complex behaviors exhibited in the demonstrated data. An approach that has shown tremendous promise is *generative behavior cloning:* fitting generative models, such

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

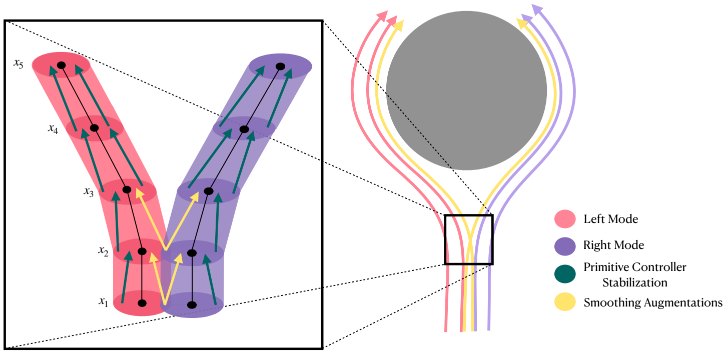

Figure 1: Consider demonstration trajectories exhibiting two modes: a “go left” and “go right” mode around an obstacle depicted in red and purple, respectively. To avoid compounding error, we imitate sequences of simple low-level feedback controllers we call “primitive controllers”, not simply raw actions. Intuitively, primitive controllers provide “tubes" around each demonstration trajectory where the system can be stabilized. Depicted in yellow, our data-noising procedure described below “fills in the gaps” in the demonstration, switching between modes in a well-behaved manner, and whilst allowing the primitive controllers to manage the stabilization.

as diffusion models [2, 19, 33], to expert demonstrations with pure supervised learning. In this paper, we ask: **When can generative behavior cloning imitate arbitrarily complex expert behavior?**

In this paper, we are interested in how *algorithmic choices* interface with the *dynamics of the agent’s environment* to render imitation possible. The key challenge separating imitation learning from vanilla supervised learning is one of *compounding error*: when the learner executes the trained behavior in its environment, small mistakes can accumulate into larger ones; this in turn may bring the agent to regions of state space not seen during training, leading to larger-still deviations from intended trajectories. Without the strong requirement that the learner can interactively query the expert at new states [40, 57], it is well understood that ensuring some form of *stability* in the imitation learning procedure is indispensable [68, 27, 50]. While many natural notions of stability exist for simple behaviors, how to *enforce* stability when imitating more complex behaviors remains an open question. Multi-modal trajectories present a key example of this challenge: consider a robot navigating around an obstacle; because there is no difference between navigating around the object to the right and around to the left, the dataset of expert trajectories may include examples of both options. This bifurcation of good trajectories can make it difficult for the agent to effectively choose which direction to go, possibly even causing the robot to oscillate between directions and run into the object. [19]. Moreover, human demonstrators correlate current actions with the past in order to *commit* to either a right or left path, which makes even formulating the idea of an “expert *policy*” a conceptually challenging one. Lastly, bifurcations are necessarily *incompatible* with previous notions of stability derived from classical control theory [68, 27, 50]. **In this work, we investigate how these strong and often unrealistic assumptions on the expert policy can be replaced by practical (and often realistic) assumptions on available algorithms.**

1.1 Contributions.

As in previous work, we formalize behavior cloning in two stages: at *train-time*, we learn a map from observations to distributions over actions, supervised by (state, action)-pairs from expert demonstrations coming from N_{exp} independent expert trajectories, while at *test-time*, the learned map, or *policy*, is executed on random initial states (distributed identically to initial training states). Following the schematic of existing theoretical analyses of behavior cloning [68, 50, 27], we demonstrate that a policy trained by minimizing a certain supervised learning objective on expert demonstrations induces trajectories that approximate those of expert demonstrations. Our work considers a significantly more general setting than past theoretical literature, and one which reflects the strength of *generative models* for imitation. One corollary of our key contributions is summarized in the following informal statement. The main technical insights leading to the proof of the theorem are detailed in the bullet points below it, and depicted in Figure 1.

Theorem (informal). *Consider a generative behavior cloner $\hat{\pi}$ that learns to predict sequences of expert actions on horizon H,* ***along with low-level controllers that locally stabilize the trajectories.*** *Then, with a suitable data noising strategy, for all times $h \le H$,*

$$\mathbb{P}[\text{expert \& imitator trajectories disagree at some time } h \text{ by } \geq \varepsilon] \\ \leq \mathcal{O}_{\text{ISS}}\left(H\varepsilon + \tfrac{1}{\varepsilon^2}\textstyle\sum_h \mathbb{E}_{\text{expert},h}\left[\mathcal{W}_1(\mathbb{P}_{\text{expert actions}}, \mathbb{P}_{\text{imitator actions}})\right]\right)$$

where $\mathbb{E}_{\text{expert},h}[\mathcal{W}_1(\mathbb{P}_{\text{expert actions}}, \mathbb{P}_{\text{imitator actions}})]$ denotes a 1-Wasserstein distance in an appropriate metric between the conditional distribution over expert and imitator actions given the observation at time step h, and where $\mathcal{O}_{\text{ISS}}$ hides constants depending polynomially on the stability properties of the low-level controllers, defined formally in Section 3.1.

We now detail the key ingredients of our results.

1. We imitate stochastic demonstrators that may exhibit both complex correlations between actions in their trajectories (e.g. be non-Markovian) and multi-modal behavior. The natural object to imitate in this setting is the conditional probability distribution of expert actions given recent states, but marginalized over past states. We require said conditional action distribution to be learnable by a **generative model**, but otherwise arbitrarily complex: in particular, the conditional distribution of an expert actions given the state can be discontinuous (in any natural distance metric) as a function of state, as in the bifurcation depicted in Figure 1*(right)*.
2. We obtain **rigorous, theoretical guarantees** and without requiring either interactive data collection (e.g. DAGGER [57, 40]), or access to gradients of the expert policy (as in TASIL[50]). Instead, we replace these assumption with an oracle, described below, which **synthesizes stabilizing, low-level policies** along training demonstrations—the green arrows in Figure 1*(left)*. This mirrors recent work on generative behavior cloning that find that providing state-commands through inverse dynamics controllers [33, 2] or position-command controllers of end effectors [19] leads to substantially improved performance.
3. We also apply a subtle-yet-significant modification to a popular **data noising** strategy, which we show yields both theoretical and empirical beneifts. Data noising ensures a helpful property we denote *total variation continuity* that interpolates between modes in probability space (without naively averaging their trajectories in world space). This effectively "fills in the missing gaps" in bifurcations, as indicated by yellow arrows in Figure 1.

Our main results, Theorems 1 and 2, are reductions from imitation of complex expert trajectories to supervised generative learning of a specific conditional distribution. For concreteness, Theorem 3 instantiates the generative modeling with Denoising Diffusion Probabilistic Models (DDPMs) of sufficient regularity and expressivity (as investigated empirically in [19, 48, 26]), and establishes end-to-end guarantees for imitation of complex trajectories with sample complexity polynomial in relevant problem parameters. Our analysis framework exposes that any sufficient powerful generative learner obtains similar guarantees. Finally, we empirically validate the benefits of our proposed smoothing strategy in simulated robotic manipulation tasks. We now summarize the algorithmic choices and analytic ideas that facilitate our reduction.

Abridged Related Work. Due to space, we defer a full comparison to past work to Appendix B. DDPMs, proposed in [29, 60], along with their relatives have seen success in image generation [62, 55], along with imitation learning (without data augmentation) [33, 19, 48], which is the starting point of our work. Smoothing data augmentation is ubiquitous in modern imitation learning [40] and our approach corresponds to that of [36] but with noise added at inference time. Despite the benefits of adaptive data collection [58, 40], adaptive demonstrations are more expensive to collect. Previous analyses of imitation learning without adaptive data collection have focused on classical control-theoretic notions of stability, notably incremental stability, [68, 27, 50], which require continuity, Markovianity, and often determinism, and preclude the bifurcations permitted in our setting.

