Sample-Efficient Optimization over Generative Priors via Coarse Learnability

Pranjal Awasthi
Google Research
Mountain View, USA
pranjalawasthi@google.com

Sreenivas Gollapudi
Google Research
Mountain View, USA
sgollapu@google.com

Ravi Kumar
Google Research
Mountain View, USA
ravi.k53@gmail.com

Kamesh Munagala[†]
Department of Computer Science
Duke University, Durham, USA
kamesh@cs.duke.edu

Abstract

We study zeroth-order optimization where solutions must minimize a cost $d(s)$ while maintaining high probability under a complex generative prior $\mathcal{L}(s)$ (e.g., a parameterized model). This reduces to sampling from a target distribution proportional to $\mathcal{L}(s)e^{-T\cdot d(s)}$. Since classical model-based optimization (MBO) lacks finite-sample guarantees for expressive approximate learners, we introduce *coarse learnability*, a flexible statistical assumption requiring only that a learned model covers the target's probability mass within a polynomial factor.

Leveraging this assumption, we design an iterative MBO algorithm called ALDRIFT with a sample correction step that provably approximates the target using only a polynomial number of samples. We apply this framework to globally optimizing non-convex objectives bounded by a quadratic envelope in $\mathbb{R}^n$, where we show this assumption is naturally satisfied for a family of "optimistic" posterior distributions. To reach global ε-optimality, this implies a sample complexity of $\widetilde{O}(\log 1/\varepsilon)$, a rate characteristic of optimistic space-partitioning methods.

We further justify coarse learnability as an assumption for generative priors theoretically, proving that in simple settings, parametric maximum likelihood estimation and over-smoothed kernel density estimators naturally satisfy it.

Finally, one motivation for our framework comes from inference-time alignment. Though our primary contribution pertains to the theoretical foundations of MBO, we provide qualitative evidence that, in simple settings, even primitive LLMs can shift their distributions toward lower-cost regions when fine-tuned with zeroth-order feedback.

1 Introduction

Model-based optimization (MBO) is a broadly applicable framework for zeroth-order optimization: a probabilistic model over candidate solutions is iteratively updated using function evaluations, guiding search toward high-performing solutions. Classical MBO methods such as the cross-entropy method [41] are well-studied and widely used, but their theoretical guarantees are asymptotic (as opposed to finite sample) and rely on strong structural assumptions. These assumptions break down when the generative model is an expressive approximate learner, such as a neural network, leaving

[†]This work was done while the author was visiting Google Research.

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

the finite-sample behavior of MBO in this regime theoretically uncharacterized. As discussed more in Appendix E, a motivating instance is inference-time optimization with learned generative priors, including language models, where one seeks to enforce global objectives while retaining high prior likelihood under the generative model. While analyzing MBO over modern generative models remains challenging, establishing finite-sample guarantees in tractable, continuous domains is a necessary first step. This requires more flexible assumptions on the generative model, which is the focus of this paper.

1.1 Model-Based Optimization (MBO) with a Generative Prior

We study a zeroth-order optimization framework, where the goal is to minimize a function $d(\cdot)$ given only oracle access to its values. The function $d(\cdot)$ might represent a continuous non-convex loss surface, or a discrete penalty for violating global constraints. To guide this search, we assume access to a generative prior $\mathcal{L}(\cdot)$ and our objective is to sample effectively from a target distribution of the form $p_T(s) \propto \mathcal{L}(s) \cdot e^{-T \cdot d(s)}$, where T is a 'temperature' parameter. Samples from this distribution naturally balance low values of $d(s)$ with high probability under the prior $\mathcal{L}(s)$. This formulation is versatile: in continuous domains, the prior provides regularization to escape local minima (see Appendix C); in combinatorial settings, it allows a division of labor where an external checker enforces global feasibility via $d(\cdot)$ while a generative model enforces semantic priors via $\mathcal{L}(\cdot)$.

Our primary object in this paper is sampling from p_T; approximate optimization of d follows by choosing T appropriately large. For such T, the target distribution p_T can become *exponentially far* from the initial prior $\mathcal{L}$, making direct sampling or simple rejection-based corrections infeasible.

Sample-Efficient Model-Based Optimization. To address this sampling challenge, we adopt the perspective of model-based optimization (MBO), most notably the cross-entropy method [41] and model reference adaptive search [18]. (Section 1.2 and Appendix G.2 contain a detailed comparison.) These methods, which are widely used for derivative-free non-convex optimization [45, 48], optimize an objective by iteratively updating a probabilistic or neural search distribution that concentrates on the optimum. However, classical guarantees for these methods typically rely on asymptotic convergence arguments via structural properties of specific surrogates, such as natural exponential families [18]. These results do not extend to the finite-sample regime, either with well-structured priors or with generative models that function as approximate learners. Our work bridges this gap by providing polynomial sample complexity bounds for this class of algorithms under a general *coarse learnability* assumption, replacing rigid structural constraints with a statistical coverage condition.

Simulated Annealing and ALDRIFT. We operationalize this framework via ALDRIFT (*Algorithm Driven Iterated Fitting of Targets*), an iterative algorithm in which a generative model is progressively refined. We construct an intermediate distribution $p_\tau(s) \propto \mathcal{L}(s) \cdot e^{-\tau \cdot d(s)}$ and detail how ALDRIFT treats τ as a (inverse) temperature parameter—drawing inspiration from simulated annealing [21, 24, 25]—and gradually increases it to T to assign more weight to the objective d (see Section 3). Crucially, at each iteration, ALDRIFT performs a Metropolis–Hastings step to generate samples from the current target distribution $p_\tau(s)$, using the learned generative model as a proposal. In this setting, using importance [53] or rejection sampling [20] would require exponentially many samples. Metropolis–Hastings circumvents this by relying solely on *relative* likelihood ratios, allowing the procedure to remain efficient even when the target and initial distributions differ exponentially.

The resulting samples are then used to fit the new generative model (via say parameter estimation) so that it produces samples that are more closely aligned with $p_\tau(s)$. At a high level, this approach is analogous to related methods in convex optimization [24], to approximate policy-iteration schemes from reinforcement learning [1, 23, 37, 38, 44], and to sequential MCMC [2, 7, 10, 34], though the details and objectives differ. Further, as discussed in Appendix G.2, ALDRIFT is an instance of the Model Reference Adaptive Search (MRAS) framework [18], and our algorithm and analysis can be interpreted as a finite-sample robustification of MRAS via the Metropolis–Hastings proposal step.

Formal Guarantees via Coarse Learnability. Our main theoretical contribution in Section 4 is a proof that ALDRIFT converges to the target distribution $p_T(s)$ with polynomial sample complexity, under a novel coverage assumption we call *coarse learnability*. (See Assumption 2 in Section 4 for a formal statement.) This assumption formalizes approximate learning as follows: given m samples from a target distribution, we can learn a generative model whose density envelopes the target density within a factor of $\text{poly}(m)$, except on an exponentially small error set.

A central contribution of our work is the formulation of coarse learnability itself. At a conceptual level, as discussed in Section 4, coarse learnability identifies a sufficient statistical condition under which any sampling-based learner can provably refine a generative model toward a target distribution. As we discuss in Section 1.2, coarse learnability is related to the notion of *coverage* in reinforcement learning [51, 62], which similarly quantifies when a policy explores enough of the optimal state-action space to permit efficient improvement. Coarse learnability differs by treating coverage as a dynamic statistical outcome of the model fitting instead of as a static assumption, regenerating the required coverage dynamically at each iteration.

Provable Bounds for Non-Convex Global Optimization. Section 4 gives a general finite-sample guarantee for ALDRIFT under Assumption 2. In Appendix C, we show the utility of this result in the classical setting of ε-approximately globally optimizing a non-convex function in $\mathbb{R}^n$, where prior results only show asymptotic convergence of MBO with infinitely many samples [18]. In contrast, we prove that on a well-studied class of quadratically bounded non-convex functions [6, 33] with known curvature bounds, ALDRIFT is sample-efficient. Unlike existing MBO methods that fit an exact exponential family model, we fit a wider Gaussian posterior, analogous to optimistic exploration in bandit literature [6]. This key difference enables us to verify Assumption 2 in this setting, and show that a strengthening of ALDRIFT achieves sample complexity $\widetilde{O}(\log 1/\varepsilon)$, with dimension- and geometry-dependent constants. This rate is characteristic of optimistic space partitioning methods [6, 33] (see Appendix C for some caveats) and is exponentially better than that achievable by local MCMC methods [15, 17]. We note that for this sample-complexity guarantee, Assumption 2 is no longer an assumption, as it is rigorously verified. Complementing our theory, we present an empirical demonstration of the need for the inflated proposal and sample correction steps with limited sample budgets in Section 6.

Theoretical Plausibility of Coarse Learnability. In Appendix D, we present theoretical evidence justifying Assumption 2 for certain generative models. We demonstrate that standard MLE naturally induces the required coverage properties in several simplified settings: the *agnostic* setting where the model is misspecified (e.g., a unimodal learner approximating a multimodal target) and precise density approximation is impossible; for the *realizable* setting for well-behaved exponential families; and *non-parametric settings* such as standard Kernel Density Estimation (KDE).

Motivating Application: Inference-time Alignment. Though our primary focus is to theoretically formalize the mechanics of MBO under generative priors, we note in Appendix E that one motivating application is the combination (via inference-time alignment and fine-tuning) of classical combinatorial algorithms with generative models [8, 20, 27] to solve problems that require both global feasibility and adherence to local priors: route planning with qualitative scenic preferences as prior, molecular design with learned drug-likeness priors, or scheduling with semantic session coherence. In such problems, the generative model encodes local constraints that a classical algorithm cannot capture, while the algorithm enforces global structure that a generative model cannot reliably guarantee. LLMs are a natural family of priors for this purpose, though, as we illustrate in Appendix E, they can fail to enforce global combinatorial properties on their own, confirming the need for the hybrid approach.

We present qualitative results in Appendix F demonstrating that even a relatively primitive LLM can iteratively acquire combinatorial constraints (initially unknown to it) from only a few samples via fine-tuning. Specifically, we show that for a simple scheduling problem and for low degree spanning trees, when guided by a heuristic version of ALDRIFT called TOPIFT, an LLM (GPT-2 [40]) can adapt its generative distribution to favor solutions with low $d(s)$ values while still respecting its prior, hence qualitatively demonstrating the "mass covering" type behavior required for coarse learnability.

While our formal guarantees do not yet apply to inference-time alignment directly (and likely require relaxed formulations), our aim is to introduce an MBO framework that makes the behavior of expressive and potentially mis-specified models more amenable to analysis. By identifying coverage as a key driver of convergence, we provide a possible theoretical lens for reasoning about how generative models could support combinatorial optimization under appropriate coverage conditions.

Summary of Contributions. We summarize our primary contributions as follows:

- **Conceptual Framework:** We introduce *coarse learnability* (Assumption 2) as a sufficient coverage-style condition to establish finite-sample guarantees for iterative Model-Based Optimization (MBO) using expressive approximate learners.

- **MBO Algorithm:** We propose ALDRIFT (Section 3.2), an MBO algorithm that pairs iterative learning of generative models with a principled sampling correction step (e.g., Metropolis–Hastings) to control the accumulation of approximation errors.
- **General Theoretical Guarantee:** We prove that under Assumption 2, ALDRIFT achieves polynomial finite-sample convergence to the target p_T (Theorem 1).
- **Unconditional Instantiation:** To ground our framework, we analyze ALDRIFT in the canonical setting of ε-globally optimizing non-convex objectives bounded by a quadratic envelope in $\mathbb{R}^d$ in Appendix C. By employing an optimistic Gaussian proposal, we show that coarse learnability holds in this setting and obtain a sample complexity of $\widetilde{\mathcal{O}}(\log 1/\varepsilon)$.
- **Supporting Evidence:** Finally, we provide proof-of-concept theoretical and qualitative empirical evidence (Appendices D and F) demonstrating respectively that certain simple generative models satisfy coarse learnability and that even primitive LLMs can exhibit the mass-covering behavior required by our theoretical framework on combinatorial tasks.

1.2 Related Work

Model-Based Optimization and MRAS. Our approach is formally an instance of the model reference adaptive search (MRAS) framework [18], which generalizes the cross-entropy method [41]. MRAS operates by using a parametric model to approximate a sequence of ideal target distributions that increasingly concentrate on the optimum. In each epoch, it generates samples from the current parametric model, selects the high-performing "elite" samples, and updates the model parameters to fit this elite set. While MRAS is proven to converge in the limit to the globally optimal solution when the parametric distributions are from an exponential family, it lacks sample complexity bounds, both for simple parametric families as well as for expressive approximate learners, which is the focus of our paper. We present this connection in more detail in Appendix G.2.

Our work also bears similarity to reinforcement learning algorithms such as reward-weighted regression [38] and relative entropy policy search [37]. Huang et al. [19] analyze language-model sharpening through supervised fine-tuning and reinforcement learning, assuming that the model class can represent the corresponding target distributions. Their framework includes KL-regularized objectives whose optimal distributions have the same form as our targets. Our framework allows these targets to lie outside the learned model family. Under coarse learnability, we obtain polynomial sample-complexity bounds by using the learned models as proposals and correcting their sampling bias via Metropolis–Hastings.

Sequential Monte-Carlo Sampling. The annealing structure of ALDRIFT, progressing through a sequence of tempered distributions $p_\tau(s) \propto \mathcal{L}(s)e^{-\tau d(s)}$ with Metropolis–Hastings corrections at each step, shares a foundational skeleton with Annealed Importance Sampling (AIS; [34]) and Sequential Monte Carlo (SMC; [2, 10]). However, these methods maintain a discrete empirical measure over particles, while ALDRIFT introduces a parametric projection step: it fits a generative model to the particles at each temperature, which is then used as the proposal for the next step. In effect, this replaces the variance associated with importance weights and particle degeneracy with approximation bias, as the fitted model $\mathcal{L}_\tau$ may not faithfully cover the target p_τ. While related ideas appear in adaptive and amortized sampling methods, this projection step introduces a distinct source of error not present in classical particle-based schemes, which we address via the coarse learnability assumption.

Population Monte Carlo (PMC; [7]) shares a similar overall structure to ALDRIFT. It iteratively fits a parametric proposal to a particle population; however, in our setting, the variance in weights can explode. ALDRIFT circumvents this by defining a sequence of intermediate targets and utilizing Metropolis–Hastings to draw unweighted samples from them. Consequently, our contribution is to present sufficient conditions under which such unweighted sampling has polynomial complexity.

Furthermore, while recent work on amortized MCMC via transport maps similarly focuses on "learning to propose" to improve acceptance rates [4, 16, 36], we introduce a finite-sample coverage assumption (Assumption 2) rather than relying on accuracy of the learned transport map to improve sampling efficiency.

Learnability and Coverage. In contrast to classical statistical learning theory concepts such as Probably Approximately Correct (PAC) learning [55], our paper introduces the notion of coarse

learnability (Assumption 2). This concept is loosely related to the coverage condition studied in reinforcement learning [51, 62, 63] and recent work on inference-time alignment [20]. In those settings, bounded coverage is typically a static assumption on the data distribution or behavior policy, ensuring sufficient overlap with the optimal policy to permit valid estimation. Coarse learnability plays a similar structural role, but differs by treating coverage as a statistical outcome of the model fitting. It posits that given polynomial samples from a target, the generative model learns to cover that target (within polynomial factors), effectively regenerating the required coverage dynamically at each iteration.

2 Optimization Framework with a Generative Model

We consider the problem of minimizing a global objective function $d : S \to \Re$, where S is the solution space.[2] As is standard in MCMC literature, we assume w.l.o.g. that $d(s) \in [0, D]$ for all $s \in S$. We assume only zeroth-order oracle access to $d(s)$: we can evaluate $d(s)$ for any $s \in S$, but do not have access to its gradients. In our canonical applications (e.g., Appendix F), $d(\cdot)$ can typically be computed efficiently by an algorithm. In addition, we assume access to a prior distribution $\mathcal{L}(s)$. This prior $\mathcal{L}(s)$ can be modeled by a generative distribution, such as that provided by parametrized or generative models.

Our goal is to identify solutions s that attain low values of $d(s)$ while also having high probability under $\mathcal{L}(s)$. This objective can be formulated as sampling from, or finding the modes of, the target distribution:

$$p_T(s) \propto \mathcal{L}(s) \cdot e^{-T \cdot d(s)},$$

where $T > 0$ is a 'temperature' parameter that controls how sharply $p_T(s)$ concentrates on solutions s that minimize $d(s)$ among those with significant probability mass under $\mathcal{L}(s)$.

This sampling problem is particularly interesting when the generative model $\mathcal{L}$ is very unlikely to generate solutions with small $d(\cdot)$, and when a random solution s with low $d(s)$ is also very unlikely to have relatively high probability under $\mathcal{L}$ compared to a carefully chosen s. We assume that given samples from any distribution $\mathcal{D}$, we can use these samples to learn a distribution that is close to $\mathcal{D}$, for instance, via parameter-fitting. We will make the learning assumption more precise later, and this will be one of our key contributions.

In the sequel, we use $s \sim \mathcal{L}$ to denote that the sample s is generated by a model $\mathcal{L}$. For distributions P, Q, let $d_{\mathrm{tv}}(P, Q)$ denote the *total variation distance* between them.

3 ALDRIFT: An Algorithm with Provable Guarantees

We now develop our algorithm ALDRIFT for solving the sampling problem above, with provable guarantees. Before presenting the algorithm, we first discuss a natural heuristic, TOPIFT, which highlights key technical challenges in obtaining formal bounds and thereby motivates both our algorithm and the assumptions underlying it.

3.1 A Simple Heuristic: TOPIFT

To build intuition for why our sampling problem is non-trivial, we begin with a natural heuristic, TOPIFT, which iteratively refines the model by sampling from the current distribution, selecting the lowest-cost samples, and fitting a model to this elite set. Although effective empirically, TOPIFT highlights the challenges that motivate the principled algorithm, ALDRIFT (Section 3.2).

At iteration r, the heuristic draws $m \cdot M$ samples from the current model $\mathcal{L}_{r-1}$, evaluates the zeroth-order cost $d(s)$ for each sample, selects the best m samples, and learns $\mathcal{L}_{r-1}$ using this set to obtain $\mathcal{L}_r$. The cost function $d(\cdot)$ thus *guides* the refinement of the model in conjunction with the prior.

The heuristic in Algorithm 1 is closely related to the cross-entropy (CE) method [41], where an "elite" set of high-performing samples is repeatedly used to update a parametric model via MLE. In our setting, the generative model plays the role of the parametric family, and model-fitting corresponds to

[2] In practice, S is conditioned on the prompt string x, which encodes the local and global constraints of the problem instance; we omit x from the subsequent discussion.

Algorithm 1 TOPIFT: Iterated Fitting of Targets Using Top Samples.

Input: m, M, Q
Data: $\mathcal{L}$ (initial model)

$\mathcal{L}_0 \leftarrow \mathcal{L}$; ▷ initialize
for $r \leftarrow 1$ **to** Q **do**
  sample S_r from $\mathcal{L}_{r-1}$ with $|S_r| = m \cdot M$; ▷ generate samples
  evaluate $d(s)$ for all $s \in S_r$; ▷ compute cost
  $S'_r \leftarrow m$ samples in S_r with smallest $d(s)$; ▷ elite set
  $\mathcal{L}_r \leftarrow$ fit a model to the samples S'_r ; ▷ update model
end
return $\mathcal{L}_Q$

the MLE update. However, CE-style updates are difficult to analyze, since the hard-min elite-selection step can induce mode collapse and eliminate coverage of important regions.