2 Setting

Notation and Preliminaries. Appendix A gives a full review of notation. Bold lower-case (resp. upper-case) denote vectors (resp. matrices). We abbreviate the concatenation of sequences via $\mathbf{z}_{1:n} = (\mathbf{z}_1, \dots, \mathbf{z}_n)$. Norms $\|\cdot\|$ are Euclidean for vectors and operator norms for matrices unless otherwise noted. Rigorous probability-theoretic preliminaries are provided in Appendix F. In short, all random variables take values in Polish spaces $\mathcal{X}$ (which include real vector spaces), the space of Borel distributions on $\mathcal{X}$ is denoted $\Delta(\mathcal{X})$. We rely heavily on ***couplings*** from optimal transport theory: given measures $X \sim \mathsf{P}$ and $X' \sim \mathsf{P}'$ on $\mathcal{X}$ and $\mathcal{X}'$ respectively, $\mathscr{C}(\mathsf{P}, \mathsf{P}')$ denotes the space of joint distributions $\mu \in \Delta(\mathcal{X} \times \mathcal{X}')$ called "couplings" such that $(X, X') \sim \mu$ has marginals $X \sim \mathsf{P}$ and $X' \sim \mathsf{P}$. $\Delta(\mathcal{X} \mid \mathcal{Y})$ denotes the space of conditional probability distributions $\mathsf{Q} : \mathcal{Y} \to \Delta(\mathcal{X})$, formally called probability ***kernels*** ; Appendix F rigorously justifies that, in our setting, all conditional distributions can be expressed as kernels (which we do throughout the paper without comment). Finally $\mathbf{I}_\infty(\mathcal{E})$ denotes the indicator taking value 1 if $\mathcal{E}$ is true and ∞ otherwise.

Dynamics and Demonstrations. We consider a discrete-time, control system with states $\mathbf{x}_t \in \mathcal{X} := \mathbb{R}^{d_x}$, and inputs $\mathbf{u}_t \in \mathcal{U} := \mathbb{R}^{d_u}$, obeying the following nonlinear dynamics

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t), \quad t \geq 1. \tag{2.1}$$

Given length $T \in \mathbb{N}$, we call sequences $\boldsymbol{\rho}_T = (\mathbf{x}_{1:T+1}, \mathbf{u}_{1:T}) \in \mathscr{P}_T := \mathcal{X}^{T+1} \times \mathcal{U}^T$ ***trajectories***. For simplicity, we assume that (2.1) is deterministic and address stochastic dynamics in Appendix N. Though the dynamics are Markov and deterministic, we consider a stochastic and possibly *non-Markovian* demonstrator, which allows for the multi-modality described in the Section 1.

Definition 2.1 (Expert Distribution)**.** Let $\mathcal{D}_{\mathrm{exp}} \in \Delta(\mathscr{P}_T)$ denote an ***expert distribution*** over trajectories to be imitated. $\mathcal{D}_{\mathbf{x}_1}$ denotes the distribution of $\mathbf{x}_1$ under $\boldsymbol{\rho}_T = (\mathbf{x}_{1:T+1}, \mathbf{u}_{1:T}) \sim \mathcal{D}_{\mathrm{exp}}$.

Primitive Controllers. Our approach is to imitate not just actions, but simple local *control policies*. In the body of this paper, we consider affine mappings $\mathcal{X} \to \mathcal{U}$ (redundantly) parameterized as $\mathbf{x} \mapsto \bar{\mathbf{u}} + \bar{\mathbf{K}}(\mathbf{x} - \bar{\mathbf{x}})$; we call these ***primitive controllers***, denoted with $\kappa = (\bar{\mathbf{u}}, \bar{\mathbf{x}}, \bar{\mathbf{K}}) \in \mathcal{K}$. We describe the synthesis of these controllers in Appendix D , and extend our results to general families of parameterized controllers in Appendix E. We argue in Appendix E that primitive controllers are in fact standard practice, and implicit via robotic position control in many applications of diffusion to robotic behavior cloning.

Chunking Policies and Indices. The expert distribution $\mathcal{D}_{\mathrm{exp}}$ may involve non-Markovian sequences of actions. We imititate these sequences via ***chunking policies***. Fix a ***chunk length*** $\tau_{\mathrm{chunk}} \in \mathbb{N}$ and ***observation length*** $\tau_{\mathrm{obs}} \leq \tau_{\mathrm{chunk}}$, and define time indices $t_h = (h-1)\tau_{\mathrm{chunk}} + 1$. For simplicity, we assume τ_{chunk} divides T, and set $H = T/\tau_{\mathrm{chunk}}$. Given a $\boldsymbol{\rho}_T \in \mathscr{P}_T$, define the ***trajectory-chunks*** and ***observation*** chunks

$$\mathsf{s}_h := (\mathbf{x}_{t_{h-1}:t_h}, \mathbf{u}_{t_{h-1}:t_h-1}) \in \mathcal{S} := \mathscr{P}_{\tau_{\mathrm{chunk}}} \qquad \text{(trajectory-chunks)}$$

$$\mathsf{o}_h := (\mathbf{x}_{t_h-\tau_{\mathrm{obs}}+1:t_h}, \mathbf{u}_{t_h-\tau_{\mathrm{obs}}+1:t_h-1}) \in \mathcal{O} := \mathscr{P}_{\tau_{\mathrm{obs}}-1} \qquad \text{(observation-chunks)}$$

for $h > 1$, and $\mathsf{s}_1 = \mathsf{o}_1 = \mathbf{x}_1$ (for simplicity, we embed o_1 into $\mathscr{P}_{\tau_{\mathrm{obs}}-1}$ via zero-padding). We call τ_{chunk}-length sequences of primitive controllers ***composite actions***

$$\mathsf{a}_h = \kappa_{t_h:t_{h-1}} \in \mathcal{A} := \mathcal{K}^{\tau_{\mathrm{chunk}}}. \qquad \text{(composite actions)}$$

A ***chunking policy*** $\pi = (\pi_h)$ consists of functions π_h mapping observation-chunks o_h to distributions $\Delta(\mathcal{A})$ over composite actions and interacting with the dynamics (2.1) by $\mathsf{a}_h = \kappa_{t_h:t_{h-1}} \sim \pi_h(\mathsf{o}_h)$, and executing $\mathbf{u}_t = \kappa_t(\mathbf{x}_t)$. We let $d_{\mathcal{A}} = \tau_{\mathrm{chunk}}(d_x + d_u + d_x d_u)$ denote the dimension of the space $\mathcal{A}$ of composite actions. The chunking scheme is represented in Figure 2, demonstrating the rationale for using primitive controllers over open-loop actions. Remark C.1 describes our rationale for studying *states* over generic observations, and considering time-dependent policies.

Desideratum. The quality of imitation of a deterministic policy is naturally measured in terms of step-wise closeness of state and action [68, 50]. With stochastic policies, however, two rollouts of even the same policy can visit different states. We propose measuring *distributional closeness* via *couplings* introduced in the preliminaries above. We define the following losses, focusing on the *marginal distributions* between trajectories.

Definition 2.2. Given $\varepsilon > 0$ and a (chunking) policy π, the (marginal distribution) imitation loss is $\mathcal{L}_{\mathrm{marg},\varepsilon}(\pi) := \max_{t \in [T]} \inf_\mu \mathbb{P}_\mu \left[\max\left\{\|\mathbf{x}^{\mathrm{exp}}_{t+1} - \mathbf{x}^{\pi}_{t+1}\|, \|\mathbf{u}^{\mathrm{exp}}_t - \mathbf{u}^{\pi}_t\|\right\} > \varepsilon\right]$, where the infimum is over all couplings μ between the distribution of $\boldsymbol{\rho}_T$ under $\mathcal{D}_{\mathrm{exp}}$ and that induced by the policy π as described above, such that $\mathbb{P}_\mu[\mathbf{x}^{\mathrm{exp}}_1 = \mathbf{x}^{\pi}_1] = 1$.

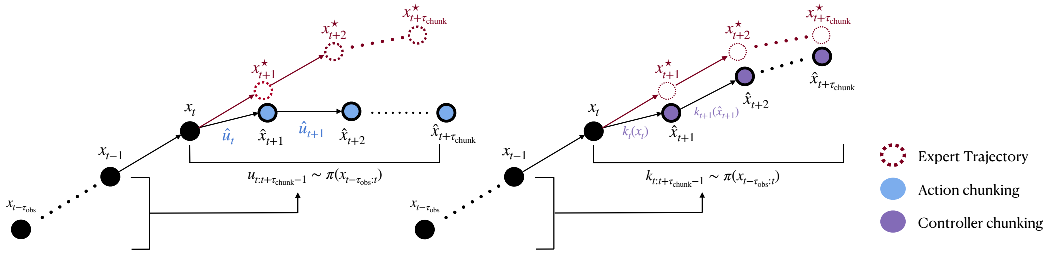
$x^\star_{t+\tau_{\text{chunk}}}$
$x^\star_{t+2}$
$x^\star_{t+1}$
x_t
$\hat{u}_t$
$\hat{x}_{t+1}$
$\hat{u}_{t+1}$
$\hat{x}_{t+2}$
$\hat{x}_{t+\tau_{\text{chunk}}}$
x_{t-1}
$u_{t:t+\tau_{\text{chunk}}-1} \sim \pi(x_{t-\tau_{\text{obs}}:t})$
$x_{t-\tau_{\text{obs}}}$
$x^\star_{t+\tau_{\text{chunk}}}$
$x^\star_{t+2}$
$x^\star_{t+1}$
$\hat{x}_{t+\tau_{\text{chunk}}}$
$\hat{x}_{t+2}$
$k_{t+1}(\hat{x}_{t+1})$
x_t
$k_t(x_t)$
$\hat{x}_{t+1}$
x_{t-1}
$k_{t:t+\tau_{\text{chunk}}-1} \sim \pi(x_{t-\tau_{\text{obs}}:t})$
$x_{t-\tau_{\text{obs}}}$
Expert Trajectory
Action chunking
Controller chunking

Figure 2: Graphical comparison of an action-chunk based policy (left) as described in [19], versus the primitive-controller chunking policy (right) proposed in this paper. The primitive controller paradigm allows for stabilizing back to the original expert trajectory, whereas using generated actions in an open-loop fashion may cause divergence from the expert in the presence of unstable system dynamics. We refer to **composite actions** as the sequence of primitve controllers given on the right.

Under stronger conditions (whose necessity we establish), we can also imitate joint distributions over actions (Appendix J). Observe that $\mathcal{L}_{\text{fin},\varepsilon} \le \mathcal{L}_{\text{marg},\varepsilon}$, and that both losses are equivalent to Wasserstein-type metrics on bounded domains. These losses are also equivalent to Lévy-Prokhorov metrics [64] under re-scaling of the Euclidean metric (even for unbounded domains), and also correspond to total variation analogues of shifted Renyi divergences [5, 6]. While empirically evaluating these infima over couplings is challenging, $\mathcal{L}_{\text{marg},\varepsilon}$ upper bounds the difference in expectation between any bounded and Lipschitz control cost decomposing across time steps, states and inputs, and $\mathcal{L}_{\text{fin},\varepsilon}$ upper bounds differences in final-state costs; see Appendix J for discussion.

Diffusion Models. Our analysis provides imitiation guarantees when chunking policies π_h select a_h via a sufficiently accurate generative model. Given their recent success, instantiate our analysis for the popular Denoising Diffusion Probabilistic Models (DDPM) framework [18, 41] that allows the learner to sample from a density $q \in \Delta(\mathbb{R}^d)$ assuming that the *score* $\nabla \log q$ is known to the learner. More precisely, suppose the learner is given an observation o_h and wishes to sample $\mathsf{a}_h \sim q(\cdot|\mathsf{o}_h)$ for some family of probability kernels $q(\cdot|\cdot)$. A DDPM starts with some a_h^0 sampled from a standard Gaussian noise and iteratively "denoises" for each DDPM-time step $0 \le j < J$:

$$\mathsf{a}_h^j = \mathsf{a}_h^{j-1} - \alpha \cdot \mathsf{s}_{\theta,h}(\mathsf{a}_h^{j-1}, \mathsf{o}_h, j) + 2 \cdot \mathcal{N}(0, \alpha^2 \mathbf{I}), \tag{2.2}$$

where $\mathsf{s}_{\theta,h}(\mathsf{a}_h^j, \mathsf{o}_h, j)$ estimates the true score $\mathsf{s}_{\star,h}(\mathsf{a}_h, \mathsf{o}_h, \alpha j)$, formally defined for any continuous argument $t \le J\alpha$ to be $\mathsf{s}_{\star,h}(\mathsf{a}, \mathsf{o}_h, t) := \nabla_{\mathsf{a}} \log q^\star_{h,[t]}(\mathsf{a} \mid \mathsf{o}_h)$, where $q^\star_{h,[t]}(\cdot|\mathsf{o}_h)$ is the distribution of $e^{-t}\mathsf{a}_h^{(0)} + \sqrt{1 - e^{-2t}}\boldsymbol{\gamma}$ with $\mathsf{a}_h^{(0)}$ is sampled from the target distribution we which to sample from, and $\boldsymbol{\gamma} \sim \mathcal{N}(0, \mathbf{I})$ is a standard Gaussian. We denote by $\mathtt{DDPM}(\mathsf{s}_\theta, \mathsf{o}_h)$ the law of a_h^J sampled according to the DDPM using $\mathsf{s}_\theta(\cdot, \mathsf{o}_h, \cdot)$ as a score estimator. Preliminaries on DPPMs are detailed in Appendix L.

3 Conditional sampling with stabilization suffices for behavior cloning

We show that trajectories of the form given in Definition 2.1 can be efficiently imitated if (a) we are given a *synthesis oracle*, described below, that produces low-level control policies that locally stabilize chunks of the trajectory with primitive controllers and (b) we can learn to generate certain appropriate distributions over composite actions, i.e. sequences of primitive controllers. All the following results apply to affine primitive controllers introduced in Section 2 and assume that the system dynamics are second-order smooth and locally stabilizable. In Appendix E, we show that our results still hold with general families of parametric primitive controllers, provided that these controllers induce the same local stability guarantee.

The synthesis oracle. We say primitive controller (cf. Section 2) $\kappa_{1:T} \in \mathcal{K}^T$ is *consistent with* a trajectory $\boldsymbol{\rho} = (\mathbf{x}_{1:T+1}, \mathbf{u}_{1:T}) \in \mathscr{P}_T$ if $\bar{\mathbf{x}}_t = \mathbf{x}_t$ and $\bar{\mathbf{u}}_t = \mathbf{u}_t$ for all $t \in [T]$; note that this implies that $\kappa_t(\mathbf{x}_t) = \mathbf{u}_t$ for all t. A ***synthesis oracle*** $\mathtt{synth}$ maps $\mathscr{P}_T \to \mathcal{K}^T$ such that, for all $\boldsymbol{\rho}_T \in \mathscr{P}_T$, $\kappa_{1:T} = \mathtt{synth}(\boldsymbol{\rho}_T)$ is consistent with $\boldsymbol{\rho}_T$. For our theory, we assume access to a synthesis oracle at training time, and assume the ability to estimate conditional distributions over joint sequences of primitive controllers; Appendix K explains how this can be implemented by solving Ricatti equations if dynamics are known (e.g. in a simulator), smooth, and stabilizable. In our experimental environment, control inputs are desired robot configurations, which the simulated robot executes by applying feedback gains. As discussed in Appendix E, learned or hand-coded low-level controllers are popular in practical implementations of generative behavior cloning. We discuss the merits of studying imitation learning with a synthesis oracle in depth in Appendix C.3.

Notions of distance. While restricting ourselves to affine primitive controllers, our approximation error of generative behavior cloner is measured in terms of optimal transport distances that use the following "maximum distance." Given two composite actions $\mathsf{a} = (\bar{\mathbf{u}}_{1:\tau_{\mathrm{chunk}}}, \bar{\mathbf{x}}_{1:\tau_{\mathrm{chunk}}}, \bar{\mathbf{K}}_{1:\tau_{\mathrm{chunk}}})$ and $\mathsf{a}' = (\bar{\mathbf{u}}'_{1:\tau_{\mathrm{chunk}}}, \bar{\mathbf{x}}'_{1:\tau_{\mathrm{chunk}}}, \bar{\mathbf{K}}'_{1:\tau_{\mathrm{chunk}}})$, we define

$$\mathsf{d}_{\max}(\mathsf{a}, \mathsf{a}') := \max_{1 \le k \le \tau_{\mathrm{chunk}}} (\|\bar{\mathbf{u}}_k - \bar{\mathbf{u}}'_k\| + \|\bar{\mathbf{x}}_k - \bar{\mathbf{x}}'_k\| + \|\bar{\mathbf{K}}_k - \bar{\mathbf{K}}'_k\|). \tag{3.1}$$

Distances between policies are defined via natural optimal transport costs. Given two policies $\pi = (\pi_h), \pi' = (\pi'_h)$ and observation chunk o_h, we define an induced optimal transport cost

$$\Delta_\varepsilon(\pi_h(\mathsf{o}_h), \pi'_h(\mathsf{o}_h)) := \inf_\mu \mathbb{P}_{(\mathsf{a}_h, \mathsf{a}'_h) \sim \mu} \left[\mathsf{d}_{\max}(\mathsf{a}_h, \mathsf{a}'_h) > \varepsilon\right],$$

where the $\inf_\mu$ denotes the infinum over all couplings between $\mathsf{a}_h \sim \pi_h(\mathsf{o}_h)$ and $\mathsf{a}'_h \sim \pi'_h(\mathsf{o}_h)$. Δ_ε corresponds to a relaxed Lévy-Prokhorov metric [64], and can always be bounded, via Markov's inequality, by $\Delta_\varepsilon(\pi_h, \pi'_h \mid \mathsf{o}_h) \le \frac{1}{\varepsilon}\mathcal{W}_{1,\mathsf{d}_{\max}}(\pi_h(\mathsf{o}_h), \pi'_h(\mathsf{o}_h))$, where $\mathcal{W}_{1,\mathsf{d}_{\max}}(\pi_h(\mathsf{o}_h), \pi'_h(\mathsf{o}_h))$ denotes the 1-Wasserstein distance between $\mathsf{a}_h \sim \pi_h(\mathsf{o}_h)$ and $\mathsf{a}'_h \sim \pi'_h(\mathsf{o}_h)$.