These difficulties manifest directly in TOPIFT: Its hard-min selection step is analytically brittle, and repeated model-fitting without explicit control over density ratios can progressively erode probability mass on rare but critical regions. ALDRIFT resolves these issues by replacing hard-min with a controlled annealing schedule and by introducing Metropolis–Hastings corrections (Section 3.2), ensuring that samples at each iteration follow the correct intermediate target distribution; this prevents coverage errors from compounding across iterations. We present the connection between ALDRIFT and TOPIFT in more detail in Appendix G.1, and empirically demonstrate the need for iterated distribution correction in Section 6.

To implement the Metropolis–Hastings correction and hence show our theoretical results, we require the following assumption. This assumption holds whenever the generative model admits tractable likelihood evaluation, including many autoregressive sequence models and open weight models.

Assumption 1. *For any $s \in S$, the probability value $\mathcal{L}(s)$ is available. This must hold not only for the initial model but also for any intermediate models obtained.*

3.2 ALDRIFT Algorithm

Our final algorithm, ALDRIFT, is presented in Algorithm 2. The input to ALDRIFT is the number of samples ($m = \text{poly}(T, D)$), and the parameter T. We set $M = m^2$. Recall that $D = \max_{s \in S} d(s)$.

In this algorithm, we intend the model $\mathcal{L}_\tau(s)$ to be an approximation to $p_\tau(s) \propto \mathcal{L}(s) \cdot e^{-\tau \cdot d(s)}$, where the temperature parameter τ will gradually increase from 0 to T. The main difference between ALDRIFT and TOPIFT is that the probability of retaining a sample is adjusted based on the target distribution p_τ. This adjustment is an interesting application of the Metropolis–Hastings algorithm [31], which runs the IMH chain for M steps until mixing. We require Assumption 1 for both $\mathcal{L}$ and the intermediate learned models $\mathcal{L}_\tau$, in particular to enable MCMC sampling from p_τ given a model for p_{τ^-}. As discussed in Section 4, this prevents errors in model learning from accumulating across iterations.

Remarks. In Algorithm 2, we use a single complexity parameter m to control multiple aspects of the algorithm for simplicity of analysis. As stated in Assumption 2, m primarily serves as a learnability parameter, i.e., the number of samples required. We also use m to determine both the number of Metropolis–Hastings iterations (running the step m times) and the length of each iteration ($M = m^2$). This ensures that the computational effort per simulated annealing step scales polynomially with the learnability parameter. Further, in certain cases like the setting in Appendix C and Section 6, the dependence of the run-time on D can be removed via a geometric annealing schedule. We also note that the model-fitting step is efficient in practice if we use the parameters of $\mathcal{L}_{\tau^-}$ as an efficient starting point to learn $\mathcal{L}_\tau$, preventing the need to learn from scratch in each iteration. Finally, Algorithm 2 is an instance of Model Reference Adaptive Search (MRAS) [18], which generalizes the cross-entropy method [41]. We present this connection in Appendix G.2.

Algorithm 2 ALDRIFT: Algorithm Driven Iterated Fitting of Targets.

Input: m, T
Data: $\mathcal{L}$ (base model), D (temperature step denominator)

$\mathcal{L}_0 \leftarrow \mathcal{L}$ $\tau \leftarrow 0$ $M \leftarrow m^2$; ▷ choice of M justified in Theorem 1.

while $\tau \leq T$ **do**

  $\tau\text{-} \leftarrow \tau$; $\tau \leftarrow \min\left(\tau + \frac{1}{D}, T\right)$; $S_\tau \leftarrow \varnothing$; ▷ initialize sample set at this temperature

  for $k \leftarrow 1$ **to** m **do**

    Sample $s_0 \sim \mathcal{L}_{\tau\text{-}}$; $w_\tau(s_0) \leftarrow \mathcal{L}(s_0) \cdot e^{-\tau \cdot d(s_0)}$; ▷ τ- is the previous value of τ

    for $i \leftarrow 1$ **to** M **do**

      Sample $\hat{s}_i \sim \mathcal{L}_{\tau\text{-}}$; $w_\tau(\hat{s}_i) \leftarrow \mathcal{L}(\hat{s}_i) \cdot e^{-\tau \cdot d(\hat{s}_i)}$; ▷ weight; note $p_\tau(s) \propto w_\tau(s)$

      $\beta_i \leftarrow \min\left(1, \frac{w_\tau(\hat{s}_i)}{\mathcal{L}_{\tau\text{-}}(\hat{s}_i)} \cdot \frac{\mathcal{L}_{\tau\text{-}}(s_{i-1})}{w_\tau(s_{i-1})}\right)$; ▷ MH acceptance probability

      if $u \sim \mathrm{Uniform}(0,1)$ *satisfies* $u \leq \beta_i$ **then**

        $s_i \leftarrow \hat{s}_i$

      else

        $s_i \leftarrow s_{i-1}$

      end

    end

    $S_\tau \leftarrow S_\tau \cup \{ s_M \}$; ▷ append final sample into multiset of samples

  end

  $\mathcal{L}_\tau \leftarrow$ fit a model to the samples S_τ; ▷ $\mathcal{L}_\tau$ is the learned model for p_τ

end

return $\mathcal{L}_T$

4 Coarse Learnability and Sample Complexity of ALDRIFT

We will analyze the sample complexity of ALDRIFT in Section 4. To do this, we need an assumption on the learning agent. In Section 4, we propose the novel "coarse learnability" assumption, which essentially says that if we can approximately generate samples from p_τ, then we can coarsely learn a model $\mathcal{L}_\tau$ for it using the samples. Coarse learnability asks only that model fitting produce a proposal with tails heavy enough to cover the target up to polynomial density ratios (in the number of samples used) on all but exponentially rare target mass.

For two distributions p and L on the same support S, we define the *coverage* of L with respect to p at a point $s \in S$ as $\mathrm{cov}_{p,L}(s) = \frac{p(s)}{L(s)}$. This notion, similar to analogous notions in reinforcement learning and inference-time alignment (e.g., [20, 62]), quantifies how well the proposal distribution L "covers" the probability mass of the target p: large values indicate under-coverage, while small values indicate over-coverage.

Assumption 2 (Coarse Learnability)**.** *For each temperature $\tau \in [0, T]$, let $p_\tau(s) \propto \mathcal{L}_0(s) \cdot e^{-\tau d(s)}$ be the target distribution and $\mathcal{L}_\tau$ the model obtained after fitting. We say the model family satisfies* coarse learnability *if for any large enough $K \geq \mathrm{poly}(D, T)$ and any error $\varepsilon \in (0,1)$ and confidence $\delta \in (0,1)$, there is a sample size $m = \mathrm{poly}(K/\delta, \ln 1/\varepsilon)$ such that the following holds:*

If $\mathcal{L}_\tau$ is trained using m samples from a distribution $\hat{p}$, then with probability at least $1 - \delta$ (over the training samples), the learned model satisfies:

$$\Pr_{s \sim p_\tau}\left[\mathrm{cov}_{p_\tau, \mathcal{L}_\tau}(s) > K\right] \leq O\left(d_{\mathrm{tv}}(\hat{p}, p_\tau)\right) + \varepsilon. \tag{1}$$

The key distinction between Assumption 2 and standard density estimation or PAC learning [55] lies in the error regime. Standard guarantees typically yield error rates polynomial in the sample size ($\varepsilon \approx \mathrm{poly}(1/m)$). In contrast, our sample complexity m scales with $\ln(1/\varepsilon)$, which means polynomial samples yield *exponentially small* error probabilities ($\varepsilon \approx e^{-m}$), albeit for a coarse coverage guarantee. This stronger tail bound is needed for the following reason. Since we assumed the global optimum could have exponentially small probability under the initial prior, a learner with only polynomial error guarantees could validly assign vanishing probability to the optimal region.

The exponential tail bound ensures that the intermediate models preserve coverage of these rare regions throughout the annealing schedule.

To build intuition, we consider the following simplification. Assume $\hat{p} = p_\tau$. For given K, set $\varepsilon = e^{-K}$ and $\delta = 1/K$. Assumption 2 now implies $K = m^\beta$ for some constant $\beta \in (0, 1)$. If we re-parametrize in terms of m, we have $\varepsilon = e^{-m^\beta}$. In this regime, $K \leq m$, so that we can use m instead of K in Eq. (1), which yields $\Pr\left[\mathrm{cov}_{p_\tau, \mathcal{L}_\tau}(s) > m\right] \leq \varepsilon = e^{-m^\beta}$. Then, Assumption 2 simplifies to the following property for general parametric models and targets:

> **Assumption 2, special case.** A model $\mathcal{L}$ coarsely learns a parametric distribution p if for sufficiently large sample size m that depends polynomially on the parameters of the distribution, w.p. $1 - 1/m^\beta$, the model $\mathcal{L}^m$ trained on m samples from p satisfies $\Pr\left[\mathrm{cov}_{p, \mathcal{L}^m}(s) > m\right] \leq e^{-m^\beta}$, where $\beta > 0$ is a small constant.

Coarse learnability guarantees polynomial coverage of the target density, except on an atypical set of exponentially small measure. Unlike density estimation guarantees that require the total variation distance to vanish, this condition allows for persistent approximation errors (model mis-specification; see Appendix D.1). This extends coverage-based analyses [20] to the iterative setting, where as long as the coverage remains polynomially bounded, the Metropolis–Hastings correction can recover the exact target distribution, bridging regimes where the initial coverage gap is exponentially large via intermediate polynomial steps. While this appears to be a strong assumption, such intuition is implicit in classical MBO [18, 41], and is fundamentally linked to statistical estimators that prioritize heavy tails (e.g., over-smoothed KDE). In Section 6, we empirically demonstrate a non-convex optimization setting where an inflated proposal with distribution correction allow for the global convergence of ALDRIFT when sample budgets are small. In Appendix D, we provide evidence supporting the plausibility of Assumption 2 for well-known generative distributions.

Our main analytical result analyzes the sample complexity of ALDRIFT under coarse learnability.

Theorem 1 (Proved in Appendix B)**.** *Under Assumption 2, for sufficiently large $m = \mathrm{poly}(T, D)$ the sample complexity of* ALDRIFT *is $O(mMTD) = \mathrm{poly}(T, D)$, and with high probability, the sampling distribution $\hat{p}$ of S_T satisfies*[3] *$d_{\mathrm{tv}}(\hat{p}, p_T) = O\left(e^{-m^\alpha}\right)$, where $\alpha > 0$ is a constant.*

5 Summary of Theoretical Results

Sample Complexity of ALDRIFT for Non-Convex Optimization. Theorem 1 shows that ALDRIFT has polynomial finite-sample complexity under Assumption 2. In Appendix C, we verify this assumption for a family of Gaussian priors on a canonical class of non-convex objectives bounded by a quadratic envelope (as studied in [6, 33]). We show that this implies that a strengthening of ALDRIFT via rejection sampling and geometric annealing has $\widetilde{O}(\log 1/\varepsilon)$ sample complexity for finding the ε-globally optimal solution, matching the rate of space partitioning methods.

Specifically, we assume the state space is $\mathbb{R}^n$ and the generative prior is a Gaussian $\mathcal{L}_0(x) \propto \exp(-U(x))$, where $U(x) = \frac{\mu_0}{2}\|x - x_0\|^2$ for some $\mu_0 > 0$. Assume the non-convex objective $d(x) \geq 0$ has a unique global minimum at x^* with $d(x^*) = 0$ w.l.o.g. We assume $d(x)$ is bounded by a quadratic envelope around its true optimum: there exist constants $\mu_d, L_d > 0$ such that for all $x \in \mathbb{R}^n$, $\frac{\mu_d}{2}\|x - x^*\|^2 \leq d(x) \leq \frac{L_d}{2}\|x - x^*\|^2$. Note that $d(x)$ is not assumed to be convex and may contain arbitrary local non-convexities within these bounds. This model is well-studied in non-convex optimization; see for instance [6, 33]. We assume the curvature bounds are known.

We run ALDRIFT for N temperature settings as described in Appendix C.3. At each temperature τ, the target distribution is $p_\tau(x) \propto \exp(-f_\tau(x))$, where $f_\tau(x) = U(x) + \tau d(x)$. The algorithm draws m samples from p_τ, computes the empirical mean $\hat{\omega}_\tau$, and fits the proposal distribution $\mathcal{L}_\tau(x) = \mathcal{N}(\hat{\omega}_\tau, \frac{2}{\mu_\tau} I_n)$, where $\mu_\tau = \mu_0 + \tau\mu_d$. It is crucial for Assumption 2 that the distribution has larger variance than desired, hence deviating from standard MBO that uses the variance of the samples. This analog of "optimistic exploration" represents one of our algorithmic contributions, and we present an empirical demonstration of its necessity with small sample budgets in Section 6.

Our main result shows that Assumption 2 holds at all intermediate steps.

[3] Since p_τ is only coarsely learnable via $\mathcal{L}_\tau$, we need to consider the d_{tv} with respect to the sampling distribution S_τ, which corrects this coarse learning error via the Metropolis–Hastings procedure.

Theorem 2 (Theorem 5 in Appendix C)**.** *If the per-temperature sample size satisfies* $m = \Omega(n(n + \ln(N/\delta)))$*, then with probability at least* $1 - \delta$*, each model* p_τ *satisfies* $\sup_\tau \sup_{x \in \mathbb{R}^n} \frac{p_\tau(x)}{\mathcal{L}_\tau(x)} \leq K$ *for a* K *that only depends exponentially on* n *and the convexity parameters, but is independent of the temperature* τ*.*

We show that to achieve an expected objective error ε, it suffices to set the final temperature to $T = O(1/\varepsilon)$. We now use a strengthening of ALDRIFT via geometric annealing and rejection sampling to exactly sample the target. We show that the total number of samples is $\widetilde{O}(\log 1/\varepsilon)$, where the constants depend on n (exponentially so), K, and $\log \frac{\log 1/\varepsilon}{\delta}$. We compare this result with existing methods in Appendix C.4, showing it matches the sample complexity of space partitioning [6, 33], though unlike these methods, ALDRIFT requires prior estimates of the curvature parameters μ_d, L_d, and its sample complexity scales with the dimension n, whereas space-partitioning bounds scale with the near-optimality dimension, which could be smaller. On the other hand, the ALDRIFT method generalizes more easily to other priors, such as semantic or discrete spaces.

Theoretical Evidence for Coarse Learnability. Complementing the sample complexity bounds in Appendix C, in Appendix D, we present theoretical evidence to support the coarse learnability assumption (Assumption 2) for various types of simple generative models and settings. First, in Appendix D.1 we prove that a structurally misspecified learner (a single Gaussian) satisfies the condition when approximating a multimodal target if the number of modes k is a constant.

Theorem 3 (Theorem 6 in Appendix D.1)**.** *Let the target* $p(x) = \frac{1}{k}\sum_{i=1}^{k} \mathcal{N}(\mu_i, 1)$ *be a mixture of* k *unit-variance Gaussians with means* $\mu_i \in [-\frac{\Delta}{2}, \frac{\Delta}{2}]$*, where* $\Delta \geq 1$*. Assume the modes are sufficiently separated such that the population variance satisfies* $\sigma^{*2} \geq 2$*. Let* $\mathcal{L}_{\mathrm{MLE}}$ *be the single Gaussian model* $\mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$ *learned via MLE on* m *i.i.d. samples from* p*. If* $m = \Omega(\Delta^4 \ln(1/\delta))$*, then with probability at least* $1 - \delta$*, the learned model satisfies* $\sup_{x \in \mathbb{R}} \frac{p(x)}{\mathcal{L}_{\mathrm{MLE}}(x)} = O(\Delta \cdot e^k)$*.*

Second, to validate consistency with classical MBO theory, we show that standard MLE in the *realizable* setting of MRAS (i.e., exponential families) satisfies the assumption with high probability (Appendix D.2). Finally, in Appendix D.3 we demonstrate that coarse learnability extends to non-parametric estimators (like certain KDE) in constant dimensions, provided the kernel is deliberately "over-smoothed" to act as a robust coverage envelope.

Theorem 4 (Theorem 8 in Appendix D.3)**.** *Let the target distribution* $p(x) = \mathcal{N}(0, \sigma^2 I_d)$ *be a* d*-dimensional isotropic Gaussian. Let* $\mathcal{L}_{KDE}(x) = \frac{1}{m}\sum_{i=1}^{m} K_h(x - X_i)$ *be the density learned via KDE using a Gaussian kernel* $K_h(z) = (2\pi h^2)^{-d/2} \exp(-\|z\|^2/2h^2)$ *on* m *i.i.d. samples from* p*. Assume we set the bandwidth to match*[4] *the target's tail decay,* $h = \sigma$*. If the sample size* $m = \Omega(d + \ln(1/\delta))$*, then with probability at least* $1 - \delta$*, we have:* $\sup_{x \in \mathbb{R}^d} \frac{p(x)}{\mathcal{L}_{KDE}(x)} \leq 8e^d$*.*

6 Empirical Illustration of ALDRIFT versus TOPIFT

Although our contributions are primarily theoretical, we present a focused empirical demonstration to validate the necessity of an optimistic, variance-inflated proposal distribution combined with sample correction. We compare ALDRIFT against TOPIFT, a baseline that relies on empirical variance without distribution correction and consequently suffers from premature convergence to local optima. While specialized heuristics exist to navigate specific non-convex landscapes, our objective here is not to achieve state-of-the-art empirical performance. Rather, our goal is to cleanly isolate and verify how the core theoretical insights of our framework hold up under strict sample budgets.

Objective Landscape. We study the non-convex optimization setting from Appendix C and define a $n = 10$ dimensional objective $d(\cdot)$ characterized by a global quadratic envelope and high-frequency local minima. For $x \in \mathbb{R}^{10}$, let $d(x) = \frac{\mu_d}{2}\|x\|^2 + A\sum_{i=1}^{10}(1 - \cos(\omega x_i))$. The global minimum is located at $x^* = \mathbf{0}$. We initialize the prior $\mathcal{L}_0 = \mathcal{N}(x_0, \frac{2}{\mu_0} I_n)$ with x_0 far enough that the optimizer must traverse multiple local optima to succeed. Following the setting in Appendix C assume μ_d is known to ALDRIFT as a lower bound on curvature at the optimum solution.

Algorithms. We evaluate both ALDRIFT and TOPIFT using a shared sample budget.

[4] The proof easily extends to $h = c \cdot \sigma$ for constant $c \geq 1$.

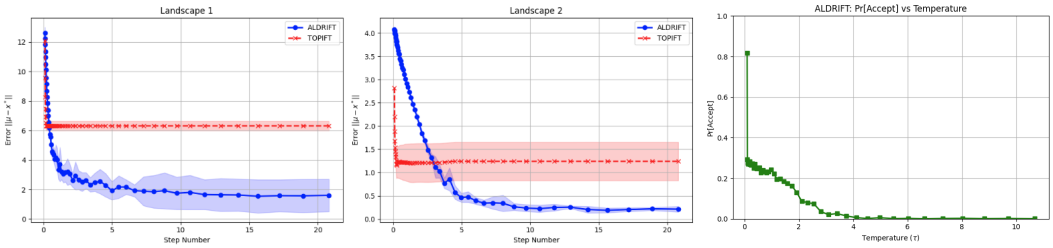


Figure 1: Optimization error trajectories ($||\mu - x^*||$) versus iteration for ALDRIFT and TOPIFT. **(Left)** Parameters: $\mu_d = 0.5, A = 2.0, \omega = 3.0, x_0 = \mathbf{6}, \mu_0 = 0.1$. **(Center)** Parameters: $\mu_d = 1.0, A = 0.5, \omega = 4.0, x_0 = \mathbf{1.35}, \mu_0 = 2.0$. **(Right)** $\Pr[\text{Accept}]$ of a sample versus the temperature τ in ALDRIFT.