3.1 Incremental Stability and the Synthesis Oracle.

We assume that synthesis oracle above produces *incrementally stabilizing* control gains, in the sense first proposed by [9]. Incremental stability has emerged as a natural desirable property for imitation limitation [50, 68, 27], because it forces the expert to be robust to small perturbations of their policy. We now supply a formal definition. Given a primitive controller $\kappa : \mathbb{R}^{d_x} \to \mathbb{R}^{d_u}$, define the closed loop dynamic map $f_{\mathrm{cl},\kappa}(\mathbf{x}, \delta\mathbf{u}) := f(\mathbf{x}, \kappa(\mathbf{x}) + \delta\mathbf{u})$. Thus, composite action a is *consistent* with a trajectory chunk $\mathsf{s} = (\mathbf{x}_{1:\tau_{\mathrm{chunk}}+1}, \mathbf{u}_{1:\tau})$ if $\mathbf{x}_{t+1} = f_{\mathrm{cl},\kappa_t}(\mathbf{x}, \mathbf{0})$ for $1 \le t \le \tau_{\mathrm{chunk}}$.[1]

Definition 3.1 (Time-Varying Incremental Stability)**.** Let $\gamma(\cdot)$ be a class K function, $\beta(\cdot,\cdot)$ be class KL function, and let $\mathsf{a} = (\kappa_1, \kappa_2, \dots, \kappa_\tau)$ denote a sequence of primitive controllers (i.e. a composite action when $\tau = \tau_{\mathrm{chunk}}$). Given a sequence of input perturbations $\delta\mathbf{u}_{1:\tau} \in (\mathbb{R}^{d_u})^\tau$ and initial condition $\boldsymbol{\xi} \in \mathbb{R}^{d_x}$, let $\mathbf{x}^{\mathsf{a}}_{i+1}(\delta\mathbf{u}_{1:\tau}, \boldsymbol{\xi}) = f_{\mathrm{cl},\kappa_i}(\mathbf{x}^{\mathsf{a}}_i(\delta\mathbf{u}_{1:\tau}, \boldsymbol{\xi}), \delta\mathbf{u}_i)$, with $\mathbf{x}^{\mathsf{a}}_1 = \boldsymbol{\xi}$. We say that composite action a is time-varying incrementally input-to-state stable (t-ISS) with moduli $\gamma(\cdot), \beta(\cdot,\cdot)$ if for all $\boldsymbol{\xi}, \boldsymbol{\xi}' \in \mathbb{R}^{d_x}, 0 \le i \le \tau$, $\|\mathbf{x}^{\mathsf{a}}_i(\mathbf{0}_{1:\tau}, \boldsymbol{\xi}) - \mathbf{x}^{\mathsf{a}}_i(\delta\mathbf{u}_{1:\tau}, \boldsymbol{\xi}')\| \le \beta(\|\boldsymbol{\xi} - \boldsymbol{\xi}'\|, \tau) + \gamma\left(\max_{1 \le s \le i-1} \|\delta\mathbf{u}_s\|\right)$. Given parameters $c_\gamma, c_\xi > 0$ we say that a is local-t-ISS at $\boldsymbol{\xi}_0$ if the above holds only for all $\boldsymbol{\xi}, \boldsymbol{\xi}', \delta\mathbf{u}_{1:\tau}$ such that $\|\boldsymbol{\xi} - \boldsymbol{\xi}_0\|, \|\boldsymbol{\xi}' - \boldsymbol{\xi}_0\| \le c_\xi$ and $\max_t \|\delta\mathbf{u}_t\| \le c_\gamma$.

Incremental stability implies that as the inital conditions $\|\boldsymbol{\xi} - \boldsymbol{\xi}'\| \to 0$ and $\max_{0 \le s \le i-1} \|\delta\mathbf{u}_t\| \to 0$, the trajectories induced by taking rolling out a from $\boldsymbol{\xi}$, and rolling out a from $\boldsymbol{\xi}'$ with additive input perturbations $\delta\mathbf{u}_{1:\tau}$ tend to zero in norm. This behavior needs only hold for initial conditions in a small neighborhood of a nominal state $\boldsymbol{\xi}_0$. Importantly, the perturbations $\delta\mathbf{u}_{1:\tau}$ are fixed perturbations of inputs, applied to the *closed loop behavior* under the controllers. Our notion of incremental stability are similar too, but sublty different similar notions of past work. We provide an extended comparisons in Appendix E.2. Our main assumption is that the synthesis oracle described above produced primitive controllers which are consistent with, and incrementally stabilizing for, the demonstrated trajectories. Figure 1 demonstrates the effect of stabilizing primitive controllers.

Assumption 3.1. We assume that our synthesis oracle enjoys the following property. Let $\boldsymbol{\rho}_T = (\mathbf{x}_{1:T+1}, \mathbf{u}_{1:T}) \sim \mathcal{D}_{\mathrm{exp}}$, and let $\kappa_{1:T} = \mathtt{synth}(\boldsymbol{\rho}_T)$, partitioned into composite actions $\mathsf{a}_{1:H}$, with $\kappa_t(\mathbf{x}) = \bar{\mathbf{K}}_t(\mathbf{x} - \bar{\mathbf{x}}_t) + \bar{\mathbf{u}}_t$. We assume that, with probability one, $\kappa_{1:T}$ is consistent with $\boldsymbol{\rho}_T$[2], and that, for each $1 \le h \le H$, $\mathsf{a}_h = (\kappa_{t_h:t_h+\tau_{\mathrm{chunk}}-1})$ is local t-ISS at $\mathbf{x}_{t_h}$ with moduli γ, β and parameters $c_\beta, c_\xi > 0$. We further assume that γ and β take the form

$$\gamma(u) = \bar{c}_\gamma \cdot u, \quad \beta(u, k) = \bar{c}_\beta e^{-(k-1)L_\beta} \cdot u, \quad \bar{c}_\gamma, \bar{c}_\beta > 0, \quad L_\beta \in (0, 1].$$

[1] Below, we recall definitions of classes of comparison functions in nonlinear control [38] as follows: we say a univariate function $\gamma : \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$ is *class* K if it is strictly increasing and satisfies $\gamma(0) = 0$. We say a bivariate function $\beta : \mathbb{R}_{\ge 0} \times \mathbb{Z}_{\ge 0} \to \mathbb{R}_{\ge 0}$ is *class* KL if $x \mapsto \beta(x, t)$ is class K for each $t \ge 0$, and $t \mapsto \beta(x, t)$ is nonincreasing in t.

[2] Note that this implies $\bar{\mathbf{x}}_t = \mathbf{x}_t$ and $\bar{\mathbf{u}}_t = \mathbf{u}_t$.

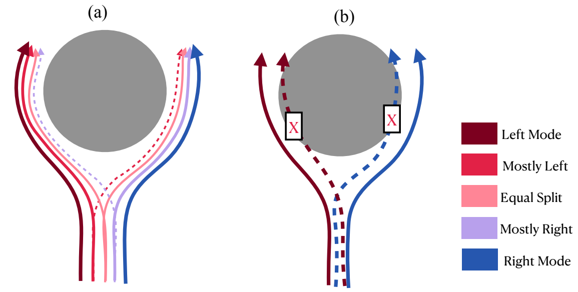

Figure 3: Graphical representation of total variation continuity (TVC) using the running "left mode/right mode" example. Panel *(a)* depicts a policy π which is TVC, and thus interpolates between left and right modes probabilistically. Importantly, the TVC property applies to the distribution over composite actions, i.e. sequences of *primitive controllers*, as, in Definition 3.3; this ensures, for example, that following the left mode from slightly to the right of the obstacle (purple dotted line) still stabilizes to the idealized left mode trajectory (red). In panel *(b)*, we consider a policy which for TVC applies to the sequences of *raw control inputs* (which *is not* what occurs in Definition 3.3). This can lead to naive mode-switching that collides with the gray obstacle.

Lastly, we assume that for the expert trajectories and the primitive controllers drawn as above, it holds that satisfy $\max\{\|\mathbf{x}_t\|, \|\mathbf{u}_t\|\} \le R_{\text{dyn}}$ and $\|\bar{\mathbf{K}}_t\| \le R_{\mathbf{K}}$ with probability one.