- **ALDRIFT (Variance Inflated):** We employ the geometric annealing schedule (as described in Appendix C), updating the temperature as $\tau_{k+1} = (1 + 1/n)\tau_k$ from an initial $\tau_0 = 0.1$ to $T = 20$. At each step, we use Metropolis–Hastings to draw $m = 200$ samples, with a chain length of $M = 50$. Crucially, the model is updated using the inflated variance $\mathcal{L}_\tau(x) = \mathcal{N}(\hat{\omega}_\tau, \frac{2}{\mu_\tau} I_n)$, where $\mu_\tau = \mu_0 + \tau\mu_d$ and $\hat{\omega}_\tau$ is the sample mean.
- **TOPIFT (Empirical Variance):** The heuristic samples $m \cdot M$ times directly from the current Gaussian model, scores them based on $d(x)$, selects the top m samples, and fits the next Gaussian model using their empirical mean and empirical covariance.

Results. Figure 1 (left and center figures) plots the error of the sample mean of the model $\mathcal{L}_\tau$ as a function of iteration number for two different settings of the parameters, averaged over 10 runs for each algorithm. TOPIFT falls into a sub-optimal local minimum because the elite samples are concentrated in a single valley and the empirical variance shrinks to near-zero. Conversely, ALDRIFT successfully converges close to the global optimum by using an inflated proposal distribution and correcting for it every iteration, empirically validating Theorem 5.

Observe that ALDRIFT converges *close* to the global optimum, but does not reach it exactly. This occurs because we operate in a highly constrained sampling regime where M is much smaller than the theoretical requirement for convergence. Consequently, as the temperature increases and the target distribution sharpens, the probability of an accepting Metropolis–Hastings transition approaches zero (see Fig. 1, Right). (A similar stalling effect is observed in our LLM experiments in Appendix F.) Thus, the primary practical advantage of ALDRIFT lies in its ability to rapidly isolate the global basin of attraction using a limited sample budget. Once there, standard local optimization methods can be applied to precisely locate the exact optimum. This contrasts with TOPIFT, which frequently becomes trapped in suboptimal local minima under comparable sample constraints. Conversely, in the asymptotic regime where sample size is unconstrained, naive exploration strategies such as grid search or TOPIFT become competitive by brute-forcing the objective landscape, analogous to the asymptotic convergence guarantees of the Cross-Entropy method and MRAS [18, 41].

7 Conclusion

Our framework opens exciting avenues for future research. An ideal goal would be to prove a condition akin to coarse learnability for general LLMs, and a particularly compelling direction is to formally characterize the class of coarse learners and determine under what conditions modern LLMs fall into this category. For instance, which architectural features, training regimes, or fine-tuning strategies enable these models to satisfy the coarse learnability assumption? Addressing this question may require synthesizing insights from recent work on transformers and chain-of-thought reasoning through the lens of circuit complexity [30]. Another direction is to explore more powerful optimization oracles, such as those providing gradients of the log-probability, which could enable the use of Langevin MCMC in place of Metropolis–Hastings, potentially enhancing the efficiency and scalability of the framework. Finally, it would be interesting to extend Appendix C to work with weaker assumptions than quadratic bounding, and to extend ALDRIFT to remove Assumption 1. Collectively, these directions suggest a rich landscape for integrating statistical learning theory with combinatorial optimization, pointing toward a principled foundation for adaptive generative models.

Acknowledgments and Disclosure of Funding

We utilized Gemini 3.1 Pro to assist with (1) identifying connections to model-based optimization in Appendix G.2; (2) identifying relevant literature to simplify the proof of Theorem 1; (3) improving comparisons to prior work by finding citations; and (4) generating the code for the experiments in Section 6. We also benefited from discussions with the model, which helped suggest possible approaches to proving the results in Appendices C and D.3. All results, including citations and algebra, were verified by the authors, who take full responsibility for the paper's accuracy and contributions.

Funding and competing interests. The authors have no additional funding or competing interests to disclose.

References

[1] Alekh Agarwal, Sham Kakade, Jason Lee, and Gaurav Mahajan. Optimality and approximation with policy gradient methods in Markov decision processes. In *COLT*, 2020.

[2] Christophe Andrieu, Arnaud Doucet, and Roman Holenstein. Particle markov chain monte carlo methods. *J. Royal Stat. Soc.: Series B (Stat. Meth.)*, 72(3):269–342, 2010.

[3] S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. OUP Oxford, 2013.

[4] James A. Brofos, Marylou Gabrié, Marcus A. Brubaker, and Roy R. Lederman. Adaptation of the independent Metropolis–Hastings sampler with normalizing flow proposals. In *AISTATS*, pages 5949–5986, 2022.

[5] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020.

[6] Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *JMLR*, 12(46):1655–1695, 2011.

[7] O Cappé, A Guillin, J. M Marin, and C. P Robert. Population Monte Carlo. *J. Comp. Graph. Stat.*, 13(4):907–929, 2004.

[8] Jiefeng Chen, Jie Ren, Xinyun Chen, Chengrun Yang, Ruoxi Sun, Jinsung Yoon, and Sercan Arık. SETS: leveraging self-verification and self-correction for improved test-time scaling. *TMLR*, 2025.

[9] DeepSeek-AI et al. Deepseek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv*, 2501.12948, 2025.

[10] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential Monte Carlo samplers. *J. Royal Stat. Soc.: Series B (Stat. Meth.)*, 68(3):411–436, 06 2006.

[11] Abraham D. Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA*, pages 385–394, 2005.

[12] Gemini Team. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv*, 2403.05530, 2024.

[13] Subhashis Ghosal and Aad van der Vaart. *Fundamentals of Nonparametric Bayesian Inference*. Cambridge University Press, 2017.

[14] Jean-Bastien Grill, Michal Valko, and Remi Munos. Black-box optimization of noisy functions with unknown smoothness. In *NIPS*, 2015.

[15] Bruce Hajek. Cooling schedules for optimal annealing. *Math. OR*, 13(2):311–329, 1988.

[16] Matthew Hoffman, Pavel Sountsov, Joshua V. Dillon, Ian Langmore, Dustin Tran, and Srinivas Vasudevan. Neutralizing bad geometry in Hamiltonian Monte Carlo using neural transport. *arXiv*, 1903.03704, 2019.

[17] Richard A Holley and Daniel W Stroock. Logarithmic Sobolev inequalities and stochastic Ising models. *J. Stat. Phys.*, 46(5-6):1159–1194, 1987.

[18] Jiaqiao Hu, Michael C Fu, and Steven I Marcus. A model reference adaptive search method for global optimization. *Oper. Res.*, 55(3):549–568, 2007.

[19] Audrey Huang, Adam Block, Dylan J. Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan T. Ash, and Akshay Krishnamurthy. Self-improvement in language models: The sharpening mechanism. In *ICLR*, 2025.

[20] Audrey Huang, Adam Block, Qinghua Liu, Nan Jiang, Akshay Krishnamurthy, and Dylan J Foster. Is best-of-n the best of them? coverage, scaling, and optimality in inference-time alignment. In *ICML*, 2025.

[21] Mark Jerrum and Alistair Sinclair. The Markov chain Monte Carlo method: An Approach to Approximate Counting and Integration. In Dorit S. Hochbaum, editor, *Approximation Algorithms for NP-Hard Problems*, pages 482–520. PWS Publishing Co., USA, 1996.

[22] Deddy Jobson and Yilin Li. Investigating the potential of using large language models for scheduling. In *AIWare*, pages 170–171, 2024.

[23] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *ICML*, 2002.

[24] Adam Tauman Kalai and Santosh Vempala. Simulated annealing for convex optimization. *MOR*, 31(2):253–266, 2006.

[25] Scott Kirkpatrick, C Daniel Gelatt Jr, and Mario P Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[26] László Lovász and Santosh Vempala. Simulated annealing in convex bodies and an $\mathcal{O}^*(n^4)$ volume algorithm. *JCSS*, 72(2):392–417, 2006.

[27] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *NeurIPS*, 2023.

[28] Hongyuan Mei, Mohit Bansal, and Matthew Walter. Listen, attend, and walk: Neural mapping of navigational instructions to action sequences. In *AAAI*, 2016.

[29] Kerrie L Mengersen and Richard L Tweedie. Rates of convergence of the Hastings and Metropolis algorithms. *Ann. Stat.*, 24(1):101–121, 1996.

[30] William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. In *ICLR*, 2024.

[31] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[32] Thomas Minka. Divergence measures and message passing. Technical Report MSR-TR-2005-173, Microsoft Research, December 2005.

[33] Rémi Munos. Optimistic optimization of a deterministic function without the knowledge of its smoothness. In *NIPS*, pages 783–791, 2011.

[34] Radford M. Neal. Annealed importance sampling, 1998.

[35] OpenAI. Introducing OpenAI o1-preview. https://openai.com/index/introducing-openai-o1-preview/, 2024.

[36] Matthew D. Parno and Youssef M. Marzouk. Transport map accelerated markov chain monte carlo. *SIAM/ASA J. Uncert. Quant.*, 6(2):645–682, 2018.

[37] Jan Peters, Katharina Mülling, and Yasemin Altun. Relative entropy policy search. In *AAAI*, pages 1607–1612, 2010.

[38] Jan Peters and Stefan Schaal. Reinforcement learning by reward-weighted regression for operational space control. In *ICML*, pages 745–750, 2007.

[39] S.A. Piyavskii. An algorithm for finding the absolute extremum of a function. *USSR Comp. Math. Math. Phys.*, 12(4):57–67, 1972.

[40] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI report*, 2018. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[41] Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *MCAP*, 1:127–190, 1999.

[42] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *MOR*, 39(4):1221–1243, 2014.

[43] Clayton Sanford, Bahare Fatemi, Ethan Hall, Anton Tsitsulin, Mehran Kazemi, Jonathan Halcrow, Bryan Perozzi, and Vahab Mirrokni. Understanding transformer reasoning capabilities via graph algorithms. In *NeurIPS*, 2024.

[44] John Schulman, Sergey Levine, Philipp Moritz, Michael Jordan, and Pieter Abbeel. Trust region policy optimization. In *ICML*, 2015.

[45] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proc. IEEE*, 104(1):148–175, 2016.

[46] Bruno O. Shubert. A sequential method seeking the global maximum of a function. *SIAM J. Num. Anal.*, 9(3):379–388, 1972.

[47] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. AI models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024.

[48] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *NIPS*, volume 25, 2012.

[49] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *ICML*, 2010.

[50] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020.

[51] Csaba Szepesvári and Rémi Munos. Finite time bounds for sampling based fitted value iteration. In *ICML*, page 880–887, 2005.

[52] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew Walter, Ashis Banerjee, Seth Teller, and Nicholas Roy. Understanding natural language commands for robotic navigation and mobile manipulation. In *AAAI*, pages 1507–1514, 2011.

[53] Philip Thomas and Emma Brunskill. Data-efficient off-policy policy evaluation for reinforcement learning. In *ICML*, 2016.

[54] Luke Tierney. Markov Chains for Exploring Posterior Distributions. *Ann. Stat.*, 22(4):1701 – 1728, 1994.

[55] L. G. Valiant. A theory of the learnable. *CACM*, 27(11):1134–1142, 1984.

[56] Michal Valko, Alexandra Carpentier, and Rémi Munos. Stochastic simultaneous optimistic optimization. In *ICML*, pages 19–27, 2013.

[57] A. W. van der Vaart. *Asymptotic Statistics*. Cambridge University Press, 1998.

[58] A. W. van der Vaart and J. H. van Zanten. Rates of contraction of posterior distributions based on gaussian process priors. *Ann. Stat.*, 36(3):1435–1463, 2008.

[59] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.

[60] Heng Wang, Shangbin Feng, Tianxing He, Zhaoxuan Tan, Xiaochuang Han, and Yulia Tsvetkov. Can language models solve graph problems in natural language? In *NeurIPS*, 2023.

[61] R. Wong. *Asymptotic Approximations of Integrals*. SIAM, 2001.

[62] Tengyang Xie, Dylan J. Foster, Yu Bai, Nan Jiang, and Sham M. Kakade. The role of coverage in online reinforcement learning. In *ICLR*, 2023.

[63] Tengyang Xie, Dylan J. Foster, Akshay Krishnamurthy, Corby Rosset, Ahmed Awadallah, and Alexander Rakhlin. Exploratory preference optimization: Harnessing implicit Q*-approximation for sample-efficient RLHF. In *ICLR*, 2025.

A Additional Related Work

Bandit Convex Optimization. Our problem involves optimizing an objective $d(s)$ where only function evaluations are available. This is related to bandit convex optimization (BCO) [11], which provides regret bounds for online convex minimization with only noisy function evaluations. Conceptually, if the LLM's probability function $\mathcal{L}(s)$ were log-concave and the algorithmic cost $d(s)$ convex, the target distribution $p_\tau(s) \propto \mathcal{L}(s)e^{-\tau d(s)}$ would be log-concave, making it amenable to BCO techniques. Our framework extends this to the general scenario where the prior $\mathcal{L}(s)$ need not be log-concave, and the function $d(s)$ need not be convex. We therefore need a somewhat weaker learnability condition that is applicable to these situations.

Bayesian Methods. Coarse learnability is conceptually related to posterior-contraction results in Bayesian nonparametrics [13, 58], which characterize how a Bayesian posterior concentrates around the true distribution with increasing data. Unlike those asymptotic consistency results, our assumption is a finite-sample condition ensuring that a learned generative model maintains sufficient mass on high-value regions during iterative optimization. Finally, our work bears superficial similarity to the use of Bayesian posteriors in guiding exploration [42, 49]. However, in those settings the posterior merely quantifies uncertainty and could be replaced by frequentist confidence bounds such as UCB. In contrast, in our framework the generative model is an integral component of the optimization process itself, progressively refining its distribution toward the optimal solution.

B Proof of Theorem 1

Our proof focuses on a particular iteration $\tau > 0$. We consider the inner loops of ALDRIFT, where we generate the set of samples S_τ using Metropolis–Hastings. Let $W_{\tau^-} = \{s \mid p_{\tau^-}(s) \le m \cdot \mathcal{L}_{\tau^-}(s)\}$ be the "typical set" where the proposal covers the previous target, and let $\eta = \Pr_{p_{\tau^-}}[W^c_{\tau^-}]$ be the mass of the atypical set.

Lemma 1. *At step τ, let $\mathcal{E}$ be the event that the Metropolis-Hastings chain in* ALDRIFT *encounters a state $s \in W^c_{\tau^-}$ at any step during the generation of the m samples in S_τ. With probability at least $1 - m^2\eta$, the event $\mathcal{E}$ does not occur. Conditioned on $\neg\mathcal{E}$, the distribution $\hat{p}$ of any sample generated by the chain satisfies:*

$$d_{\mathrm{tv}}(\hat{p}, p_\tau) \le e \cdot \eta + e^{-m/e}. \tag{2}$$

Proof. First, since $p_{\tau^-}(s) > m \cdot \mathcal{L}_{\tau^-}(s)$ for all $s \in W^c_{\tau^-}$, we have

$$\Pr_{s\sim\mathcal{L}_{\tau^-}}[s \in W^c_{\tau^-}] \le \frac{1}{m}\Pr_{p_{\tau^-}}[W^c_{\tau^-}] = \frac{\eta}{m}.$$

The algorithm takes $M = m^2$ steps per sample, for a total of m samples. By a union bound over all these steps, the probability that a state in $W^c_{\tau^-}$ is ever proposed is at most $m^3 \cdot (\eta/m) = m^2\eta$. This establishes the probability bound for event $\mathcal{E}$.

Conditioned on $\neg\mathcal{E}$, the chain operates entirely within W_{τ^-}. Since $p_\tau(s) \propto p_{\tau^-}(s)e^{-d(s)/D}$ and since $d(s) \in [0, D]$ for all s, this means $p_\tau(s) \le e \cdot p_{\tau^-}(s)$. Thus, for any $s \in W_{\tau^-}$, the density ratio with respect to the *new* target is bounded:

$$\frac{p_\tau(s)}{\mathcal{L}_{\tau^-}(s)} \le e \cdot \frac{p_{\tau^-}(s)}{\mathcal{L}_{\tau^-}(s)} \le e \cdot m.$$

The sampling procedure is therefore equivalent to an independent Metropolis–Hastings (IMH) chain restricted to W_{τ^-} with density ratio bounded by em. By the uniform ergodicity of IMH [29, 54], the variation distance to the restricted target $p'_\tau \propto p_\tau|_{W_{\tau^-}}$ after $M = m^2$ steps is:

$$d_{\mathrm{tv}}(\hat{p}, p'_\tau) \le \left(1 - \frac{1}{em}\right)^M \le e^{-m/e}.$$

We finally bound the distance to the true target p_τ. The error introduced by restricting the target to W_{τ^-} is bounded by the mass of the excluded set under p_τ:

$$d_{\mathrm{tv}}(p'_\tau, p_\tau) = p_\tau(W^c_{\tau^-}) \le e \cdot p_{\tau^-}(W^c_{\tau^-}) = e \cdot \eta.$$

By the triangle inequality:

$$d_{\mathrm{tv}}(\hat{p}, p_\tau) \leq d_{\mathrm{tv}}(\hat{p}, p'_\tau) + d_{\mathrm{tv}}(p'_\tau, p_\tau) \leq e^{-m/e} + e \cdot \eta.$$

This completes the proof. □

Using the above lemma, we now complete the proof of Theorem 1.

Proof of Theorem 1. As in the text following Assumption 2, we set $\varepsilon = \exp(-K)$ and $\delta = 1/K$, so that $K = m^\beta$ for constant $\beta > 0$. We now choose m large enough that $m^\beta = \omega(TD)$, and note that $m \geq K$.

We prove by induction that if the algorithm succeeds (does not hit atypical sets), then for all τ, $d_{\mathrm{tv}}(\hat{p}, p_\tau) = O(e^{-m^\alpha})$ for some constant $\alpha \in (0, \beta)$. For the base case, note that $d_{\mathrm{tv}}(\hat{p}, p_0) = 0$ since we begin with the model $\mathcal{L}_0 = \mathcal{L}$. For the inductive step, assume $d_{\mathrm{tv}}(\hat{p}, p_{\tau^-}) \leq \varepsilon_{\mathrm{prev}}$. By Assumption 2, the learned model $\mathcal{L}_{\tau^-}$ has an atypical set mass $\eta \leq O(\varepsilon_{\mathrm{prev}}) + e^{-m^\beta}$. Applying Lemma 1, the error at the next step is:

$$d_{\mathrm{tv}}(\hat{p}, p_\tau) \leq e \cdot \left(O\left(\varepsilon_{\mathrm{prev}}\right) + e^{-m^\beta} \right) + e^{-m/e}.$$

Unrolling this recurrence over $T \cdot D$ steps results in an error growth of ϕ^{TD} for some constant $\phi > 1$. Thus:

$$d_{\mathrm{tv}}(\hat{p}, p_T) = O\left(\phi^{TD} \cdot e^{-m^\beta} \right) = e^{O(TD) - m^\beta}.$$

Assuming $m^\beta = \omega(TD)$, we have $d_{\mathrm{tv}}(\hat{p}, p_T) = O(e^{-m^\alpha})$. This completes the induction.

The algorithm fails at any step τ if (a) learning fails (probability δ) or (b) sampling hits W^c (probability $m^2 \cdot \eta$). The above derivation yields $\eta = O(e^{-m^\alpha})$, so that we have $m^2 \eta = o\left(m^{-\beta}\right)$. Further, $\delta = 1/K = m^{-\beta}$. By a union bound over $T \cdot D$ steps, total failure probability is $O(T \cdot D \cdot m^{-\beta}) = o(1)$ since we assume $m^\beta = \omega(TD)$.

Finally, at given any τ, the total number of samples is $O(m^3) = \mathrm{poly}(T, D)$. This completes the proof of Theorem 1. □

C Sample Complexity of ALDRIFT for Bounded Non-Convex Optimization

Classical zeroth-order global optimization in $\mathbb{R}^n$ often relies on local Markov Chain Monte Carlo (MCMC) methods [24]. While these methods achieve polynomial time complexity on strictly log-concave target distributions, they can require exponential time to mix on non-convex landscapes due to local energy barriers [15, 17]. We prove that Model-Based Optimization (MBO) overcomes this limitation, and achieves logarithmic sample complexity in the precision parameter, hence providing a finite-sample analog to the asymptotic convergence results for MRAS.

By leveraging a modification of ALDRIFT with an inflated proposal distribution $\mathcal{L}_\tau$, we show that it can efficiently (globally) optimize non-convex functions that are bounded by a quadratic envelope. In particular, we show that Assumption 2 holds at all intermediate steps with a bound K that is independent of the temperature parameter τ, guaranteeing sample complexity $\widetilde{O}(\log 1/\varepsilon)$ to achieve error ε in the global optimum, matching the bounds of space partitioning methods [6, 33] for this setting (under some caveats that we mention in Appendix C.4). The $O(\cdot)$ notation hides constants that depend on n, K, and $\log \frac{1}{\delta}$.

Recall that in Section 6, we presented an empirical demonstration of the need for an inflated proposal distribution (with sample correction) in ensuring the convergence of ALDRIFT with limited sample budgets, comparing it to TOPIFT that uses the empirical variance and samples without distribution correction, and gets trapped in local optima.

C.1 Optimization Setting and Algorithm

The domain is $\mathbb{R}^n$. Assume the generative prior is a Gaussian $\mathcal{L}_0(x) \propto \exp(-U(x))$, where $U(x) = \frac{\mu_0}{2} \|x - x_0\|^2$ for some $\mu_0 > 0$.

Assume the objective $d(x)$ has a unique global minimum at x^*. We assume $d(x)$ is bounded by a quadratic envelope around its true optimum: there exist constants $\mu_d, L_d > 0$ such that for all $x \in \mathbb{R}^n$,

$$\frac{\mu_d}{2}\|x - x^*\|^2 \leq d(x) - d(x^*) \leq \frac{L_d}{2}\|x - x^*\|^2.$$

Note that $d(x)$ is not assumed to be convex and may contain arbitrary local non-convexities within these bounds. This model is well-studied in non-convex optimization; see for instance [6, 33].

We now run ALDRIFT for N steps as described in Appendix C.3. There are changes needed to the algorithm to handle the lack of an upper bound on $d(x)$, and to make the sampling process exact, that we detail there. We also detail the parameter settings there, and re-do the proof of sample complexity.

When the temperature parameter is τ, the target distribution is $p_\tau(x) \propto \exp(-f_\tau(x))$, where $f_\tau(x) = U(x) + \tau d(x)$. The algorithm draws m samples from p_τ (where the parameter m will be chosen later), computes the empirical mean $\hat{\omega}_\tau$, and fits the proposal distribution (or generative model) $\mathcal{L}_\tau(x) = \mathcal{N}(\hat{\omega}_\tau, \frac{2}{\mu_\tau} I_n)$, where $\mu_\tau = \mu_0 + \tau\mu_d$. We assume the sampling is exactly from p_τ, and we justify this in Appendix C.3. It is also crucial for our proof of Theorem 5 below that the distribution has larger variance than desired and is set based on the convex envelope parameters μ_d, L_d, hence deviating from standard MBO that uses the sample variance directly.

C.2 Verifying Coarse Learnability

We now show that Assumption 2 holds at all intermediate steps of the algorithm, for the proposal distribution $\mathcal{L}_\tau(x) = \mathcal{N}(\hat{\omega}_\tau, \frac{2}{\mu_\tau} I_n)$. Note that in this setting, Assumption 2 is not an assumption; we show that it holds because of our choice of the model $\mathcal{L}_\tau(x)$.

To simplify presentation, we define the following global geometric constants:

- The proxy condition number: $\kappa_{\max} = \max\left(2, \frac{L_d}{\mu_d}\right)$.
- The initial shift penalty: $E_{\text{shift}} = \frac{\mu_0}{2}\|x_0 - x^*\|^2$.
- The inflation factor: $B = \kappa_{\max}^{n/2} \exp(E_{\text{shift}})$.

We will further assume in the rest of the section that $d(x^*) = 0$, so that $d(x) \geq 0$ for all x. This is w.l.o.g., since the normalized density p_τ remains unchanged if we replace $d(x)$ with $\tilde{d}(x) = d(x) - d(x^*)$, so that the algorithm's execution remains identical.

Theorem 5 (Coarse Learnability of p_τ)**.** *Given the setting and algorithm defined above, if the per-step sample size satisfies $m \geq C\left(\frac{n}{2}\ln\kappa_{\max} + E_{\text{shift}}\right)\left(n + \ln\frac{N}{\delta}\right)$ for constant $C > 0$, then for any temperature τ, the model $\mathcal{L}_\tau$ globally satisfies the following condition[5] with probability at least $1 - \delta/(2N)$:*

$$\sup_{x \in \mathbb{R}^n} \frac{p_\tau(x)}{\mathcal{L}_\tau(x)} \leq 2^{n/2} \cdot B^5 \cdot \exp(1 + 2n) := K.$$

Remarks. Theorem 5 can be viewed as a special case of the Coarse Learnability assumption (Assumption 2). Because Theorem 5 establishes a global supremum bound on the density ratio, we have $\Pr[\text{ratio} > K] = 0$. Furthermore, because we utilize Rejection Sampling (Appendix C.3), the empirical samples are drawn perfectly from the target, meaning $d_{\text{tv}}(\hat{p}, p_\tau) = 0$. Consequently, the inequality in Assumption 2 is satisfied for any $\varepsilon > 0$. Further, the sample complexity is $m = \text{poly}(\ln K, \ln 1/\delta)$, which is stronger than the bound in Assumption 2, assuming K is a large enough function of the problem parameters (but independent of T). Theorem 5 suffices for the specialized analysis of ALDRIFT that we perform in Appendix C.3.

Note that K (and consequently the algorithm's sample complexity) does not grow with τ. This non-asymptotically captures the limiting behavior predicted by Laplace's method [61] and the Bernstein-von Mises theorem [57] in Bayesian inference: as the τ-scaled regularizer overwhelms the

[5] This condition is slightly different from how Assumption 2 is phrased, but suffices for the sample complexity proof in Appendix C.3.

prior, the prior collapses into a constant spatial penalty evaluated at the optimum. However, while those classical theorems are valid only as $\tau \to \infty$, our quadratic envelope technique provides an upper bound for any finite $\tau \geq 0$.

The rest of this subsection is devoted to proving this theorem.

Upper Bounding p_τ. We now show a sequence of lemmas that will bound the density ratio. We start with p_τ, and bound it via standard methods.

Lemma 2 (Target Envelope)**.** *The target distribution $p_\tau(x)$ is globally upper-bounded by a scaled Gaussian envelope:*

$$p_\tau(x) \leq B \cdot \left(\frac{\mu_\tau}{2\pi}\right)^{n/2} \exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2\right)$$

where $\bar{x}_\tau = \frac{\mu_0 x_0 + \tau\mu_d x^*}{\mu_\tau}$.

Proof. Let the target probability distribution be defined as $p_\tau(x) = \frac{1}{Z_\tau}\exp(-f_\tau(x))$, where $f_\tau(x) = \frac{\mu_0}{2}\|x - x_0\|^2 + \tau d(x)$ and $Z_\tau = \int_{\mathbb{R}^n}\exp(-f_\tau(x))dx$ is the normalizing constant.

By assumption, $d(x)$ is bounded by quadratics: $\frac{\mu_d}{2}\|x - x^*\|^2 \leq d(x) \leq \frac{L_d}{2}\|x - x^*\|^2$. We can therefore sandwich $f_\tau(x)$ between a lower bounding quadratic $f_{\tau,\text{low}}(x)$ and an upper bounding quadratic $f_{\tau,\text{up}}(x)$:

$$\underbrace{\frac{\mu_0}{2}\|x - x_0\|^2 + \frac{\tau\mu_d}{2}\|x - x^*\|^2}_{f_{\tau,\text{low}}(x)} \leq f_\tau(x) \leq \underbrace{\frac{\mu_0}{2}\|x - x_0\|^2 + \frac{\tau L_d}{2}\|x - x^*\|^2}_{f_{\tau,\text{up}}(x)}.$$

We first simplify $f_{\tau,\text{low}}(x)$ as follows:

$$f_{\tau,\text{low}}(x) = \frac{1}{2}(\mu_0 + \tau\mu_d)\|x\|^2 - (\mu_0 x_0 + \tau\mu_d x^*)^T x + \frac{1}{2}(\mu_0\|x_0\|^2 + \tau\mu_d\|x^*\|^2).$$

To consolidate this into a single quadratic, we define $\mu_\tau = \mu_0 + \tau\mu_d$ and the weighted center $\bar{x}_\tau = \frac{\mu_0 x_0 + \tau\mu_d x^*}{\mu_\tau}$. Substituting these into the expansion yields:

$$\begin{aligned}
f_{\tau,\text{low}}(x) &= \frac{\mu_\tau}{2}\|x\|^2 - \mu_\tau\bar{x}_\tau^T x + \frac{1}{2}(\mu_0\|x_0\|^2 + \tau\mu_d\|x^*\|^2) \\
&= \frac{\mu_\tau}{2}\left(\|x\|^2 - 2\bar{x}_\tau^T x + \|\bar{x}_\tau\|^2\right) - \frac{\mu_\tau}{2}\|\bar{x}_\tau\|^2 + \frac{1}{2}(\mu_0\|x_0\|^2 + \tau\mu_d\|x^*\|^2) \\
&= \frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2 + C_{\tau,\text{low}},
\end{aligned}$$

where

$$\begin{aligned}
C_{\tau,\text{low}} &= \frac{1}{2}(\mu_0\|x_0\|^2 + \tau\mu_d\|x^*\|^2 - \mu_\tau\|\bar{x}_\tau\|^2) \\
&= \frac{1}{2}\left[\mu_0\|x_0\|^2 + \tau\mu_d\|x^*\|^2 - \frac{1}{\mu_0 + \tau\mu_d}\|\mu_0 x_0 + \tau\mu_d x^*\|^2\right] \\
&= \frac{1}{2(\mu_0 + \tau\mu_d)}\Big[\mu_0^2\|x_0\|^2 + \mu_0\tau\mu_d\|x_0\|^2 + \tau\mu_d\mu_0\|x^*\|^2 + (\tau\mu_d)^2\|x^*\|^2 \\
&\qquad\qquad - \mu_0^2\|x_0\|^2 - 2\mu_0\tau\mu_d\langle x_0, x^*\rangle - (\tau\mu_d)^2\|x^*\|^2\Big] \\
&= \frac{1}{2(\mu_0 + \tau\mu_d)}\Big[\mu_0\tau\mu_d\left(\|x_0\|^2 - 2\langle x_0, x^*\rangle + \|x^*\|^2\right)\Big] \\
&= \frac{1}{2}\frac{\mu_0\tau\mu_d}{\mu_0 + \tau\mu_d}\|x_0 - x^*\|^2.
\end{aligned}$$

Analogously, substituting μ_d with L_d, completing the square for $f_{\tau,\text{up}}(x)$ yields a quadratic with smoothness $L_\tau = \mu_0 + \tau L_d$, centered at $\tilde{x}_\tau = \frac{\mu_0 x_0 + \tau L_d x^*}{\mu_0 + \tau L_d}$:

$$f_{\tau,\text{up}}(x) = \frac{L_\tau}{2}\|x - \tilde{x}_\tau\|^2 + C_{\tau,\text{up}}, \quad \text{where} \quad C_{\tau,\text{up}} = \frac{1}{2}\frac{\mu_0\tau L_d}{\mu_0 + \tau L_d}\|x_0 - x^*\|^2.$$

Observe now that the fractional term in $C_{\tau,\text{up}}$:

$$\frac{\tau L_d}{\mu_0 + \tau L_d} = 1 - \frac{\mu_0}{\mu_0 + \tau L_d} \leq 1.$$

Thus, for all $\tau \geq 0$:

$$C_{\tau,\text{up}} \leq \frac{\mu_0}{2}\|x_0 - x^*\|^2 = E_{\text{shift}}.$$

Since $C_{\tau,\text{low}} \geq 0$, we have:

$$C_{\tau,\text{up}} - C_{\tau,\text{low}} \leq C_{\tau,\text{up}} \leq E_{\text{shift}}.$$

Since $f_\tau(x) \leq f_{\tau,\text{up}}(x)$ for all x, it follows that $\exp(-f_\tau(x)) \geq \exp(-f_{\tau,\text{up}}(x))$. We now lower-bound the normalizer Z_τ as:

$$\begin{aligned} Z_\tau = \int_{\mathbb{R}^n} \exp(-f_\tau(x))dx &\geq \int_{\mathbb{R}^n} \exp(-f_{\tau,\text{up}}(x))dx \\ &= \int_{\mathbb{R}^n} \exp\left(-\frac{L_\tau}{2}\|x - \tilde{x}_\tau\|^2 - C_{\tau,\text{up}}\right) dx \\ &= e^{-C_{\tau,\text{up}}} \int_{\mathbb{R}^n} \exp\left(-\frac{L_\tau}{2}\|x - \tilde{x}_\tau\|^2\right) dx. \end{aligned}$$

The remaining integral is an unnormalized multivariate Gaussian with covariance matrix $\Sigma = \frac{1}{L_\tau} I$. Since $\int \exp(-\frac{1}{2}x^T\Sigma^{-1}x)dx = \sqrt{(2\pi)^n \det(\Sigma)}$, we get:

$$Z_\tau \geq e^{-C_{\tau,\text{up}}} \left(\frac{2\pi}{L_\tau}\right)^{n/2}.$$

We finally upper-bound the numerator of $p_\tau(x)$ as: $f_\tau(x) \geq f_{\tau,\text{low}}(x) \implies \exp(-f_\tau(x)) \leq \exp(-f_{\tau,\text{low}}(x))$. Applying this to the numerator and dividing by our lower bound for Z_τ:

$$\begin{aligned} p_\tau(x) = \frac{\exp(-f_\tau(x))}{Z_\tau} &\leq \frac{\exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2 - C_{\tau,\text{low}}\right)}{e^{-C_{\tau,\text{up}}}\left(\frac{2\pi}{L_\tau}\right)^{n/2}} \\ &= \left(\frac{L_\tau}{2\pi}\right)^{n/2} \exp(C_{\tau,\text{up}} - C_{\tau,\text{low}}) \exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2\right) \\ &= \left(\frac{L_\tau}{\mu_\tau}\right)^{n/2} e^{E_{\text{shift}}} \left(\frac{\mu_\tau}{2\pi}\right)^{n/2} \exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2\right). \end{aligned}$$

Since $L_\tau/\mu_\tau \leq \kappa_{\max}$, we can bundle the leading factors into a single constant $B = \kappa_{\max}^{n/2} e^{E_{\text{shift}}}$, yielding the final bound:

$$p_\tau(x) \leq B \cdot \left(\frac{\mu_\tau}{2\pi}\right)^{n/2} \exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2\right).$$

This completes the proof. □

The previous lemma immediately yields the following corollary:

Corollary 1. *The normalizer Z_τ is bounded as:*

$$Z_\tau^{low} := e^{-C_{\tau,up}} \left(\frac{2\pi}{L_\tau}\right)^{n/2} \leq Z_\tau \leq e^{-C_{\tau,low}} \left(\frac{2\pi}{\mu_\tau}\right)^{n/2} := Z_\tau^{up}.$$

In particular, the ratio between these bounds is:

$$\frac{Z_\tau^{up}}{Z_\tau^{low}} \leq e^{E_{shift}} \cdot \left(\frac{L_\tau}{\mu_\tau}\right)^{n/2} \leq e^{E_{shift}} (\kappa_{\max})^{n/2} = B.$$

Sub-Gaussian Upper Bound. Let $q(x) = \mathcal{N}(\bar{x}_\tau, \frac{1}{\mu_\tau} I_n)$. From Lemma 2, we have: $p_\tau(x) \leq B \cdot q(x)$. We now have the following lemmas using this inequality, which will help us bound the density ratio. We provide the proof of the lemma below for completeness.

Lemma 3. *Let $\tilde{\omega}_\tau = \mathbb{E}_{p_\tau}[x]$ be the true expectation. Then,*

$$\mu_\tau \|\tilde{\omega}_\tau - \bar{x}_\tau\|^2 \leq \left(\sqrt{n} + \sqrt{2 \ln B}\right)^2 \leq 2n + 4 \ln B.$$

Proof. By Jensen's inequality, $\|\tilde{\omega}_\tau - \bar{x}_\tau\|^2 \leq \mathbb{E}_{p_\tau}[\|x - \bar{x}_\tau\|^2]$. Let $z = \sqrt{\mu_\tau}(x - \bar{x}_\tau)$. Under the proposal distribution q, $z \sim \mathcal{N}(0, I_n)$, which we denote by $\phi(z)$. Under the target distribution, z follows some density $p(z)$. Since z and x are linearly related, we have $p(z) \leq B\phi(z)$. We seek to bound $\mathbb{E}_p[\|z\|^2]$. Since $p(z) \leq B\phi(z)$, the Kullback-Leibler divergence is bounded:

$$D_{KL}(p\|\phi) = \mathbb{E}_p\left[\ln \frac{p(z)}{\phi(z)}\right] \leq \mathbb{E}_p[\ln B] = \ln B.$$

By the Donsker-Varadhan representation [3], for any measurable function $f(z)$ and any $\lambda > 0$:

$$\mathbb{E}_p[\lambda f(z)] - \ln \mathbb{E}_\phi[e^{\lambda f(z)}] \leq D_{KL}(p\|\phi) \implies \mathbb{E}_p[f(z)] \leq \frac{1}{\lambda}\left(D_{KL}(p\|\phi) + \ln \mathbb{E}_\phi[e^{\lambda f(z)}]\right).$$

We choose $f(z) = \|z\|^2$. Under $\phi(z)$, $\|z\|^2$ follows a χ^2_n distribution, whose moment generating function is $\mathbb{E}_\phi[e^{\lambda\|z\|^2}] = (1-2\lambda)^{-n/2}$ for $\lambda < 1/2$. Substituting this and our KL bound yields:

$$\mathbb{E}_p[\|z\|^2] \leq \frac{\ln B}{\lambda} - \frac{n}{2\lambda}\ln(1-2\lambda).$$

Using $-\ln(1-x) \leq \frac{x}{1-x}$ for $x \in (0,1)$, we set $x = 2\lambda$ to obtain:

$$\mathbb{E}_p[\|z\|^2] \leq \frac{\ln B}{\lambda} + \frac{n}{1-2\lambda}.$$

Choosing $\lambda = \left(2 + \sqrt{\frac{2n}{\ln B}}\right)^{-1} \in (0, 1/2)$ yields:

$$\mathbb{E}_p[\|z\|^2] \leq \left(\sqrt{n} + \sqrt{2\ln B}\right)^2 \leq 2n + 4\ln B.$$

Replacing z with $\sqrt{\mu_\tau}(x - \bar{x}_\tau)$ concludes the proof. □

Before proceeding, we formally define the *variance proxy* of a sub-Gaussian random variable [59]. A random variable Z with mean μ_Z is sub-Gaussian with variance proxy ν^2 if Z satisfies the tail bound $\Pr(|Z - \mu_Z| \geq t) \leq 2\exp(-t^2/\nu^2)$ for all $t \geq 0$. A random vector $X \in \mathbb{R}^n$ is sub-Gaussian with variance proxy ν^2 if its one-dimensional projections $\langle X - \mu_X, u\rangle$ onto any unit vector $u \in \mathbb{S}^{n-1}$ are sub-Gaussian with variance proxy ν^2. The following lemma is now straightforward (see [59]).