In Appendix K, we show that Assumption 3.1 holds whenever (a) the dynamics of our system are smooth (but not necessarily linear!) (b) the affine gains are chosen to stabilize the Jacobian linearizations of the system around the nominal trajectory.

Definition 3.2 (Problem constants)**.** Throughout, we refer to constants $c_1, c_2, c_3, c_4, c_5 > 0$, which are polynomial in the terms in Assumption 3.1, and which are defined formally in Appendix K.

3.2 Simplified guarantees under total variation continuity

This section presents our main theoretical result: if one learns a chunking policy $\hat{\pi}$ that can compute the conditional distribution of composite actions at time steps given observation-chunks, then a stochastically smoothed version of this policy, $\hat{\pi}_\sigma$, has low imitation error. Define, for any length $\tau \in \mathbb{N}$, the *trajectory distance* between trajectories $\boldsymbol{\rho} = (\mathbf{x}_{1:\tau+1}, \mathbf{u}_{1:\tau}), \boldsymbol{\rho}' = (\mathbf{x}'_{1:\tau+1}, \mathbf{u}'_{1:\tau}) \in \mathscr{P}_\tau$

$$\mathsf{d}_{\text{traj}}(\boldsymbol{\rho}, \boldsymbol{\rho}') := \max_{1 \le k \le \tau+1} \|\mathbf{x}_k - \mathbf{x}'_k\| \vee \max_{1 \le k \le \tau} \|\mathbf{u}_k - \mathbf{u}'_k\|. \tag{3.2}$$

In particular, we define $\mathsf{d}_{\text{traj}}(\mathsf{o}_h, \mathsf{o}'_h)$ and $\mathsf{d}_{\text{traj}}(\mathsf{s}_h, \mathsf{s}'_h)$ by viewing these as trajectories of length $\tau_{\text{obs}} - 1$ and τ_{chunk}, respectively. Lastly, we define a per-timestep restriction of the expert distribution. In this section, we consider the case where the learner policy satisfies a total variation continuity (TVC) condition, defined below.

Definition 3.3 (TVC of Chunking Policies)**.** We say that a chunking policy $\pi = (\pi_h)$ is total variation continuous with modulus $\gamma_{\text{TVC}} : \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$, written γ_{TVC}-TVC, if, for all $h \in [H]$ and any observation-chunks $\mathsf{o}_h, \mathsf{o}'_h \in \mathscr{P}_{\tau_{\text{obs}}-1}$, $\mathsf{TV}(\pi_h(\mathsf{o}_h), \pi_h(\mathsf{o}'_h)) \le \gamma_{\text{TVC}}(\mathsf{d}_{\text{traj}}(\mathsf{o}_h, \mathsf{o}'_h))$.

We depict the TVC property using our running left-right obstacle example in Figure 3. We stress that, in Definition 3.3, the TV bound on $\mathsf{TV}(\pi_h(\mathsf{o}_h), \pi_h(\mathsf{o}'_h))$ applies to the *composite actions* consisting of primitive controllers $\mathsf{a}_h = \kappa_{t_h:t_h+\tau_{\text{chunk}}-1} \sim \pi_h(\mathsf{o}_h)$; it does not upper bound the TV distance between raw control inputs. Indeed, ensuring TVC of the latter can lead to the failure modes depicted in Figure 3(b). Next, we extract an expert "policy" from the expert demonstrations.

Definition 3.4 (Expert "policy" with synthesized controllers)**.** For $h \in [H]$, we let $\mathcal{D}_{\text{exp},h}$ denote the joint distribution of $(\mathsf{a}_h, \mathsf{o}_h)$, induced by drawing a trajectory $\boldsymbol{\rho}_T = (\mathbf{x}_{1:T+1}, \mathbf{u}_{1:T}) \sim \mathcal{D}_{\text{exp}}$ from the expert distribution, $\kappa_{1:T} = \mathtt{synth}(\boldsymbol{\rho}_T)$ be the associated primitive controllers, letting $\mathsf{o}_h = (\mathbf{x}_{t_h-\tau_{\text{obs}}+1:t_h}, \mathbf{u}_{t_h-\tau_{\text{obs}}+1:t_h-1})$ be the associated observation-chunk at time h, and $\mathsf{a}_h = \kappa_{t_h:t_{h+1}-1}$ the associated composite action. We let $\pi^\star_h(\cdot) : \mathcal{O} \to \Delta(\mathcal{A})$ denote the condition distribution of $\mathsf{a}_h \mid \mathsf{o}_h$ under $\mathcal{D}_{\text{exp},h}$.

The conditional distributions $\pi^\star_h(\cdot)$ are estimated when training a generative model to predict a_h from observations o_h. Note that $\pi^\star_h(\cdot)$ (and $\mathcal{D}_{\text{exp},h}$) is defined in terms of *both* expert demonstration from $\mathcal{D}_{\text{exp}}$ and the associated synthesized primitive controllers. In Lemma J.6, we show that when the synthesis oracle $\kappa_{1:T} = \texttt{synth}(\boldsymbol{\rho}_T)$ produces primitive controllers consistent with the trajectories, than $\pi^\star = (\pi^\star_h)$ produces the same marginals over states as $\mathcal{D}_{\text{exp}}$; that is, $\mathcal{L}_{\text{marg},\varepsilon}(\pi^\star) = 0$.

Theorem 1. Suppose Assumption 3.1 holds, and suppose that $0 \le \varepsilon < c_2$, and $\tau_{\text{chunk}} \ge c_3$. Then, for any non-decreasing non-negative $\gamma_{\text{TVC}}(\cdot)$ and γ_{TVC}-TVC chunking policy $\hat{\pi}$, it holds that $\mathcal{L}_{\text{marg},\varepsilon}(\hat{\pi}) \le H\gamma_{\text{TVC}}(\varepsilon) + \sum_{h=1}^{H} \mathbb{E}_{\mathsf{o}_h \sim \mathcal{D}_{\text{exp},h}} \Delta_{(\varepsilon/c_1)} \left(\pi^\star_h(\mathsf{o}_h), \hat{\pi}_h(\mathsf{o}_h)\right)$, which is at most $H\gamma_{\text{TVC}}(\varepsilon) + \frac{c_1}{\varepsilon} \sum_{h=1}^{H} \mathbb{E}_{\mathsf{o}_h \sim \mathcal{D}_{\text{exp},h}} \left[\mathcal{W}_{1,\mathsf{d}_{\max}} \left(\pi^\star_h(\mathsf{o}_h), \hat{\pi}_h(\mathsf{o}_h)\right)\right]$.

The above result reduces the marginal imitation error of $\hat{\pi}$ to the sum over optimal transport errors between $\hat{\pi}$ and $\mathsf{a} \mid \mathsf{o}_h$ chosen by the expert demonstrators. Thus, if these are small, the local stabilization properties of the primitive controllers guaranteed by Assumption 3.1 ensure that errors compound at most linearly in problem horizon. The key ideas of the proof are given Appendix D, via a general template for imitation learning of general stochastic policies. This template is instantiated with a details in Appendix J.

3.3 A general guarantee via data noising.

To circumvent assuming that the learner's policy is TVC, we study estimating the conditionals under a popular data augmentation technique [36], where the learner is trained to imitate the conditional sequence of $\mathsf{a} \mid \tilde{\mathsf{o}}_h$, where $\tilde{\mathsf{o}}_h \sim \mathcal{N}(\mathsf{o}_h, \sigma^2 \mathbf{I})$ adds σ^2-variance Gaussian noise to the true observation-chunk. To understand this better, consider the following *smoothed* policy:

Definition 3.5 (The smoothed policy)**.** Let $\hat{\pi} = (\hat{\pi}_h)$ be a chunking policy. We define the *smoothed policy* $\hat{\pi}_\sigma = (\hat{\pi}_{\sigma,h})$ by letting $\hat{\pi}_{\sigma,h}(\cdot \mid \mathsf{o}_h)$ be distributed as $\hat{\pi}_h(\cdot \mid \tilde{\mathsf{o}}_h)$, where $\tilde{\mathsf{o}}_h \sim \mathcal{N}(\mathsf{o}_h, \sigma^2 \mathbf{I})$.

Appendix J.7.2 show's that Pinsker's inequality implies noising automatically enforces TVC This suggests that we can use some form of data noising to enforce the TVC property in Definition 3.3. Let's now consider a related problem: trying to estimate the optimal distribution over composite actions *conditioned on* a noised observation. This gives rise to a *deconvolution* of the expert policy, which can be thought as an inverse operation of data noising.

Definition 3.6 (Noised Data Distribution and Deconvolution Policy)**.** Let $\mathcal{D}_{\text{exp},h}$ be as in Definition 3.4. Define $\mathcal{D}_{\text{exp},\sigma,h}$ as the distribution over $(\tilde{\mathsf{o}}_h, \mathsf{a}_h)$ generated by $(\mathsf{o}_h, \mathsf{a}_h) \sim \mathcal{D}_{\text{exp},h}$ and $\tilde{\mathsf{o}}_h \sim \mathcal{N}(\mathsf{o}_h, \sigma^2 \mathbf{I})$. We define the *deconvolution policy* $\pi^\star_{\text{dec},\sigma,h}(\tilde{\mathsf{o}}_h)$ as the conditional distribution of $\mathsf{a}_h \mid \tilde{\mathsf{o}}_h$ under $\mathcal{D}_{\text{exp},\sigma,h}$.

Analogously to $\pi^\star$, the policy $\pi^\star_{\text{dec},\sigma,h}$ is what a generative model trained to generate a_h from noised observations $\tilde{\mathsf{o}}_h$ of $\mathsf{o}_h \sim \mathcal{D}_{\text{exp}}$ learns to generate. Our next theorem states that, if our $\hat{\pi}$ approximates the idealized conditional distributation of composite actions given noised observations, then $\hat{\pi}_\sigma$, the smoothed policy, imitates the expert distribution with provable bounds on its imitation error:

Theorem 2 (Reduction to conditional sampling under nosing)**.** Suppose Assumption 3.1 holds. Let $c_1, \dots, c_5 > 0$, defined in Definition 3.2, and let $\Theta_{\text{Iss}}(x)$ denote a term which is upper and lower bounded by a x times a polynomial in those constants and their inverses. Then, for $\varepsilon \le \Theta_{\text{Iss}}(1)$, if we choose $\sigma = \varepsilon / \Theta_{\text{Iss}}(\sqrt{d_x} + \log(1/\varepsilon))$ and let $\tau_{\text{chunk}} \le c_3$ and $\tau_{\text{chunk}} - \tau_{\text{obs}} \ge \frac{1}{L_\beta} \log(c_1/\varepsilon)$,

$$\mathcal{L}_{\text{marg},\varepsilon}(\hat{\pi}_\sigma) \le \Theta_{\text{Iss}}\left(\varepsilon H \sqrt{\tau_{\text{obs}}} \cdot (\sqrt{d_x} + \log(\tfrac{1}{\varepsilon})\right) + \sum_{h=1}^{H} \mathbb{E}_{\tilde{\mathsf{o}}_h \sim \mathcal{D}_{\text{exp},\sigma,h}} \left[\Delta_{(\varepsilon^2)} \left(\pi^\star_{\text{dec},\sigma,h}\left(\tilde{\mathsf{o}}_h\right), \hat{\pi}_h\left(\tilde{\mathsf{o}}_h\right)\right)\right],$$

which is upper bounded by at most $\Theta_{\text{Iss}}\left(\varepsilon H \sqrt{\tau_{\text{obs}}} \cdot (\sqrt{d_x} + \log(1/\varepsilon)\right) + \frac{1}{\varepsilon^2} \sum_{h=1}^{H} \mathbb{E}_{\mathsf{o}_h \sim \mathcal{D}_{\text{exp},\sigma,h}} \left[\mathcal{W}_{1,\mathsf{d}_{\max}} \left(\pi^\star_{\text{dec},\sigma,h}(\tilde{\mathsf{o}}_h), \hat{\pi}_h(\tilde{\mathsf{o}}_h)\right)\right]$.

To reiterate, Theorem 2 guarantees imitation of the distribution of marginals and final states of $\mathcal{D}_{\text{exp}}$ by replacing the explicit TVC assumption with noising, and the resulting guarantee applies to the *smoothed policy* $\hat{\pi}_\sigma$ which adds smoothing noise back in. Appendix J gives a number of additional results. In Appendix I, we show that the proof framework, outlined in Appendix D, which under lies the proofs of Theorems 1 and 2, is essentially sharp in the worst case. Moreover, in Appendix C.3, we discuss the merits and drawbacks of our use of the synthesis oracle, and how it circumvents some of the challenges encountered in behavior cloning in past work. The key intuition behind the proof of Theorem 2 is depicted in Figure 4, and full proof sketch is deferred to Appendix C.2

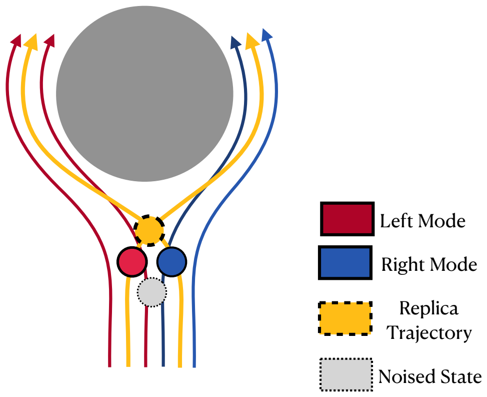

Figure 4: Multi-modal demonstrations traverse an obstacle left or right, exhibiting a pure bifurcation. We consider perturbing expert data on the right mode (blue circle) to a noised datum (gray circle). We show that generative behavior cloners learn to deconvolve this noise, creating a virtual "replica" sample (red circle) following the left mode, such that the replica and original are i.i.d. given the noised one. When the red circle's primitive controllers are rolled from from the blue circle, this leads to a trajectory (yellow circle) which interpolates across the bifurcations. Marginalizing over this process, the yellow trajectories probabilistically interpolate between red and blue modes, and (approximately) match the per-time-step marginal over expert distributions.

4 HINT: Instantiating Data Noising with DDPMs

We now instantiate Theorem 2 by showing that one can learn a policy $\hat{\pi}$ for which the error terms in Theorems 1 and 2 are small by fitting a DDPM to noise-smoothed data. Our proposed algorithm, HINT (Algorithm 1) combines DDPM-learning of chunked policies as in [19] with a popular form of data-augmentation [36]. We collect N_{exp} expert trajectories, synthesize gains, and segment trajectories into observation-chunks o_h and composite actions a_h as described in Section 2. We perturb each o_h to form N_{aug} chunks $\tilde{\mathsf{o}}_h$, as well as horizon indices $j \in [J]$ and inference noises $\boldsymbol{\gamma} \sim \mathcal{N}(0, (\alpha j_h)^2\mathbf{I})$, and add these tuples $(\mathsf{a}_h, \tilde{\mathsf{o}}_h, j_h, \boldsymbol{\gamma}_h, h)$ to our data $\mathfrak{D}$. We end the training phase by minimizing the standard DDPM loss [62]:

Algorithm 1 **H**ierarchical **I**mitation via **N**oising at Inference **T**ime (HINT)

[h]

```
1: Initialize Synthesis oracle $\mathtt{synth}$, sample sizes $N_{\mathsf{exp}}, N_{\mathsf{aug}} \in \mathbb{N}$, $\sigma \geq 0$, DDPM step size $\alpha > 0$,
   DDPM horizon $J$, function class $\{\mathbf{s}_\theta\}_{\theta \in \Theta}$, gain magnitude $R > 0$, empty data buffer $\mathfrak{D} \leftarrow \emptyset$.
   % For no smoothing, set $\sigma = 0$ and $N_{\mathsf{aug}} = 1$
2: for $n = 1, 2, \ldots N_{\mathsf{exp}}$ do
3:     Sample $\boldsymbol{\rho}_T = (x_{1:T+1}, u_{1:T}) \sim \mathcal{D}_{\mathrm{exp}}$ and set $\kappa_{1:T} = \mathtt{synth}(\boldsymbol{\rho})$
      % Segment $\mathsf{o}_{1:H}$ from $\boldsymbol{\rho}_T$ and $\mathsf{a}_{1:H}$ from $\kappa_{1:T}$
4:     for $i = 1, 2, \ldots, N_{\mathsf{aug}}$ and $h = 1, 2, \ldots, H$ do
5:         Sample $\tilde{\mathsf{o}}_h \sim \mathcal{N}(\mathsf{o}_h, \sigma^2\mathbf{I})$, $j_h \sim \mathrm{Unif}([J])$ and $\boldsymbol{\gamma}_h \sim \mathcal{N}(0, (j_h\alpha)^2\mathbf{I})$.
6:         $\mathfrak{D} \leftarrow \mathfrak{D}.\mathrm{append}\left(\{(\mathsf{a}_h, \tilde{\mathsf{o}}_h, j_h, \boldsymbol{\gamma}_h, h)\}\right)$
7: Fit $\theta \in \arg\min_{\theta \in \Theta} \mathcal{L}_{\mathrm{DDPM}}(\theta, \mathfrak{D})$, and let $\hat{\pi} = (\hat{\pi}_h)$ be given by $\hat{\pi}(\cdot \mid \mathsf{o}_h) = \mathtt{DDPM}(\mathbf{s}_{\theta,h}, \mathsf{o}_h)$.
8: return $\hat{\pi}_\sigma = (\hat{\pi}_{\sigma,h})$, by smoothing $\hat{\pi}$ as per Definition 3.5.
```

$$\mathcal{L}_{\mathrm{DDPM}}(\theta, \mathfrak{D}) = \textstyle\sum_{(\mathsf{a}_h, \tilde{\mathsf{s}}_h, j_h, \boldsymbol{\gamma}_h, h) \in \mathfrak{D}} \left|\left|\boldsymbol{\gamma}_h - \mathbf{s}_{\theta,h}\left(e^{-\alpha j}\mathsf{a}_h + \sqrt{1 - e^{-2\alpha j}}\boldsymbol{\gamma}_h, \tilde{\mathsf{o}}_h, j_h\right)\right|\right|^2. \quad (4.1)$$