Lemma 4. *$X \sim p_\tau$ is sub-Gaussian with a variance proxy bounded by $\nu^2 = O\left(\frac{1+\ln B}{\mu_\tau}\right)$.*

We will need the following concentration bound; see [59] for a proof.

Lemma 5 (Empirical Mean of Sub-Gaussian Vectors)**.** *Let $X_1, \ldots, X_m$ be independent, identically distributed random vectors in $\mathbb{R}^n$ with mean $\omega = \mathbb{E}[X_1]$. Assume each X_i is sub-Gaussian with variance proxy ν^2. Let $\hat{\omega} = \frac{1}{m}\sum_{i=1}^m X_i$ be the empirical mean. Then, there exists a constant $C > 0$ such that for any $t > 0$, with probability at least $1 - e^{-t}$:*

$$\|\hat{\omega} - \omega\|^2 \leq \frac{C\nu^2}{m}(n + t).$$

Proof of Theorem 5. Using these ingredients, we now prove Theorem 5.

Proof of Theorem 5. We evaluate the density ratio $r_\tau(x) = p_\tau(x)/\mathcal{L}_\tau(x)$. The proposal generative model is $\mathcal{L}_\tau(x) = \mathcal{N}(\hat{\omega}_\tau, \frac{2}{\mu_\tau} I_n)$ with density $(\frac{\mu_\tau}{4\pi})^{n/2} \exp(-\frac{\mu_\tau}{4}\|x - \hat{\omega}_\tau\|^2)$. Using the upper bound from Lemma 2 for the numerator:

$$r_\tau(x) \le \frac{B\left(\frac{\mu_\tau}{2\pi}\right)^{n/2} \exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2\right)}{\left(\frac{\mu_\tau}{4\pi}\right)^{n/2} \exp\left(-\frac{\mu_\tau}{4}\|x - \hat{\omega}_\tau\|^2\right)} = B 2^{n/2} \exp\left(-\frac{\mu_\tau}{2}\|x - \bar{x}_\tau\|^2 + \frac{\mu_\tau}{4}\|x - \hat{\omega}_\tau\|^2\right).$$

We apply triangle inequality: $\|x - \hat{\omega}_\tau\|^2 \le 2\|x - \bar{x}_\tau\|^2 + 2\|\hat{\omega}_\tau - \bar{x}_\tau\|^2$ to the second term in the exponent. The x-dependent terms cancel and we have:

$$r_\tau(x) \le 2^{n/2} \cdot B \cdot \exp\left(\frac{\mu_\tau}{2}\|\hat{\omega}_\tau - \bar{x}_\tau\|^2\right). \tag{3}$$

By the triangle inequality, $\|\hat{\omega}_\tau - \bar{x}_\tau\|^2 \le 2\|\hat{\omega}_\tau - \tilde{\omega}_\tau\|^2 + 2\|\tilde{\omega}_\tau - \bar{x}_\tau\|^2$. The exponent in the above expression is therefore bounded by $\mu_\tau\|\hat{\omega}_\tau - \tilde{\omega}_\tau\|^2 + \mu_\tau\|\tilde{\omega}_\tau - \bar{x}_\tau\|^2$. From Lemma 3, we have $\mu_\tau\|\tilde{\omega}_\tau - \bar{x}_\tau\|^2 \le 2n + 4\ln B$.

To bound $\mu_\tau\|\hat{\omega}_\tau - \tilde{\omega}_\tau\|^2$, note that $\hat{\omega}_\tau$ is the empirical mean of m i.i.d. draws from the target distribution p_τ, while $\tilde{\omega}_\tau$ is the true mean. By Lemma 4, these independent draws from p_τ are sub-Gaussian random vectors with a variance proxy bounded by:

$$\nu^2 = C_{\text{sg}} \frac{1 + \ln B}{\mu_\tau}$$

for some constant $C_{\text{sg}} > 0$. We now apply Lemma 5. Setting the failure probability to $e^{-t} = \frac{\delta}{2N}$, which implies $t = \ln\left(\frac{2N}{\delta}\right)$, we obtain with probability at least $1 - \frac{\delta}{2N}$:

$$\begin{aligned}\|\hat{\omega}_\tau - \tilde{\omega}_\tau\|^2 &\le \frac{C}{m}\nu^2\left(n + \ln\left(\frac{2N}{\delta}\right)\right) \\ &= \frac{C}{m}\left(C_{\text{sg}}\frac{1 + \ln B}{\mu_\tau}\right)\left(n + \ln\left(\frac{2N}{\delta}\right)\right).\end{aligned}$$

so that

$$\mu_\tau\|\hat{\omega}_\tau - \tilde{\omega}_\tau\|^2 \le \frac{C \cdot C_{\text{sg}}}{m}(1 + \ln B)\left(n + \ln\left(\frac{2N}{\delta}\right)\right).$$

To satisfy our requirement that $\mu_\tau\|\hat{\omega}_\tau - \tilde{\omega}_\tau\|^2 \le 1$, we set:

$$m \ge C'(1 + \ln B)\left(n + \ln\left(\frac{2N}{\delta}\right)\right)$$

where $C' = C \cdot C_{\text{sg}}$. Therefore with probability at least $1 - \frac{\delta}{2N}$, we have $\frac{\mu_\tau}{2}\|\hat{\omega}_\tau - \bar{x}_\tau\|^2 \le 1 + 2n + 4\ln B$. Substituting this into Eq. (3) yields the bound on K. □

C.3 Implementing ALDRIFT via Geometric Annealing and Rejection Sampling

Given that we have a simple analytic generative model, we make two changes that both simplify ALDRIFT and lead to an exponentially improved sample complexity. Further note that we did not assume an upper bound on $d(x)$ in this setting, which needs a different approach than Theorem 1.

In the description below, we set the number of generated samples m for each temperature from the statement of Theorem 5.

Geometric Annealing. We use a *geometric schedule* of temperatures τ. In particular, we set $\tau_0 = 0$, $\tau_1 = \frac{\mu_0}{L_d}$, and $\tau_{k+1} = (1 + \gamma) \cdot \tau_k$ for $k \ge 1$, where $\gamma = 1/n$. Since $\tau_N \ge T$, this implies the number of outer iterations of ALDRIFT (temperature settings) is $N = O\left(n \ln \frac{TL_d}{\mu_0}\right)$. Such a geometric schedule not only improves the dependence on T exponentially, but is also robust to $d(x)$ being unbounded.

Rejection Sampling. In the inner loop, to generate m samples from $p_{\tau_{k+1}}$, we use rejection sampling instead of Metropolis–Hastings. Although the normalizing constant $Z_{\tau_{k+1}}$ is unknown, we use $\bar{Z}_\tau := (2\pi/\mu_\tau)^{n/2}$ and define the sub-normalized density

$$\hat{p}_\tau(x) := \frac{\exp(-f_\tau(x))}{\bar{Z}_\tau}.$$

By Corollary 1, $Z_\tau \le \bar{Z}_\tau$. At each sampling step, we generate $x \sim \mathcal{L}_{\tau_k}$ and accept it with probability

$$\Pr[\text{Accept} \mid x] = \frac{1}{\sqrt{e}KB} \frac{\hat{p}_{\tau_{k+1}}(x)}{\mathcal{L}_{\tau_k}(x)}.$$

To generate a sample, we run this process for at most $t_s = 10KB^2 \ln \frac{2Nm}{\delta}$ steps, declaring failure if a sample is not generated. We generate m samples and use them to fit $\mathcal{L}_{\tau_{k+1}}$.

Remark. This implementation does not require knowing the exact normalizing constant Z_τ in order to evaluate the acceptance probability. It only requires a valid upper bound on the envelope constant B. If a loose bound $R \ge \|x_0 - x^*\|^2$ is known, we may replace B throughout by the computable upper bound $\widehat{B} := \kappa_{\max}^{n/2} \exp(\mu_0 R/2)$. Under the normalization $d(x^*) = 0$, the lower quadratic envelope also gives the computable choice $R = 2d(x_0)/\mu_d$.

Focusing just on T and ignoring the problem-dependent parameters of B, n, K, the total sample complexity is therefore $\widetilde{O}\left(\ln T \cdot m\right) = \widetilde{O}\left(\ln T \cdot \ln \frac{N}{\delta}\right) = \widetilde{O}\left(\ln T \cdot \ln \frac{\ln T}{\delta}\right)$, and we show below that the algorithm fails with probability at most δ.

C.3.1 Analysis of the Sampling Step

We first show that the rejection sampling step is feasible, meaning that the accept probability is at most one. Clearly, if this is true, this process will generate m unbiased samples from $p_{\tau_{k+1}}$.

Lemma 6. *Let $\tau' := \tau_{k+1}$ and $\tau = \tau_k$ and let γ be the multiplicative growth rate of the annealing procedure. Then, for all iterations $k \ge 0$, with probability $1 - \frac{\delta}{2N}$, we have:*

$$\frac{p_{\tau'}(x)}{\mathcal{L}_\tau(x)} \le K \cdot B \cdot (1+\gamma)^{n/2}.$$

Consequently, setting the geometric rate to $\gamma = \frac{1}{n}$ yields $\sup_x \frac{p_{\tau'}(x)}{\mathcal{L}_\tau(x)} \le B \cdot K \cdot \sqrt{e}$.

Proof. We decompose the density ratio into two terms:

$$\frac{p_{\tau'}(x)}{\mathcal{L}_\tau(x)} = \frac{p_{\tau'}(x)}{p_\tau(x)} \cdot \frac{p_\tau(x)}{\mathcal{L}_\tau(x)} := r(x) \cdot \frac{p_{\tau'}(x)}{p_\tau(x)}.$$

From Theorem 5, we already know that with probability $1 - \frac{\delta}{2N}$, we have $r(x) \le K$ for all x.

To bound the first term, we expand the target densities:

$$\frac{p_{\tau'}(x)}{p_\tau(x)} = \frac{\frac{1}{Z_{\tau'}} \exp(-f_{\tau'}(x))}{\frac{1}{Z_\tau} \exp(-f_\tau(x))} = \frac{Z_\tau}{Z_{\tau'}} \exp\Big(- (f_{\tau'}(x) - f_\tau(x))\Big).$$

By definition, $f_{\tau'}(x) - f_\tau(x) = (\tau' - \tau) d(x)$. Because $\tau' > \tau$ and $d(x) \ge 0$, the exponential term is upper-bounded by 1. Therefore, the ratio of the densities is bounded as:

$$\frac{p_{\tau'}(x)}{p_\tau(x)} \le \frac{Z_\tau}{Z_{\tau'}}. \tag{4}$$

We now bound the ratio $Z_\tau / Z_{\tau'}$ using the bounds established in Corollary 1. For the numerator, we have

$$Z_\tau \le \exp(-C_{\tau,\text{low}}) \left(\frac{2\pi}{\mu_\tau}\right)^{n/2} \le \left(\frac{2\pi}{\mu_\tau}\right)^{n/2},$$

where the last inequality follows because $C_{\tau,\text{low}} \geq 0$. For the denominator, we have:

$$Z_{\tau'} \geq \exp(-C_{\tau',\text{up}}) \left(\frac{2\pi}{L_{\tau'}}\right)^{n/2} \geq \exp(-E_{\text{shift}}) \left(\frac{2\pi}{L_{\tau'}}\right)^{n/2}.$$

where we recall from Lemma 2 that $C_{\tau',\text{up}} \leq E_{\text{shift}}$. Therefore:

$$\frac{Z_\tau}{Z_{\tau'}} \leq \exp(E_{\text{shift}}) \left(\frac{L_{\tau'}}{\mu_\tau}\right)^{n/2}.$$

We first assume $k \geq 1$. We substitute the definitions $L_{\tau'} = \mu_0 + \tau' L_d$ and $\mu_\tau = \mu_0 + \tau\mu_d$, along with the geometric schedule $\tau' = (1+\gamma)\tau$:

$$\frac{L_{\tau'}}{\mu_\tau} = \frac{\mu_0 + (1+\gamma)\tau L_d}{\mu_0 + \tau\mu_d} = \frac{\mu_0 + \tau L_d}{\mu_0 + \tau\mu_d} + \gamma \frac{\tau L_d}{\mu_0 + \tau\mu_d} \leq \kappa_{\max}(1+\gamma).$$

where the last inequality follows since $\kappa_{\max} \geq \sup_\tau (L_\tau/\mu_\tau)$. Plugging this back into the partition function ratio gives:

$$\frac{Z_\tau}{Z_{\tau'}} \leq \exp(E_{\text{shift}}) \kappa_{\max}^{n/2} (1+\gamma)^{n/2} \leq B(1+\gamma)^{n/2}$$

Substituting this back into Eq. (4) and multiplying by the second term $r(x) \leq K$, we have:

$$\frac{p_{\tau'}(x)}{\mathcal{L}_\tau(x)} \leq K \cdot B \cdot (1+\gamma)^{n/2}.$$

When $k = 0$, note that $\tau = 0$ and $\tau' = \mu_0/L_d$, so that

$$\frac{L_{\tau'}}{\mu_\tau} = \frac{2\mu_0}{\mu_0} = 2 \leq \kappa_{\max}.$$

Repeating the above argument completes the proof. Note that setting $\gamma = \frac{1}{n}$ yields the bound:

$$\sup_x \frac{p_{\tau'}(x)}{\mathcal{L}_\tau(x)} \leq K \cdot B \cdot \left(1 + \frac{1}{n}\right)^{n/2} \leq K \cdot B \cdot \sqrt{e}.$$

This justifies the use of $\sqrt{e}KB$ as the bounding constant in the rejection sampling step. □

The previous lemma implies the accept probability in rejection sampling is at most one. We now bound the sampling steps needed to generate an accepted sample.

Lemma 7. *In any sampling step, the probability of acceptance is at least $\frac{1}{\sqrt{e}B^2K}$. Therefore, the probability that the sampling process ever fails over the entire course of the algorithm is at most δ.*

Proof. First, the acceptance probability is well-defined. Since $\bar{Z}_{\tau_{k+1}} \geq Z_{\tau_{k+1}}$, we have $\hat{p}_{\tau_{k+1}}(x) \leq p_{\tau_{k+1}}(x)$ for all x. By Lemma 6, $p_{\tau_{k+1}}(x)/\mathcal{L}_{\tau_k}(x) \leq \sqrt{e}KB$, and hence $\frac{1}{\sqrt{e}KB} \frac{\hat{p}_{\tau_{k+1}}(x)}{\mathcal{L}_{\tau_k}(x)} \leq 1$.

Conditional on acceptance, the accepted sample has density proportional to $\mathcal{L}_{\tau_k}(x) \cdot \frac{1}{\sqrt{e}KB} \frac{\hat{p}_{\tau_{k+1}}(x)}{\mathcal{L}_{\tau_k}(x)}$, which is proportional to $\exp(-f_{\tau_{k+1}}(x))$. Thus the accepted sample is distributed exactly according to $p_{\tau_{k+1}}$. It remains to lower-bound the acceptance probability. Averaging over $x \sim \mathcal{L}_{\tau_k}$ gives

$$\begin{aligned}\Pr[\text{Accept}] &= \int \mathcal{L}_{\tau_k}(x) \frac{1}{\sqrt{e}KB} \frac{\hat{p}_{\tau_{k+1}}(x)}{\mathcal{L}_{\tau_k}(x)} dx = \frac{1}{\sqrt{e}KB} \int \hat{p}_{\tau_{k+1}}(x) dx \\ &= \frac{1}{\sqrt{e}KB} \frac{Z_{\tau_{k+1}}}{\bar{Z}_{\tau_{k+1}}} \geq \frac{1}{\sqrt{e}KB^2},\end{aligned}$$

where the last inequality uses $\bar{Z}_{\tau_{k+1}}/Z_{\tau_{k+1}} \leq B$. To see this, note that by the lower bound in Corollary 1, $Z_\tau \geq e^{-C_{\tau,\text{up}}} (2\pi/\Lambda_\tau)^{n/2}$, where $\Lambda_\tau := \mu_0 + \tau L_d$. Therefore

$$\frac{\bar{Z}_\tau}{Z_\tau} \leq \frac{(2\pi/\mu_\tau)^{n/2}}{e^{-C_{\tau,\text{up}}} (2\pi/\Lambda_\tau)^{n/2}} = e^{C_{\tau,\text{up}}} \left(\frac{\Lambda_\tau}{\mu_\tau}\right)^{n/2} \leq e^{E_{\text{shift}}} \kappa_{\max}^{n/2} = B,$$

where we used $C_{\tau,\text{up}} \leq E_{\text{shift}}$ and $\Lambda_\tau/\mu_\tau \leq \kappa_{\max}$.

Since the rejection process follows a geometric distribution, the probability a sample is not generated in $t_s = 10KB^2 \ln \frac{2Nm}{\delta}$ steps is at most $\exp(-t_s/(\sqrt{e}KB^2)) \leq \frac{\delta}{2Nm}$. Summing over $N \cdot m$ total samples across all temperatures yields a failure probability of at most $\delta/2$. Note that in Lemma 6, there is a probability $\delta/(2N)$ per temperature that the density ratio itself is not bounded. Taking the union bound over all such bad events yields a total failure probability of at most δ. □

C.3.2 Relating Temperature to the Precision of Global Optimization

We now find the T needed to achieve precision $\mathbb{E}_{p_T}[d(x)] \leq \varepsilon$ for a given ε, hence casting our sample complexity in more standard terms. Recall we assumed $d(x^*) = 0$.