Our algorithm differs subtly from past work in Line 8: motivated by Theorem 2, we add smoothing noise *back in* at test time. Here, the notation $\mathtt{DDPM}(\mathbf{s}_{\theta,h}, \cdot) \circ \mathcal{N}(\mathsf{o}_h, \sigma^2\mathbf{I})$ means, given o_h, we perturb it to $\tilde{\mathsf{o}}_h \sim \mathcal{N}(\mathsf{o}_h, \sigma^2\mathbf{I})$, and sample $\mathsf{a}_h \sim \mathtt{DDPM}(\mathbf{s}_{\theta,h}, \tilde{\mathsf{o}}_h)$. We now state an informal guarantee for HINT, deferring a formal statement to Appendix C.5.

Theorem (Informal Theorem)**.** Suppose that the system dynamics are smooth and that Assumption 3.1 holds for the linearized system. Then there is a choice of the parameters in HINT that is polynomial in all problem parameters such that for N_{exp}, polynomially large in problem parameters, $\mathcal{L}_{\mathrm{marg},\varepsilon}(\widehat{\pi}_\sigma) \leq \Theta\left(\varepsilon H \sqrt{\tau_{\mathrm{obs}}}(\sqrt{d_x} + \log(1/\varepsilon))\right)$ with high probability.

4.1 Experimental Results

In this section, we demonstrate the benefits of diffusing low level controllers, and of our approach to data noising. We explain the environments in greater detail, along with all training and compu-

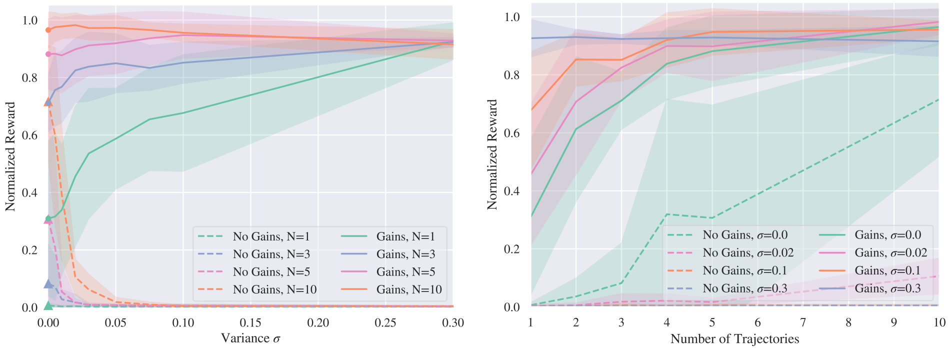

Figure 5: Performance of diffusing chunks of actions $\bar{\mathbf{u}}_{t_{h-1}:t_h-1}$ ("No Gains") versus jointly diffusing actions $u_{t_{h-1}:t_h-1}$, reference states $\bar{\mathbf{x}}_{t_{h-1}:t_h-1}$ and gains $\mathbf{K}_{t_{h-1}:t_h-1}$ for a 2-D quadrotor system with thrust-and-torque-based control. Different noise levels σ and number of trajectories N are shown. Mean and standard deviation are shown across 5 training seeds.

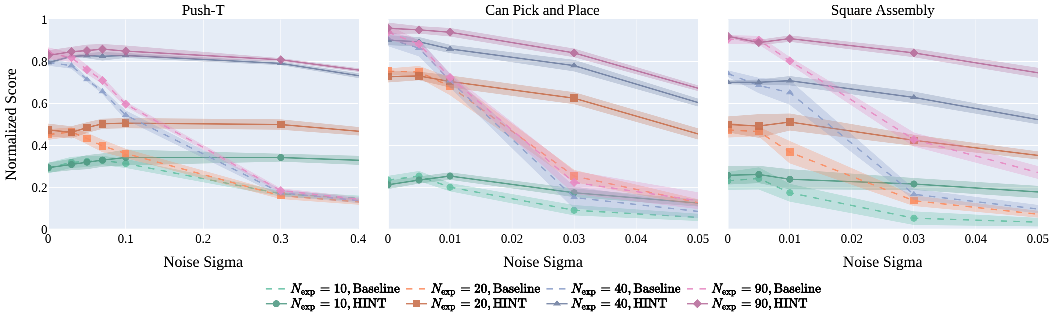

Figure 6: Performance of baseline $\hat{\pi}$ and noise-injected $\hat{\pi} \circ \mathsf{W}_\sigma$ HINT policy for different σ. We use 4 training seeds with 50 and 22 test trajectories per seed for PushT and Can and Square Environments respectively. Mean and standard deviation of the test performance on the 3 best checkpoints across the 4 seeds are plotted. The σ values correspond to noise in the normalized $[-1, 1]$ range.

tational details in Appendix O. [3] Figure 5 compares the performance of diffusing (chunks of) raw control inputs to diffusing (chunks of) gain matrices for a canonical model of a 2-d quadrotor. We find that diffusing gain matrices yields dramatic improvements in performance, in particular allowing a *single imitated trajectory* to outperform learning raw control inputs from 10 demonstrations.

Next, empirically evaluate the effect on policy performance of our proposal to inject noise back into the dynamics at inference time. We consider three challenging robotic manipulation tasks studied in prior work: PushT block-pushing [19]; Robomimic Can Pick-and-Place and Square Nut Assembly [42] (we direct the reader to Chi et al. [19] for an extensive empirical investigation into the performance of diffusion policies in the un-noised $\sigma = 0$ regime). We display the results of our experiments in Figure 6. Observe that the performance degredation of the replica policy from the unnoised $\sigma = 0$ variant is minimal across all environments and even leads to a slight but noticeable improvement in the small-noise regime for PushT (and low-data Can Pick and Place). In the presence of non-negligible noise HINT significantly outperforms the conventional policy $\hat{\pi}$ (obtained by noising observations at training but not test time), as predicted by our theory.

5 Discussion

This work considerably loosened assumptions placed on the *expert distribution* by introducing a synthesis oracle responsible for stabilization. How best to achieve low-level stabilization remains an open question. We hope that this work encourages further empirical research into improving the stability of imitation learning, either via the hierarchical route proposed in this paper or via new innovations.

[3] Code for PushT, Robomimic experiments can be found at `https://github.com/pfrommerd/diffusion_policy_pt`. Quadrotor experiments are in `https://github.com/pfrommerd/stanza/`

Acknowledgments

We would like to thank Cheng Chi for his extensive help in running the code in [19] and Benjamin Burchfiel for numerous insightful discussions and intuitions about DDPM policies. Lastly, we thank Alexandre Amice for his helpful feedback on an earlier draft of this manuscript. AB acknowledges support from the National Science Foundation Graduate Research Fellowship under Grant No. 1122374 as well as the support of DOE through award DE-SC0022199. MS and RT acknowledge support from Amazon.com Services LLC grant; PO# 2D-06310236. DP and AJ acknowledge support from the Office of Naval Research under ONR grant N00014-23-1-2299.

References

[1] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.

[2] A. Ajay, Y. Du, A. Gupta, J. Tenenbaum, T. Jaakkola, and P. Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv preprint arXiv:2211.15657*, 2022.

[3] A. Ajay, A. Gupta, D. Ghosh, S. Levine, and P. Agrawal. Distributionally adaptive meta reinforcement learning. *arXiv preprint arXiv:2210.03104*, 2022.

[4] A. Ajay, S. Han, Y. Du, S. Li, A. Gupta, T. Jaakkola, J. Tenenbaum, L. Kaelbling, A. Srivastava, and P. Agrawal. Compositional foundation models for hierarchical planning. *arXiv preprint arXiv:2309.08587*, 2023.

[5] J. Altschuler and K. Talwar. Privacy of noisy stochastic gradient descent: More iterations without more privacy loss. *Advances in Neural Information Processing Systems*, 35:3788–3800, 2022.

[6] J. M. Altschuler and S. Chewi. Faster high-accuracy log-concave sampling via algorithmic warm starts. *arXiv preprint arXiv:2302.10249*, 2023.

[7] B. D. Anderson and J. B. Moore. *Optimal control: linear quadratic methods*. Courier Corporation, 2007.

[8] O. Angel and Y. Spinka. Pairwise optimal coupling of multiple random variables. *arXiv preprint arXiv:1903.00632*, 2019.

[9] D. Angeli. A lyapunov approach to incremental stability properties. *IEEE Transactions on Automatic Control*, 47(3):410–421, 2002.

[10] M. Bansal, A. Krizhevsky, and A. Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.

[11] P. L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[12] P. L. Bartlett, A. Montanari, and A. Rakhlin. Deep learning: a statistical viewpoint. *Acta numerica*, 30:87–201, 2021.

[13] A. Block, Y. Mroueh, and A. Rakhlin. Generative modeling with denoising auto-encoders and langevin sampling. *arXiv preprint arXiv:2002.00107*, 2020.

[14] A. Block, Y. Mroueh, A. Rakhlin, and J. Ross. Fast mixing of multi-scale langevin dynamics under the manifold hypothesis. *arXiv preprint arXiv:2006.11166*, 2020.

[15] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[16] J. D. Chang, M. Uehara, D. Sreenivas, R. Kidambi, and W. Sun. Mitigating covariate shift in imitation learning via offline data without great coverage. *arXiv preprint arXiv:2106.03207*, 2021.

[17] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

[18] S. Chen, S. Chewi, J. Li, Y. Li, A. Salim, and A. Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022.

[19] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

[20] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and C. Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

[21] P. De Haan, D. Jayaraman, and S. Levine. Causal confusion in imitation learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] R. Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

[23] C. Finn, T. Yu, T. Zhang, P. Abbeel, and S. Levine. One-shot visual imitation learning via meta-learning. In *Conference on robot learning*, pages 357–368. PMLR, 2017.

[24] D. J. Foster and A. Rakhlin. L-infinity vector contraction for rademacher complexity. *arXiv preprint arXiv:1911.06468*, 6, 2019.

[25] J. W. Hagood and B. S. Thomson. Recovering a function from a dini derivative. *The American Mathematical Monthly*, 113(1):34–46, 2006.

[26] P. Hansen-Estruch, I. Kostrikov, M. Janner, J. G. Kuba, and S. Levine. Idql: Implicit q-learning as an actor-critic method with diffusion policies. *arXiv preprint arXiv:2304.10573*, 2023.

[27] A. Havens and B. Hu. On imitation learning of linear control policies: Enforcing stability and robustness constraints via lmi conditions. In *2021 American Control Conference (ACC)*, pages 882–887. IEEE, 2021.

[28] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, et al. Jacob steinhardt et justin gilmer. the many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020.

[29] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[30] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

[31] A. Hussein, E. Elyan, M. M. Gaber, and C. Jayne. Deep imitation learning for 3d navigation tasks. *Neural computing and applications*, 29:389–404, 2018.

[32] A. Hyvärinen and P. Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

[33] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[34] C. Jin, P. Netrapalli, R. Ge, S. M. Kakade, and M. I. Jordan. A short note on concentration inequalities for random vectors with subgaussian norm. *arXiv preprint arXiv:1902.03736*, 2019.

[35] L. P. Kaelbling and T. Lozano-Pérez. Hierarchical task and motion planning in the now. In *2011 IEEE International Conference on Robotics and Automation*, pages 1470–1477. IEEE, 2011.

[36] L. Ke, J. Wang, T. Bhattacharjee, B. Boots, and S. Srinivasa. Grasping with chopsticks: Combating covariate shift in model-free imitation learning for fine manipulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6185–6191. IEEE, 2021.

[37] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8077–8083. IEEE, 2019.

[38] H. Khalil. *Nonlinear Systems*. Pearson Education. Prentice Hall, 2002. ISBN 9780130673893. URL https://books.google.com/books?id=t_d1QgAACAAJ.

[39] I. Kostrikov, D. Yarats, and R. Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

[40] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg. Dart: Noise injection for robust imitation learning. In *Conference on robot learning*, pages 143–156. PMLR, 2017.

[41] H. Lee, J. Lu, and Y. Tan. Convergence of score-based generative modeling for general data distributions. In *International Conference on Algorithmic Learning Theory*, pages 946–985. PMLR, 2023.

[42] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *arXiv preprint arXiv:2108.03298*, 2021.

[43] T. Marcucci, J. Umenberger, P. A. Parrilo, and R. Tedrake. Shortest paths in graphs of convex sets. *arXiv preprint arXiv:2101.11565*, 2021.

[44] A. Maurer. A vector-contraction inequality for rademacher complexities. In *Algorithmic Learning Theory: 27th International Conference, ALT 2016, Bari, Italy, October 19-21, 2016, Proceedings 27*, pages 3–17. Springer, 2016.

[45] M. Mezard and A. Montanari. *Information, physics, and computation*. Oxford University Press, 2009.

[46] D. Misra. Mish: A self regularized non-monotonic activation function. *arXiv preprint arXiv:1908.08681*, 2019.

[47] A. Q. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

[48] T. Pearce, T. Rashid, A. Kanervisto, D. Bignell, M. Sun, R. Georgescu, S. V. Macua, S. Z. Tan, I. Momennejad, K. Hofmann, et al. Imitating human behaviour with diffusion models. *arXiv preprint arXiv:2301.10677*, 2023.

[49] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[50] D. Pfrommer, T. Zhang, S. Tu, and N. Matni. Tasil: Taylor series imitation learning. *Advances in Neural Information Processing Systems*, 35:20162–20174, 2022.

[51] D. Pfrommer, M. Simchowitz, T. Westenbroek, N. Matni, and S. Tu. The power of learned locally linear models for nonlinear policy optimization. *arXiv preprint arXiv:2305.09619*, 2023.

[52] Y. Polyanskiy and Y. Wu. *Information Theory: From Coding to Learning*. Cambridge University Press, 2022+.

[53] M. Raginsky, A. Rakhlin, and M. Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis. In *Conference on Learning Theory*, pages 1674–1703. PMLR, 2017.

[54] A. Rakhlin, K. Sridharan, and A. B. Tsybakov. Empirical entropy, minimax regret and minimax risk. *Bernoulli Society for Mathematical Statistics and Probability*, 2017.

[55] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[56] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[57] S. Ross and D. Bagnell. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 661–668. JMLR Workshop and Conference Proceedings, 2010.

[58] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[59] N. M. Shafiullah, Z. Cui, A. A. Altanzaya, and L. Pinto. Behavior transformers: Cloning k modes with one stone. *Advances in neural information processing systems*, 35:22955–22968, 2022.

[60] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[61] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[62] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[63] Y. Song, S. Garg, J. Shi, and S. Ermon. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*, pages 574–584. PMLR, 2020.

[64] V. Strassen. The existence of probability measures with given marginals. *The Annals of Mathematical Statistics*, 36(2):423–439, 1965.

[65] X. Sun, S. Yang, and R. Mangharam. Mega-dagger: Imitation learning with multiple imperfect experts. *arXiv preprint arXiv:2303.00638*, 2023.

[66] R. Tedrake. Lqr-trees: Feedback motion planning on sparse randomized trees. 2009.

[67] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.

[68] S. Tu, A. Robey, T. Zhang, and N. Matni. On the sample complexity of stability constrained imitation learning. In *Learning for Dynamics and Control Conference*, pages 180–191. PMLR, 2022.

[69] R. Van Handel. Probability in high dimension. Technical report, PRINCETON UNIV NJ, 2014.

[70] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[71] R. Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[72] C. Villani. *Topics in optimal transportation*, volume 58. American Mathematical Soc., 2021.

[73] C. Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

[74] P. Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

[75] M. J. Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

[76] T. Westenbroek, M. Simchowitz, M. I. Jordan, and S. S. Sastry. On the stability of nonlinear receding horizon control: a geometric perspective. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 742–749. IEEE, 2021.

[77] Y. Wu and K. He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[78] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5628–5635. IEEE, 2018.

[79] S. Zhao and A. Grover. Decision stacks: Flexible reinforcement learning via modular generative models. *arXiv preprint arXiv:2306.06253*, 2023.

[80] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.