Lemma 8 (Precision Scaling)**.** *To achieve an expected objective error $\mathbb{E}_{p_T}[d(x)] \leq \varepsilon$, it suffices to set the final temperature to $T = O(1/\varepsilon)$. This implies an overall sample complexity of $\widetilde{O}(\log 1/\varepsilon)$.*

Proof. Since $d(x) \leq \frac{L_d}{2}\|x - x^*\|^2$, taking the expectation over the final target distribution $p_T(x)$:

$$\mathbb{E}_{p_T}[d(x)] \leq \frac{L_d}{2}\mathbb{E}_{p_T}[\|x - x^*\|^2] \leq L_d\mathbb{E}_{p_T}[\|x - \bar{x}_T\|^2] + L_d\|\bar{x}_T - x^*\|^2,$$

where we have used the triangle inequality. The second term expands to:

$$\|\bar{x}_T - x^*\|^2 = \left\|\frac{\mu_0 x_0 + T\mu_d x^*}{\mu_0 + T\mu_d} - x^*\right\|^2 = \frac{2\mu_0 E_{\text{shift}}}{(\mu_0 + T\mu_d)^2} = O\left(\frac{1}{T^2}\right).$$

To bound the term $\mathbb{E}_{p_T}[\|x - \bar{x}_T\|^2]$, we use Lemma 2. Let $q_T = \mathcal{N}(\bar{x}_T, \mu_T^{-1} I_n)$, where $\mu_T = \mu_0 + T\mu_d$. Since $p_T(x) \leq Bq_T(x)$ for all x, we have

$$\mathbb{E}_{p_T}[\|x - \bar{x}_T\|^2] \leq B\mathbb{E}_{q_T}[\|x - \bar{x}_T\|^2] = \frac{nB}{\mu_0 + T\mu_d} = O\left(\frac{1}{T}\right).$$

For sufficiently large T, the $O(1/T)$ dominates, so that the error is $O(\varepsilon)$. Recall that the sample complexity is $\widetilde{O}\left(\ln T \cdot \ln \frac{\ln T}{\delta}\right)$, which to the higher order multiplicative term is $\widetilde{O}(\log 1/\varepsilon)$. □

C.4 Comparison to Other Non-convex Optimization Methods

Our sample complexity bound decouples the problem parameters such as dimension n from the precision ε, yielding a $\tilde{O}(\log 1/\varepsilon)$ sample complexity, with constants depending on problem parameters. We compare this guarantee with extant methods for non-convex optimization.

1. Grid Search (Lipschitz Optimization): Standard spatial discretization algorithms require $O((1/\varepsilon)^n)$ function evaluations, so the precision parameter ε gets exponentially modified.

2. Local MCMC and Simulated Annealing: While local Markov chains scale polynomially in dimension on convex bodies, they can take exponential time in non-convex landscapes. The Holley-Stroock perturbation lemma [17] implies escaping a local minimum of depth ΔE at inverse temperature τ requires mixing time $O(\exp(\tau\Delta E))$, and Hajek's theorem [15] implies guaranteeing convergence in local MCMC simulated annealing requires a logarithmic cooling schedule $\tau_t = O(\ln t)$. These results mean that in the worst case, classical simulated annealing may require $O(\exp(1/\varepsilon))$ steps, so again, the precision parameter is exponentially modified.

To mitigate these slow mixing times, classical MCMC and simulated annealing often employ variance-inflated distributions, either as heavier-tailed proposals to guarantee ergodicity [29] or as "warm starts" to bound the density ratio between temperature steps [26]. However, classical methods utilize variance inflation to aid *local* random walks, which can take exponentially many samples for non-convex landscapes. In contrast, ALDRIFT adapts the variance-inflation principle to bound the mass coverage of a *generative proxy*. The proxy dynamically absorbs empirical estimation errors independent of the annealing temperature, allowing the algorithm to achieve logarithmic sample complexity.

3. Deterministic Optimistic Optimization (DOO): Among finite-sample methods, the closest comparators are optimistic partitioning methods such as DOO [6], SOO [33] and their stochastic/noisy variants (e.g. HOO/StoSOO [56] and POO [14]). In contrast, classical Lipschitz branch-and-bound

methods [39, 46] only convergence in different regimes. When the objective is bounded by a quadratic envelope, spatial-partitioning algorithms [6, 33] achieve $\widetilde{O}(\log(1/\varepsilon))$ sample complexity. These methods hierarchically search the space via optimistic estimates of the objective within grid partitions. ALDRIFT achieves this same characteristic rate, but under different trade-offs. Unlike these methods, ALDRIFT requires prior estimates of both the upper and lower bounds on the curvature parameters (μ_d, L_d), and its sample complexity scales with the full ambient dimension n, whereas space-partitioning bounds scale with the near-optimality dimension, which could be smaller. Furthermore, extensions like SOO [33] only require the existence of an upper bounding quadratic without needing to know its parameters, a setting ALDRIFT does not handle. Addressing these limitations in ALDRIFT is an exciting direction for future work.

On the other hand, space-partitioning operations are harder to define for discrete, combinatorial, or semantic generative spaces (e.g., sequences generated by LLMs). The ALDRIFT algorithm matches the sample complexity of space partitioning methods for quadratic-bounded functions, while generalizing as an algorithmic framework to other generative priors and discrete optimization objectives whenever Assumption 2 holds.

D Theoretical Evidence for Coarse Learnability

Complementing the sample complexity proof in Appendix C, we now present theoretical evidence to support the coarse learnability assumption (Assumption 2) for simple generative models. First, to demonstrate robustness in the *agnostic* setting, we prove that a structurally misspecified learner (a single Gaussian) naturally satisfies the coverage condition when approximating a multimodal target (Appendix D.1). Second, to validate consistency with classical MBO theory, we show that standard MLE in the *realizable* setting of MRAS (i.e., exponential families) satisfies the assumption with high probability (Appendix D.2). Finally, we demonstrate that coarse learnability extends to a simple KDE task, provided the kernel is deliberately "over-smoothed" to act as a robust coverage envelope (Appendix D.3).

D.1 Gaussian Mixtures and Robustness to Mis-specification

Assumption 2 particularly relevant in the agnostic or misspecified setting, where the learner cannot model the complex shape of the target but can learn to cover its support. Indeed, when a lower-capacity model (like a restricted neural network or a single Gaussian) approximates a complex, multimodal target, the MLE objective forces the learner to "spread out" and envelope the target's support rather than collapsing to a single mode [32]. In our setting, the generative model learns to act as this relaxed envelope.

We illustrate this with a Gaussian mixture model target and a single Gaussian learner. Let the target $p(x)$ be a mixture of k unit-variance Gaussians:

$$p(x) = \frac{1}{k}\sum_{i=1}^{k}\mathcal{N}(\mu_i, 1).$$

Let the learner $\mathcal{L}$ be restricted to the family of single Gaussians $\mathcal{N}(\theta, \sigma^2)$. This is a mis-specified setting where $d_{\mathrm{TV}}(p, \mathcal{L})$ cannot be made arbitrarily small. However, we show that MLE naturally satisfies Assumption 2.

Theorem 6. *Let the target $p(x) = \frac{1}{k}\sum_{i=1}^{k}\mathcal{N}(\mu_i, 1)$ be a mixture of k unit-variance Gaussians with means $\mu_i \in [-\frac{\Delta}{2}, \frac{\Delta}{2}]$, where $\Delta \geq 2$. Assume the modes are sufficiently separated such that the population variance satisfies $\sigma^{*2} \geq 2$. Let $\mathcal{L}_{\mathrm{MLE}}$ be the single Gaussian model $\mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$ learned via MLE on m i.i.d. samples from p.*

If $m = \Omega(\Delta^4 \ln(1/\delta))$, then with probability at least $1-\delta$, the learned model satisfies the global coverage condition:

$$\sup_{x\in\mathbb{R}} \frac{p(x)}{\mathcal{L}_{\mathrm{MLE}}(x)} = O(\Delta \cdot e^k) := K.$$

Thus, for a fixed number of modes k, the model family satisfies Assumption 2 with m scaling polynomially in $K = O(\Delta)$ and $\ln(1/\delta)$.

Proof. Let μ^* and σ^{*2} denote the true mean and variance of the target mixture p.

The mean of the mixture is $\mu^* = \frac{1}{k}\sum \mu_i$. The variance of the mixture is given by the law of total variance:

$$\sigma^{*2} = 1 + \frac{1}{k}\sum_{j=1}^{k}(\mu_j - \mu^*)^2.$$

From this identity, we observe that the squared deviation of any single component mean is bounded by the total variance scaled by k:

$$(\mu_i - \mu^*)^2 \leq \sum_{j=1}^{k}(\mu_j - \mu^*)^2 = k(\sigma^{*2} - 1). \tag{5}$$

By the premise, $\sigma^{*2} \geq 2$. Since the means are bounded in width Δ, we also have $\sigma^{*2} \leq 1 + \Delta^2$.

The MLE parameters are the sample mean $\hat{\mu}$ and sample variance $\hat{\sigma}^2$. The target distribution p is a mixture of Gaussians with bounded parameter support. Since the component means are bounded by $\Delta/2$ and the component variances are fixed at 1, the random variable $X \sim p$ is sub-Gaussian with norm $\|X\|_{\psi_2} = O(\Delta)$. First, by Hoeffding's inequality [59], for any $t > 0$ (where c is a constant).

$$\Pr\left(|\hat{\mu} - \mu^*| \geq t\right) \leq 2\exp\left(-cm\frac{t^2}{\Delta^2}\right).$$

Next, the variable X^2 is sub-exponential with $\|X^2\|_{\psi_1} = O(\Delta^2)$, so by Bernstein's inequality [59]:

$$\Pr\left(|\hat{\sigma}^2 - \sigma^{*2}| \geq t\right) \leq 2\exp\left(-cm\min\left(\frac{t^2}{\Delta^4}, \frac{t}{\Delta^2}\right)\right).$$

We choose a sampling error tolerance η as a sufficiently small constant. To ensure $|\hat{\mu} - \mu^*| \leq \eta$ and $|\hat{\sigma}^2 - \sigma^{*2}| \leq \eta$ with probability $1 - \delta$, we need $m = \Omega(\Delta^4 \ln(1/\delta))$. Conditioned on this high-probability event, and using $\sigma^{*2} \geq 2$, we have $\hat{\sigma}^2 > 1$.

The coverage ratio is $r(x) = \frac{p(x)}{\mathcal{L}(x)} = \frac{1}{k}\sum_{i=1}^{k}\frac{\mathcal{N}(x;\mu_i,1)}{\mathcal{N}(x;\hat{\mu},\hat{\sigma}^2)}$. It suffices to bound an arbitrary term $T_i(x) = \frac{\mathcal{N}(x;\mu_i,1)}{\mathcal{N}(x;\hat{\mu},\hat{\sigma}^2)}$:

$$T_i(x) = \hat{\sigma}\exp\left(-\frac{(x-\mu_i)^2}{2} + \frac{(x-\hat{\mu})^2}{2\hat{\sigma}^2}\right).$$

Let $E_i(x)$ be the exponent. Differentiating with respect to x:

$$E_i'(x) = -(x - \mu_i) + \frac{(x - \hat{\mu})}{\hat{\sigma}^2}.$$

Setting $E_i'(x) = 0$ yields the unique maximum at:

$$x^* = \frac{\hat{\sigma}^2\mu_i - \hat{\mu}}{\hat{\sigma}^2 - 1}.$$

Substituting this back into $E_i(x)$, the exponent at the maximum simplifies to:

$$E_i(x^*) = \frac{(\mu_i - \hat{\mu})^2}{2(\hat{\sigma}^2 - 1)}.$$

We now apply the finite sample bounds derived above. First, consider the numerator $(\mu_i - \hat{\mu})^2$. Since $|\hat{\mu} - \mu^*| \leq \eta$, we have

$$(\mu_i - \hat{\mu})^2 \leq (\mu_i - \mu^*)^2 + 2\eta|\mu_i - \mu^*| + \eta^2.$$

Using the population variance bound from Eq. (5), let $V = \sigma^{*2} - 1$. Then $(\mu_i - \mu^*)^2 \leq kV$, which implies $|\mu_i - \mu^*| \leq \sqrt{kV}$. Thus:

$$(\mu_i - \hat{\mu})^2 \leq kV + 2\eta\sqrt{kV} + \eta^2.$$

Next, consider the denominator $2(\hat{\sigma}^2 - 1)$. Using $|\hat{\sigma}^2 - \sigma^{*2}| \leq \eta$:

$$\hat{\sigma}^2 - 1 \geq \sigma^{*2} - 1 - \eta = V - \eta.$$

Since we assumed $\sigma^{*2} \geq 2$, we have $V \geq 1$. Since $\eta \leq 0.5$, we have $V - \eta \geq V(1-\eta) \geq 0.5V$. For small enough constant η, the exponent $E_i(x^*)$ is bounded by:

$$E_i(x^*) \leq \frac{kV + 2\eta\sqrt{kV} + \eta^2}{2(V-\eta)} = \frac{k}{2}\left(\frac{1 + \frac{2\eta}{\sqrt{kV}} + \frac{\eta^2}{kV}}{1 - \frac{\eta}{V}}\right) \leq k.$$

Consequently, the maximum value of the component ratio is:

$$\sup_x T_i(x) = \hat{\sigma}\exp(E_i(x^*)) \leq \hat{\sigma} \cdot e^k.$$

Since $\hat{\sigma}^2 \leq \sigma^{*2} + \eta \leq 2 + \Delta^2$, we have $\hat{\sigma} \leq O(\Delta)$.

Summing over the k components (each with weight $1/k$):

$$\sup_x \frac{p(x)}{\mathcal{L}(x)} \leq \frac{1}{k}\sum_{i=1}^{k} \sup_x T_i(x) \leq \frac{1}{k}\sum_{i=1}^{k} O(\Delta \cdot e^k) = O(\Delta \cdot e^k).$$

We can now choose any $m \geq \text{poly}\left(\ln\frac{1}{\delta}, \Delta, e^k\right)$ to satisfy Assumption 2. Note that the coverage condition holds globally. □

D.2 Exponential Families and Realizable Setting

We now present theoretical justification for coarse learnability for simple distributions. We consider the class of exponential families with a single parameter and show that under benign assumptions on the family, the MLE estimation problem is coarsely learnable. This result extends to some multi-parameter settings, and we present a sketch at the end omitting the details. Note that single-parameter exponential families capture most common distributions such as Gaussian, Exponential, Poisson, Binomial, etc. This result shows that coarse learnability is a natural property of standard statistical estimation in well-behaved settings.

It is important to contextualize the result below within the model reference adaptive search (MRAS) literature. As discussed in Appendix G.2, standard MRAS utilizes natural exponential families for the parametric model [18]. While true target distributions are generally complex, analyzing the idealized "realizable" setting, where the target falls within the model family, serves as a theoretical baseline. The analysis below confirms that in this realizable setting, the finite-sample MLE inherently satisfies our coarse learnability assumption. This indicates that our statistical condition is well-grounded, holding for standard parametric families in an idealized realizable setting. Crucially, however, as discussed in Appendix D.1, our framework extends guarantees beyond this idealized case to the more general agnostic setting where the target is complex and the model is misspecified, by utilizing the Metropolis–Hastings correction.

Single-Parameter Setting. Let $p(x \mid \eta) = h(x)\cdot e^{\eta x - A(\eta)}$ be a single-parameter exponential family and let X denote a random variable following this distribution. Let $\mu(\eta) = \mathbb{E}_{X\sim p(x|\eta)}[X] = A'(\eta)$. We suppose the exponential family satisfies the following simple assumption.

Assumption 3 (Regularity)**.** *We assume the density $p(x \mid \eta)$ is sub-exponential.*[6] *Further, the true parameter η_0 lies in a constant-sized closed interval Ω_0 such that for all $\eta \in \Omega_0$, we have that $A(\eta)$ is analytic and $0 < v_{\min} \leq A''(\eta) \leq v_{\max} < \infty$ for some positive constants $v_{\min}, v_{\max}$.*

The above assumption holds for common distributions such as Gaussian, Bernoulli, Exponential, and Poisson under mild conditions on their parameters. These densities are clearly sub-exponential. Furthermore, we have the following:

Gaussian (with known variance). We have $p(x \mid \mu) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-(x-\mu)^2/(2\sigma^2)}$. Therefore, $\eta = \mu/\sigma^2$ and $A(\eta) = \eta^2\sigma^2/2$, so that $A''(\eta) = \sigma^2$, which is a constant.

Bernoulli. We have $p(x \mid \mu) = \mu^x(1-\mu)^{1-x}$ for $x \in \{0,1\}$. We have $\eta = \ln(\mu/(1-\mu))$, and $A(\eta) = \ln(1+e^\eta) = -\ln(1-\mu)$. We have $A''(\eta) = e^\eta/(1+e^\eta)^2 = \mu(1-\mu)$, which satisfies the above assumption when $\mu \in [\delta, 1-\delta]$ for constant $\delta > 0$.

[6] A sub-exponential random variable satisfies $\Pr[|X| \geq K] \leq 2e^{-C\cdot K}$ for all $K \geq 0$ and some constant $C > 0$.

Exponential. We have $p(x \mid \lambda) = \lambda e^{-\lambda x}$ for $x \geq 0$. Further, we have $\eta = -\lambda$ and $A(\eta) = -\ln(-\eta)$, so that $A''(\eta) = 1/\eta^2$, which satisfies the above assumption when $\lambda > 0$ is a constant.

Poisson. We have $p(x \mid \lambda) = e^{-\lambda}\lambda^x/x!$ for $x = 0, 1, 2, \ldots$. We have $\eta = \ln \lambda$, $A(\eta) = e^\eta$, and $A''(\eta) = e^\eta = \lambda$, which satisfies the assumption when $\lambda > 0$ is a constant.

The derivation of coarse learnability under Assumption 3 uses standard properties of sub-exponential distributions; see, e.g., [59] for details. Since we assumed the exponential family is sub-exponential, this means there exist positive constants $C_1, C_2 > 0$ such that for $X \sim p(x \mid \eta_0)$:

$$\Pr\left(|X - \mu(\eta_0)| > \theta\right) \leq C_1 e^{-C_2 \cdot \theta}. \tag{6}$$

This also means [59] that there are constants ν^2, α such that for random variable X following $p(x \mid \eta)$, we have:

$$\mathbb{E}[e^{\lambda X}] \leq e^{\lambda^2 \nu^2/2} \qquad \forall |\lambda| \leq \frac{1}{\alpha}.$$

Let $Z = X - \mu(\eta)$, so that $\mathbb{E}[Z] = 0$. Clearly, Z is also sub-exponential, and satisfies the above equation for some ν^2, α. Suppose we draw m samples from $p(x \mid \eta)$ and let $\hat{X}$ be their average:

$$\begin{aligned}\Pr\left(\hat{X} - \mu(\eta) > \varepsilon\right) &= \Pr\left(\sum_{i=1}^m Z_i > m \cdot \varepsilon\right) = \Pr\left(\prod_{i=1}^m e^{\lambda Z_i} > e^{m \cdot \lambda \cdot \varepsilon}\right) \\ &\leq \frac{\left(\mathbb{E}[e^{\lambda Z}]\right)^m}{e^{m \cdot \lambda \cdot \varepsilon}} \leq e^{m\left(\lambda^2 \cdot \nu^2/2 - \lambda \cdot \varepsilon\right)}.\end{aligned}$$

Choosing $\lambda = \frac{\varepsilon}{\nu^2} < \frac{1}{\alpha}$ (assuming $\varepsilon = o(1)$), we have

$$\Pr\left(\left|\hat{X} - \mu(\eta)\right| > \varepsilon\right) \leq 2e^{-\frac{m \cdot \varepsilon^2}{2\nu^2}}.$$

Choosing $\varepsilon = m^{-1/3}$, this implies:

$$\Pr\left(\left|\hat{X} - \mu(\eta)\right| > m^{-1/3}\right) \leq 2\exp\left(-\frac{m^{1/3}}{2\nu^2}\right). \tag{7}$$

We are now ready to show the following theorem, which we show implies Assumption 2.

Theorem 7. *Suppose the exponential family satisfies Assumption 3. Let $\hat{\eta}_m$ be the MLE of η_0 based on m samples, satisfying $A'(\hat{\eta}_m) = \bar{X}_m$, where $\bar{X}_m$ is the average of m samples. Let $\mathcal{L}(x) \equiv p(x \mid \hat{\eta}_m)$. For any small constant $\gamma > 0$, define*

$$W_m = \left\{x : |\ln p(x \mid \eta_0) - \ln p(x \mid \hat{\eta}_m)| \leq m^{-1/3+\gamma}\right\}.$$

Then, for any $m \geq \text{poly}\left(\frac{v_{\max}}{v_{\min}}, \frac{1}{\nu}\right)$ and a sufficiently small constant $\gamma > 0$, with probability $1 - 2e^{-m^\gamma}$, the parameter $\hat{\eta}_m$ satisfies $\Pr[X \notin W_m] \leq e^{-m^\gamma}$.

Proof. Let $\Delta\eta = \hat{\eta}_m - \eta_0$. The log-density ratio is

$$\mathcal{R}(x) = \ln p(x \mid \eta_0) - \ln p(x \mid \hat{\eta}_m) = (\eta_0 - \hat{\eta}_m) \cdot x + (A(\hat{\eta}_m) - A(\eta_0)).$$

Using a Taylor expansion for $A(\hat{\eta}_m) = A(\eta_0 + \Delta\eta)$ around η_0 up to the second term, for some ξ between η_0 and $\hat{\eta}_m$, we have

$$A(\hat{\eta}_m) - A(\eta_0) = A'(\eta_0)\Delta\eta + \frac{1}{2}A''(\xi)(\Delta\eta)^2.$$

Substituting this into $\mathcal{R}(x)$:

$$\mathcal{R}(x) = \Delta\eta(A'(\eta_0) - x) + \frac{1}{2}A''(\xi)(\Delta\eta)^2.$$

Let $\mu(\eta_0) = A'(\eta_0)$ be the true mean. By Eq. (7), the event $\mathcal{E}_{\text{param}} := \{\left|\bar{X}_m - \mu(\eta_0)\right| \leq m^{-1/3}\}$ occurs with probability at least $1 - \exp(-m^{1/3}/\nu^2)$. The MLE satisfies $A'(\hat{\eta}_m) = \bar{X}_m = \mu(\hat{\eta}_m)$.

On $\mathcal{E}_{\text{param}}$, we therefore have $|A'(\hat{\eta}_m) - A'(\eta_0)| \le m^{-1/3}$. By the Mean Value Theorem, $A'(\hat{\eta}_m) - A'(\eta_0) = A''(\eta^*)(\hat{\eta}_m - \eta_0)$ for some η^* between η_0 and $\hat{\eta}_m$. By Assumption 3, $A''(\eta^*) \ge v_{\min} > 0$. Thus, on $\mathcal{E}_{\text{param}}$, we have

$$|\Delta\eta| = |\hat{\eta}_m - \eta_0| \le \frac{1}{v_{\min}} |A'(\hat{\eta}_m) - A'(\eta_0)| \le \frac{1}{v_{\min}} m^{-1/3}.$$

This implies $(\Delta\eta)^2 \le \frac{m^{-2/3}}{v_{\min}^2}$. Also, on $\mathcal{E}_{\text{param}}$, $A''(\xi) \le v_{\max}$ by Assumption 3. This implies the first term $\Delta\eta(A'(\eta_0) - x)$ dominates in the equation for $\mathcal{R}(x)$ when $m \ge \text{poly}\left(\frac{v_{\max}}{v_{\min}}\right)$. Thus, for $|\mathcal{R}(x)| \le m^{-1/3+\gamma}$, we require:

$$|\Delta\eta(A'(\eta_0) - x)| \le \frac{1}{2} m^{-1/3+\gamma}.$$

Since $|\Delta\eta| \le \frac{m^{-1/3}}{v_{\min}}$, this requires $|A'(\eta_0) - x| \le \frac{1}{2} v_{\min} m^{\gamma} := K_m$. The set W is then defined by x such that $|x - \mu(\eta_0)| \le K_m$. By Eq. (6), we have $\Pr[x \notin W] = O\left(e^{-O(K_m)}\right)$. Choosing $\gamma > 0$ an appropriately small constant, this completes the proof. □

Note that the failure probabilities in the above theorem are monotonically decreasing in m. Therefore, if we are given $K > 1, \varepsilon > 0, \delta > 0$ as in Assumption 2, then we can simply choose a large enough $m \ge \text{poly}\left(\frac{v_{\max}}{v_{\min}}, \frac{1}{\nu}, \ln \frac{1}{\varepsilon\delta}\right)$ that ensures $e^{-m^{\beta}} \le \min(\delta, \varepsilon)$. Further, note that $m^{-1/3+\gamma} < \ln K$. This ensures that with probability $1 - \delta$, we have $\Pr[p(x|\eta_0) \ge K \cdot p(x|\hat{\eta}_m)] \le \varepsilon$, hence satisfying Assumption 2.

Multi-Parameter Setting. Let $p(\mathbf{x} \mid \boldsymbol{\eta}) = h(\mathbf{x}) \exp(\boldsymbol{\eta}^T \mathbf{x} - A(\boldsymbol{\eta}))$ be a k-parameter exponential family, where $\boldsymbol{\eta} \in \Re^k$ and $\mathbf{x} \in \Re^k$, and k being a constant. This for instance, captures multi-dimensional Gaussian distributions with a known covariance matrix, or multinomial distributions. Assuming the resulting density is sub-exponential in the ℓ_2-norm, it can be shown that such a distribution satisfies Eq. (6) and Eq. (7) with the absolute value replaced by the ℓ_2-norm, and the constants depending on k. In particular, the ℓ_2-norm of the deviation from the mean is a sub-exponential random variable. By re-working the same proof as that of Theorem 7, this implies an analog of Theorem 7 to the multi-dimensional setting, under suitable regularity assumptions on $A(\boldsymbol{\eta})$. The details are easy to fill in, and omitted.

D.3 Over-smoothed Non-Parametric Estimation

The previous sections demonstrated that coarse learnability arises naturally in parametric settings due to the mass-covering properties of Maximum Likelihood Estimation. We now show that this principle extends to a simple non-parametric KDE task, provided the estimator is deliberately "over-smoothed."

In classical density estimation, the bandwidth h of a kernel is typically shrunk as the sample size $m \to \infty$ to recover the exact target density. However, in our optimization framework, a shrinking bandwidth creates light tails that exponentially under-cover the target's support, violating Assumption 2. Conversely, if we fix a kernel whose bandwidth dominates the target's tail decay, the KDE forms a robust coverage envelope. We formalize this in d-dimensions.

Theorem 8. *Let the target distribution $p(x) = \mathcal{N}(0, \sigma^2 I_d)$ be a d-dimensional isotropic Gaussian. Let $\mathcal{L}_{KDE}(x) = \frac{1}{m}\sum_{i=1}^{m} K_h(x - X_i)$ be the density learned via KDE using a Gaussian kernel $K_h(z) = (2\pi h^2)^{-d/2} \exp(-\|z\|^2/2h^2)$ on m i.i.d. samples $X_1, \ldots, X_m$ from p.*

Assume we set the bandwidth to match[7] the target's tail decay, $h = \sigma$. There exists a universal constant $C > 0$ such that if the sample size satisfies $m \ge C(d + \ln(1/\delta))$, then with probability at least $1 - \delta$ over the samples, the learned model globally satisfies the coarse learnability condition:

$$\sup_{x \in \mathbb{R}^d} \frac{p(x)}{\mathcal{L}_{KDE}(x)} \le 8e^d := K.$$

This implies $m = O\left(\ln \frac{K}{\delta}\right)$, showing Assumption 2.

[7] The proof easily extends to $h = c \cdot \sigma$ for constant $c \ge 1$.

Proof. We first show that for any query point $x \in \mathbb{R}^d$, a constant fraction of the empirical samples simultaneously resides near the origin and directionally points toward x.

Let $\mathcal{B} = \{y \in \mathbb{R}^d : \|y\|^2 \leq 2\sigma^2 d\}$ be the closed ball of radius $\sigma\sqrt{2d}$ centered at the origin. For $X \sim \mathcal{N}(0, \sigma^2 I_d)$, the expected squared norm is $\mathbb{E}[\|X\|^2] = \sigma^2 d$. By Markov's inequality, the probability that a sample falls outside this ball is bounded by:

$$\Pr(\|X\|^2 \geq 2\sigma^2 d) \leq \frac{\mathbb{E}[\|X\|^2]}{2\sigma^2 d} = \frac{1}{2}$$

Thus, the true probability mass of the ball is $\Pr(X \in \mathcal{B}) \geq 1/2$.

For any query point $x \in \mathbb{R}^d$, define the halfspace pointing toward x as $H_x = \{y \in \mathbb{R}^d : \langle x, y\rangle \geq 0\}$. Because the Gaussian distribution is spherically symmetric and centered at the origin, exactly half of the mass of $\mathcal{B}$ lies in H_x. Therefore, the true probability that a random sample falls into their intersection is:

$$p_x := \Pr(X \in \mathcal{B} \cap H_x) = \frac{1}{2}\Pr(X \in \mathcal{B}) \geq \frac{1}{4}$$

We require the empirical fraction of samples in $\mathcal{B} \cap H_x$ to tightly concentrate around its true expectation uniformly over all possible directions $x \in \mathbb{R}^d$. The set of such halfspaces $\{H_x : x \in \mathbb{R}^d\}$ has VC dimension of d. Intersecting this class with a fixed set $\mathcal{B}$ does not increase the VC dimension. By uniform convergence, there exists a universal constant $c > 0$ such that with probability at least $1 - \delta$:

$$\sup_{x \in \mathbb{R}^d} \left| \frac{1}{m} \sum_{i=1}^{m} \mathbf{1}_{\{X_i \in \mathcal{B} \cap H_x\}} - p_x \right| \leq c\sqrt{\frac{d + \ln(1/\delta)}{m}}$$

We choose the sample size m sufficiently large such that this uniform error is bounded by $1/8$. Specifically, this requires $m \geq C(d + \ln(1/\delta))$ for $C = 64c^2$.

Under this condition, with probability $1 - \delta$, the empirical fraction of samples falling in $\mathcal{B} \cap H_x$ is at least $p_x - 1/8 \geq 1/4 - 1/8 = 1/8$ for all x simultaneously. Let $S_x = \{X_i : X_i \in \mathcal{B} \cap H_x\}$ be this subset of samples for a given x. We are guaranteed that $|S_x| \geq m/8$ for every $x \in \mathbb{R}^d$.

Conditioned on this uniform convergence event, we evaluate the ratio $r(x) = p(x)/\mathcal{L}_{\text{KDE}}(x)$ for an arbitrary $x \in \mathbb{R}^d$. We lower bound the KDE sum by dropping all kernel components except those corresponding to the samples in S_x:

$$\mathcal{L}_{\text{KDE}}(x) = \frac{1}{m}\sum_{i=1}^{m} K_h(x - X_i) \geq \frac{1}{m} \sum_{X_i \in S_x} \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|x - X_i\|^2}{2\sigma^2}\right)$$

For every $X_i \in S_x$, we know two facts by definition of the set: $\|X_i\|^2 \leq 2\sigma^2 d$ and $\langle x, X_i\rangle \geq 0$. Expanding the squared distance yields:

$$\|x - X_i\|^2 = \|x\|^2 - 2\langle x, X_i\rangle + \|X_i\|^2 \leq \|x\|^2 + 2\sigma^2 d$$

Substituting this upper bound on the distance into each kernel exponent in the sum gives:

$$\begin{aligned}\mathcal{L}_{\text{KDE}}(x) &\geq \frac{1}{m} \sum_{X_i \in S_x} \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right) \exp(-d) \\ &= \frac{|S_x|}{m} \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right) \exp(-d)\end{aligned}$$

Because $|S_x| \geq m/8$ for all x, we have

$$\mathcal{L}_{\text{KDE}}(x) \geq \frac{1}{8}\exp(-d) \cdot p(x) \qquad \Rightarrow \qquad \sup_{x \in \mathbb{R}^d} \frac{p(x)}{\mathcal{L}_{\text{KDE}}(x)} \leq 8e^d := K$$

This completes the proof. □

Our proof easily extends to the setting where $h = c \cdot \sigma$ for constant $c \geq 1$ and yields $K = O((ce)^d)$. Further, the same proof idea generalizes to centrally symmetric sub-Gaussian distributions with similar bounds.

E Motivating Application: Algorithm-LLM Interaction at Inference-time

One of our motivations for studying MBO with expressive, black-box generative priors comes from algorithm-LLM interaction, specifically via inference-time alignment. Increasingly, real-world combinatorial optimization requires balancing strict global constraints (e.g., graph connectivity, efficiency) with informal, context-dependent local specifications (e.g., scenic preferences, stylistic coherence). While classical algorithms efficiently solve for global feasibility, they are brittle when handling qualitative or ambiguous requirements [60]. Similarly, while modern LLMs excel at interpreting open-ended requirements [5, 40], they consistently struggle to enforce global combinatorial properties. We illustrate this limitation empirically using a cycle detection task. As shown in Figure 2, when asked to find a length-k cycle in a graph containing a planted cycle, models such as GPT-4, Claude 3.5 Sonnet, and Claude 3.7 Sonnet show rapidly diminishing success rates as the cycle length k increases. This shows that although models continue to improve rapidly [9, 12, 35], their capacity for exact combinatorial search remains bounded by a "short-chain" reasoning frontier.

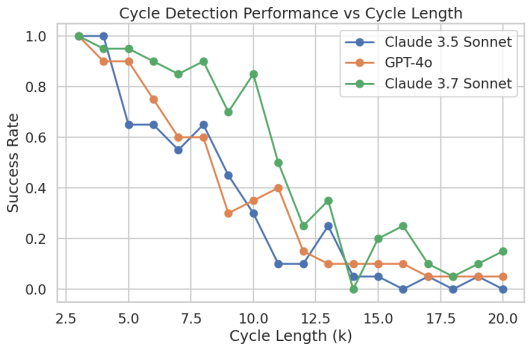


Figure 2: Success rate of the models in finding a length k-cycle in a graph as a function of k. For each k, the graph is constructed by starting with a cycle of length k and randomly adding $\lceil k/2 \rceil$ other edges. For each cycle length k, we generate 50 random instances and compute the success rate, i.e., the fraction of times the model returns a length-k cycle. See Appendix H for the exact prompt used.

While this combinatorial reasoning capability can occasionally be improved by fine-tuning a model for a specific global objective [30, 43], such a tuned model would lose its interpretive flexibility when presented with informal, context-dependent requirements at test time. This trade-off between interpretive flexibility and global optimality arises broadly in domains like robot navigation [52], molecular design, and automated scheduling [22].

Therefore, problems requiring both interpretive flexibility and enforcement of global constraints exceed the reach of either pure LLMs or pure classical algorithms. For example, in generating a scenic route between two locations, an LLM may expertly evaluate individual route segments for scenic value based on natural language criteria, but it struggles to ensure those segments connect to form a valid, continuous path [28]. Conversely, a combinatorial algorithm easily ensures path connectivity but cannot interpret the qualitative concept of "scenic-ness." Similar complementary trade-offs exist in conference planning [22], where an LLM can semantically analyze abstracts to cluster sessions, but a classical algorithm is required to schedule them into a conflict-free timetable. This interaction is therefore best studied as model-based optimization with expressive black-box generative priors, providing a high-level motivation for the current work.

Within this space, our motivation for MBO comes from inference-time alignment, where a fixed base model is adapted at test time using a reward signal or cost function. Indeed, TopIFT and ALDrIFT are analogous to a test-time training step in LLMs [50], in that model fitting (that is, fine-tuning) is performed separately for each problem instance. See also [8, 20, 27] for other empirical inference time alignment procedures. Our framework provides complementary theoretical bounds (under the coarse learnability assumption) to this line of work.

F Empirical Dynamics of Algorithm-LLM Interaction

The rigorous theoretical guarantees established in Section 4 and Appendix C rely on analytic generative models. Given the difficulty with showing corresponding results for algorithm-LLM interaction, we treat the coarse learnability condition (Assumption 2) in this section as an empirical heuristic rather than a provable guarantee. The primary contribution of this work is theoretical, and the goal of these experiments is merely to qualitatively study the empirical dynamics of these models. Specifically, we aim to test whether iterative fitting on a very small number of low-cost samples can heuristically shift an LLM's distribution in the spirit of coarse learnability, namely, by demonstrating broader coverage over lower-cost regions.

We use a primitive model, GPT-2, since it will demonstrate that our results are not a byproduct of the complex reasoning capabilities of frontier models, and applies to simple models as well. We also use the heuristic algorithm TOPIFT (Algorithm 1), which does not require estimating the LLM probabilities $\mathcal{L}(s)$ (Assumption 1) that is required for implementing ALDRIFT. This algorithm has the advantage of being simpler and implementable with frontier models where estimating probabilities directly is harder. We therefore present results for this heuristic, noting that the results for ALDRIFT are comparable and omitted for brevity. Accordingly, these experiments should be interpreted as qualitative evidence for the learnability intuition underlying Assumption 2 for LLMs rather than as an evaluation of ALDRIFT versus TOPIFT.

F.1 Line Scheduling

We focus on a simplified line scheduling problem here, and also present results for the low degree spanning tree problem in Appendix F.2. In the line scheduling problem, there are K stations on a line labeled $\{1, \ldots, K\}$. For $1 \leq i < K$, the travel time between station i and $i+1$ is t_i. Each station i has opening time o_i. The user specifies the intervals $[\ell_i, u_i]$ for $1 \leq i \leq K$ that capture the lower bound and the upper bound on how long they wish to spend at each station. The goal is to find a visit duration v_i for each $i \in \{1, \ldots, K\}$ that specifies how much time the user should spend at each station; let $\vec{v}$ be the vector of these durations. If the user reaches a station before time o_i, then they need to wait there until time o_i. We constrain all numbers, including those generated by the LLM, to be integers.

We split the problem constraints between a combinatorial algorithm that provides zeroth-order feedback on the objective $d(\cdot)$ and the LLM as follows.

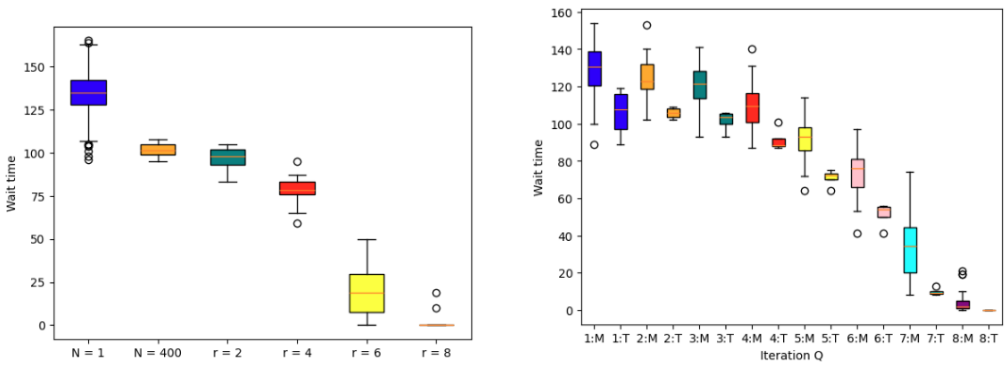
Algorithm's Constraints. Given a solution $\vec{v}$, for $i \geq 2$, let a_i be the arrival time at station i, and w_i be the waiting time for the station to open; we assume $w_1 = o_1 = 0$. The total wait time is $d(\vec{v}) = \sum_{i=1}^{K} w_i$, and this is the algorithm's cost (or objective function).

LLM's Constraints. The LLM's constraints correspond to the visit time intervals $[\ell_i, u_i]$ for each point i. Its cost is the total violation of the visit time computed as $\sum_{i=1}^{K} \eta_i$, where η_i is the distance of the visit time v_i from the interval $[\ell_i, u_i]$. This quantity is 0 if $v_i \in [\ell_i, u_i]$.

The LLM (GPT-2) is initially fine-tuned to satisfy only the local visit duration intervals, making it "oblivious" to the global waiting time constraint. This yields the model $\mathcal{L}_0$. We generate instances with $K = 10$ and where each t_i, ℓ_i, u_i is an integer in the range $[1, 20]$ with the constraint $\ell_i \leq u_i$. For $i \geq 2$, we set $o_i = o_{i-1} + t_{i-1} + u_{i-1}$, so that there is a solution with wait time 0 if we set $v_i \geq u_i$ for each i. The only solution among these that also has a visit time violation 0 is the solution that sets $v_i = u_i$ for each i.

In our experiment, we will consider the specific instance where the visit time bounds are set as $\ell_i = 1$ and $u_i = 20$, hence being vacuous, and all travel times are set to $t_i = 10$. For this instance, the optimal visit time vector is $\langle 20, \ldots, 20 \rangle$, and this vector has wait time zero and visit time violation of 0. If the visit time at a station is $v < 20$, then the solution has to wait $20 - v$ steps before the next station opens.

In Figure 3(a), we show the box plot, for 20 runs of the algorithm, of the algorithm's cost (total wait time) for the BEST-OF-LLM baseline that draws N samples from $\mathcal{L}_0$ and outputs the solution with minimum cost $d(\cdot)$. We plot this for $N = 1$ and $N = 400$ samples. Note that the median algorithm's cost for BEST-OF-LLM for $N = 400$ is at least 100. We observe that for any station, if v is the visit time assigned to the station in a sample from $\mathcal{L}_0$, then assuming this is a uniform distribution on



(a) (b)

Figure 3: Box-plots of algorithm's cost for the setting where the LLM's visit time constraint is $[1, 20]$. In (a), for 20 runs of TOPIFT, the left two plots are the distribution of the algorithm's cost for BEST-OF-LLM(N) baseline, while the right four plots are for TOPIFT after r iterations. In (b), for one run of TOPIFT, for different values of iteration r, the box-plot "$r : M$" is the distribution of the $m \cdot M = 48$ samples generated by the previous model, while "$r : T$" is the distribution of $m = 4$ samples among these which have lowest algorithm's cost (waiting time), and which are used for fine-tuning the new model.

$[1, 20]$, we would have $\mathbb{E}[v] = 10.5$, so that the expected wait time is $(20 - \mathbb{E}[v]) \cdot (K - 1) = 85.5$. This roughly agrees with the baseline. Further note that $\Pr[v = 20] \leq 0.05$. This is because the LLM is initially only aware of the bound $v \in [0, 20]$, and hence, is likely to choose an integer visit time uniformly at random from this range. It is unaware of the global waiting time constraint. Assuming the dimensions behave independently, this means $\Pr[\text{Wait time } = 0] \leq 10^{-10}$, so that merely sampling from $\mathcal{L}_0$ is very unlikely to find a solution with optimal waiting time.

We then compare this to TOPIFT (Algorithm 1) that fine-tunes $\mathcal{L}$ iteratively using zeroth-order feedback about $d(\cdot)$. For TOPIFT, we set the parameters as $m = 4$, $M = 12$, and $Q = 8$, so that the total number of samples used is comparable to $N = 400$ used by the baseline. We consider 20 independent runs of the algorithm, and in Figure 3(a), we show the Box plot of the best waiting time obtained after r iterations of the outer loop of TOPIFT. Though the BEST-OF-LLM baseline yields waiting time (objective $d(\cdot)$) of at least 100 even with $N = 400$ samples, we note that TOPIFT improves on this even with $r = 4$, generating the optimal visit-time vector almost always (wait time = 0) at $r = 8$ iterations.

In Figure 3(b), we show, for one run of the algorithm, the distribution of the $m \cdot M = 48$ samples generated by the model L_{r-1} for $1 \leq r \leq 8$, and the distribution of the best $m = 4$ samples (according to the algorithm's cost) among these used for fine-tuning L_r. Note that initially the fine-tuning samples move the model only slightly, while in later iterations the fine-tuned model produces solutions with substantially lower algorithmic cost than the fine-tuning data itself. Thus, the update is not merely memorizing the elite set, but shifting the model's distribution further toward the low-cost region. This behavior is qualitatively analogous to Assumption 2: a learner trained on a small sample from a target distribution produces a proposal that covers that target directionally better than the raw sample alone would suggest.[8] While Figure 3(b) does not verify the density-ratio conclusion of Assumption 2, it provides empirical evidence for the kind of coarse, mass-covering extrapolation that the assumption posits.

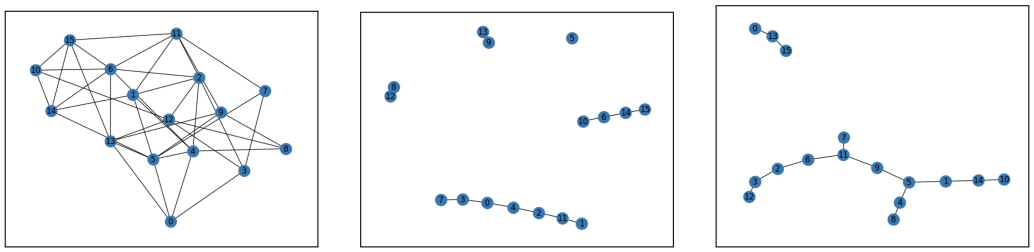
This phenomenon, where the model collapses to the optimal solution [47] demonstrates a remarkable ability of the LLM to extrapolate beyond the limited fine-tuning data. The iterative guidance from the zeroth-order feedback, even with a small sample size, enables the LLM to learn complex combinatorial constraints that it was initially unaware of.

[8]We note that the same effect cannot be observed with only one round of fine-tuning — if we set $Q = 1$, $m = 30$, and $M = 12$, and take the best waiting time of the samples from $\mathcal{L}_0$ and 40 samples from $\mathcal{L}_1$ (so that the total number of samples remains 400), we observe the waiting time is 81. This shows the advantage of the iterated framework.

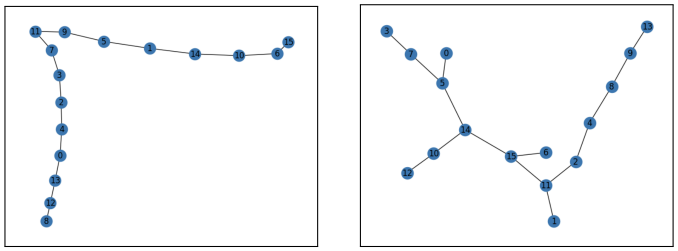
F.2 Low Degree Spanning Tree

In the low degree spanning tree problem, the input is a graph with $n = 16$ vertices numbered $0, 1, \ldots, 15$. We start by including the Hamiltonian path that includes all edges of the form $(i, i+1)$ for $i \in \{0, 1, \ldots, 14\}$. Next, for every pair of vertices, we add an edge between this pair with probability $p = 0.4$, removing duplicates. Note that the expected degree of a vertex lies in $[6, 8]$, and the graph has approximately 60 edges in expectation.

The goal is to output a spanning tree minimizing the number of vertices with degree larger than two; in the ideal case, a Hamiltonian path. We split this problem between a combinatorial algorithm that captures the objective $d(\cdot)$ and LLM as follows:



(a) Input graph. (b) BEST-OF-LLM, $N = 1$. (c) BEST-OF-LLM, $N = 600$.



(d) TOPIFT for $Q = 3$. (e) BEST-OF-ALG, $N = 1$.

Figure 4: Input graph and outputs of TOPIFT and the various baselines. Note that (b) is simply the output of the base model $\mathcal{L}_0$. The output of BEST-OF-ALG for $N = 600$ and random spanning trees is comparable to (e), which means the model $\mathcal{L}_0$ assigns comparable probabilities to different random spanning trees.

Algorithm's Cost. The algorithm simply wants to output a spanning tree. Given a solution x as a list of edges, its cost $d(x)$ is the number of connected components induced by x minus 1, so that the optimal cost is 0, and the maximum possible cost is $n - 1 = 15$.

LLM's Cost. These correspond to the local node-wise degree constraints. We measure the cost of the LLM's solution (set of edges) as the number of vertices whose degree is greater than 2.

We fine-tune GPT-2 on forests with maximum degree two, yielding the base model $\mathcal{L}_0$. We then run the TOPIFT heuristic (Algorithm 1) with $m = 4$, $M = 50$, and $Q = 3$. Note that the total number of samples is $N = m \cdot M \cdot Q = 600$. We compare to two baselines: In the BEST-OF-ALG baseline, we generate N random spanning trees (all with cost $d(\cdot) = 0$) and compute the probability assigned by $\mathcal{L}_0$ to each of them, choosing the best one. In the BEST-OF-LLM baseline, we generate N solutions from $\mathcal{L}_0$ and score them according to the cost $d(\cdot)$, choosing the best one.

In Fig. 4, we illustrate the solutions found for a specific random test instance by the different baselines and TOPIFT. Note that samples from $\mathcal{L}_0$ satisfy the degree constraints, but are disconnected. BEST-OF-LLM for $N = 600$ (Fig. 4(c)) improves connectivity but still finds a solution that is not only disconnected, but has degree violations. On the other hand, TOPIFT finds a Hamiltonian path[9], hence achieving optimum cost for both the algorithm and LLM. Finally, note that random spanning trees (Fig. 4(e)), though connected, have many degree violations. (The latter is also essentially what BEST-OF-ALG for $N = 1$ and $N = 600$ generate.) This visually shows that TOPIFT is

[9]This does not happen for all 15 test instances; however, TOPIFT finds one connected component in all instances.

incorporating both the algorithm's and the LLM's constraints in a non-trivial fashion, indeed, finding the "optimum" solution on this instance. This showcases the power of the LLM in learning a complicated combinatorial constraint from a few samples, which aligns with coarse learnability.

G Additional Discussion on ALDRIFT and Coarse Learnability

G.1 Connecting TOPIFT to ALDRIFT

Recall TOPIFT from Algorithm 1. We now describe a smooth approximation of TOPIFT that avoids the hard selection of the m smallest-cost samples, and show that this naturally leads to a procedure akin to ALDRIFT.

Towards this end, we position TOPIFT within a framework in which the algorithm tries to make the distribution $\mathcal{L}_r$ a progressively good approximation to the distribution p_T^*. Unfortunately, obtaining a closed form for $\mathcal{L}_r$ in TOPIFT seems difficult because we choose the samples with the smallest $d(\cdot)$, i.e., hard-min. To circumvent this difficulty, we will use a soft-min distribution to approximate this step; this leads to a simulated annealing algorithm that we describe next.

Note that the construction of S'_r in Algorithm 1 can be split into two steps:

1. We generate s from the distribution $\mathcal{L}_{r-1}$. If this is repeated $m \cdot M$ times, this corresponds to generating S_r.
2. We select the samples from S_r with a minimum $d(\cdot)$. This can be smoothly approximated by keeping each $s \in S_r$ with probability $e^{-\tau_r \cdot d(s)}$. This is a "soft-min" that favors solutions s with smaller $d(s)$. The parameter τ_r can be chosen so that if $D = \max_{s \in S} d(s)$, then $\tau_r = \frac{\ln M}{D}$. This ensures that any $s \in S$ is sub-selected in this step with probability at least $\frac{1}{M}$, so that $\mathbb{E}[|S'_r|] \geq m$.

Therefore, the sampling distribution of S'_r is proportional to $\mathcal{L}_{r-1}(s) \cdot e^{-\tau_r \cdot d(s)}$. If we assume that $\mathcal{L}_r$ is faithfully learned from these samples, we obtain $\mathcal{L}_r(s) \propto \mathcal{L}(s) \cdot e^{-\sum_{t=1}^{r} \tau_t \cdot d(s)}$. We can now choose the number of iterations Q so that $\sum_{t=1}^{Q} \tau_t \approx T$, which leads to the distribution p_T. This yields a smoothed variant of TOPIFT, analogous to simulated annealing, where the temperature $\tau = \sum_{t=1}^{r} \tau_t$ is gradually increased across iterations and a model p_τ is learned. The gradual increase ensures that the set S'_r at each iteration remains sufficiently large.

A key challenge is that in each iteration, the model $\mathcal{L}_r$ only approximates the sampling distribution S'_r. This approximation can be quite crude, since the distribution $p_\tau(s) \propto \mathcal{L}(s) \cdot e^{-\tau \cdot d(s)}$ encodes combinatorial constraints that the model cannot fully capture, even with many samples. Consequently, errors can accumulate multiplicatively across fine-tuning iterations, making it difficult to formally prove convergence for the simple heuristic. We address this issue by incorporating a Metropolis–Hastings step to select the top samples. This allows the approximately learned model $\mathcal{L}_{r-1}$ to serve as an efficient proposal distribution, while the acceptance rule ensures sampling from the true target distribution p_τ, preventing error accumulation. This leads to the ALDRIFT algorithm.

G.2 ALDRIFT vis–a–vis Model Reference Adaptive Search (MRAS)

Our analysis of ALDRIFT can be interpreted as a finite-sample robustification of the model reference adaptive search (MRAS) framework [18] via the Metropolis–Hastings proposal step. In the MRAS framework, one defines a sequence of ideal target distributions, $g_k(s)$, and updates a parametric model $f(\cdot; \theta_k)$ to approximate g_k by minimizing the KL-divergence $D_{\text{KL}}(g_k \| f_\theta)$. In our instantiation, $g_k(s) \propto \mathcal{L}_0(s) e^{-\tau_k d(s)}$.

Standard MRAS theory guarantees $f(\cdot; \theta_k)$ converges to a point mass at the globally optimal solution for d. However, this analysis relies both on exactly updating the parameter θ via infinite samples as well as the specific structural properties of natural exponential families to ensure the optimum always remains within the support of this generated distributions [18]. Therefore, this theory does not provide explicit sample complexity bounds. In the finite-sample regime, projecting a complex target p_τ onto a generative model introduces empirical errors. Without specific statistical guarantees, there is no assurance that a polynomial number of samples suffices to maintain coverage of the optimal region, and as mentioned before, this aspect may break the algorithm.

Our framework identifies the statistical assumption (coarse learnability) and algorithmic modification (utilizing the coarse generative model as a proposal for Metropolis–Hastings rather than sampling from it directly) needed for achieving polynomial sample complexity in classical MBO. While methods like MRAS rely on the rigid geometry of exponential families to guarantee support preservation, Assumption 2 requires that the learner captures the target's mass within a polynomial factor. Our framework therefore shifts the burden from architectural correctness (choosing the right parametric family) to statistical expressivity (learning a cover), making the theory applicable to settings where exact parametric alignment is impossible. We presented a simple case of such model mis-match in Appendix D.1.

H Prompts for Empirical Results

Finding Cycle in a Graph

"""Identify a cycle of length exactly {k} in this undirected graph. Edges: {edges}
Think step by step. Finally the last line of your response must be the final output. The last line MUST be a list like [0,1,2,...,0] or 'No cycle found'. No other text:"""

Line Scheduling

""" There are {n} museums on a line. They are numbered from 0 to {m}. I'm currently located at museum 0 and the current timestamp is 0. I want to visit all these museums one by one in a sequence. Each museum has an opening time. If I reach a particular museum before it opens then I may have to wait. The opening times for the museums are as follows:
[**opening times go here**]
In addition, the following list of {m} numbers contains the time to travel from museum i to i+1. So the first number is the time to travel from 0 to 1 and so on:
[**travel times go here**]
Finally, I have certain constraints in terms of the minimum and maximum amount of time I want to visit each museum. This is described as the following list of arrays:
[**constraints go here**]
Give me a schedule in terms of a list of {n} numbers describing how much time I should spend at each place so that all my constraints are satisfied and at the same time my total wait time is as little as possible. Do not use code. Simply output the list of {n} numbers (one per line) and nothing else."""

Spanning Tree

"""You are given a graph with vertices labeled from 0 to {num_vertices-1}. Each line below lists an edge of the graph as (i,j).
[**edges go here**]
Your goal is to output a list of H of a subset of the edges that form a spanning tree, i.e., the subgraph induced by H should be connected. Furthermore, each vertex should appear in at most {deg} times in the list. Simply output the list of edges and nothing else. Format your answer by producing one edge per new line."""

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 abstract and introduction accurately state the paper's contributions, which are the introduction of the coarse learnability assumption, the ALDRIFT algorithm, and the corresponding theoretical and empirical results.

2. **Limitations**

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

   Answer: [Yes]

   Justification: The paper acknowledges the limitations of Assumption 2 and discusses avenues for future work to better characterize it in Section 7.

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 theoretical results are stated as theorems, and their assumptions are explicitly provided. Full proofs for all theorems are included in the appendices.

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: The paper provides detailed descriptions of the experimental setups, including the models used (GPT-2), algorithm parameters, and the specific prompts used, in Section 6 and Appendix F and Appendix H.

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: We provide code for all experiments in the supplemental material.

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: Section 6 and Appendix F provide the necessary details for the experimental settings, including the generation of problem instances and the hyperparameters chosen for the algorithm.

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: The empirical results in Fig. 3 are presented as box plots over 20 independent runs, which illustrates the statistical distribution and variability of the outcomes. Similarly for the plots in Fig. 1.

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: [No]

Justification: The experiments are not computationally intensive, so we did not provide detailed compute resource information.

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: This research is theoretical and algorithmic, and to the best of our knowledge, it conforms to the NeurIPS Code of Ethics.

10. **Broader impacts**

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

    Answer: [No]

    Justification: This is foundational theoretical work on optimization algorithms. While it could lead to more efficient and reliable AI systems, we believe a detailed discussion of broader societal impacts is out of the scope of this work.

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 introduces an algorithm and does not release new models or datasets that would pose a high risk for misuse.

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: We use the publicly available GPT-2 model and properly cite the original work.

13. **New assets**

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

    Answer: [Yes]

    Justification: We release the code for our experiments, which is documented to allow for the reproduction of our results.

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 paper does not involve crowdsourcing or research with human subjects.

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 paper does not involve research with human subjects.

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: [Yes]

Justification: Yes, the Acknowledgments and Disclosure of Funding section describes how a large language model was used to assist in the research and writing process.