

Sample-Efficient Optimisation over the Outputs of Generative Models

Samuel Willis^{*,1}

Paul Duckworth⁶

Jack Simons⁶

Aleksandra Kalisz⁶

Krisztina Sinkovics⁶

Noam Ghenassia⁶

Shikha Surana⁶

Henry T. Oldroyd²

Alexandru I. Stere⁴

Dragos D. Margineantu⁵

Carl Henrik Ek^{1,3}

Henry Moss^{1,2}

Erik Bodin^{*,1,7}

¹University of Cambridge

²Lancaster University

³Karolinska Institutet

⁴Boeing Commercial Airplanes

⁵Boeing AI

⁶InstaDeep

⁷Monumo

Abstract

Modern generative AI models, such as diffusion and flow matching models, can sample from rich data distributions. However, many applications, especially in science and engineering, require more than drawing samples from the model distribution: they require searching within this distribution for samples that optimise task-specific criteria. In this work, we propose O^3 (Optimisation Over the Outputs of Generative Models), a method for sample-efficient black-box optimisation over continuous-variable diffusion and flow-matching models. O^3 is built around *surrogate latent spaces*: low-dimensional Euclidean embeddings that can be extracted from a generative model without additional training. The resulting representations have controllable dimensionality and support the direct application of standard optimisation algorithms. We show, on image and protein design tasks, that surrogate-space optimisation finds substantially higher-scoring samples than standard sampling or optimisation in the original latent space. Our method is model- and optimiser-agnostic, incurs negligible additional cost over standard generation, and requires no retraining or fine-tuning of the generative model.

1 Introduction

Black-box optimisation algorithms aim to find an optimum of a function whose explicit form is unknown, but which can be queried through evaluations. Their performance is strongly affected by the dimensionality of the search space and the structure of the objective function. In high-dimensional settings, one way to simplify the problem is to optimise over an alternative representation of the data, for example one that is lower-dimensional or in which the objective is smoother or more stationary.

One route to such representations is to embed the data in a low-dimensional space using projections. Examples include REMBO [51], which uses random projections, methods that select axis-aligned subspaces [12], and one-dimensional search sequences [35]. Another route is to use representations learned by generative models. While many latent variable models are applicable in this setting, variational autoencoders (VAEs) [24] have been especially influential. VAEs can learn low-dimensional

^{*}Equal contribution.

continuous representations amenable to optimisation; however, the local structure of the variational distribution can lead to latent regions with low data support, making it difficult to faithfully unravel the data manifold. Several works address this limitation by fine-tuning the model during optimisation [18, 49, 33, 8] or by constraining the generation process directly [6, 34].

In recent years, generative models have undergone a step-change in performance. Modern stochastic interpolation-based generative models [1], such as diffusion and flow-matching models, provide a more general framework and substantially improved generative capabilities. However, unlike VAEs, these models do not directly provide a low-dimensional latent space suitable for optimisation. Existing optimisation approaches for such models therefore typically modify or condition the generative process itself. This includes training task-specific models from scratch [27], fine-tuning pre-trained models for a given objective [15, 10], requiring gradients, or learning auxiliary models of the objective function [19, 57, 37, 48, 56, 55]. While these approaches demonstrate that stochastic interpolant models can be used for optimisation, they are typically task-specific and require either training the generative model or learning an additional auxiliary model, limiting their general applicability.

In this paper, we introduce a general framework for extracting optimisation-suitable latent representations from diffusion and flow-matching models. The approach is both model- and optimiser-agnostic, supports expensive gradient-free objectives, and incurs negligible additional cost. These search spaces are defined by example latents and mapped to valid model outputs, allowing standard optimisers to operate directly over the outputs of stochastic interpolation-based generative models. This enables highly sample-efficient optimisation without retraining the generative model.

“A photo of two astronauts on the moon playing badminton while drinking tea”

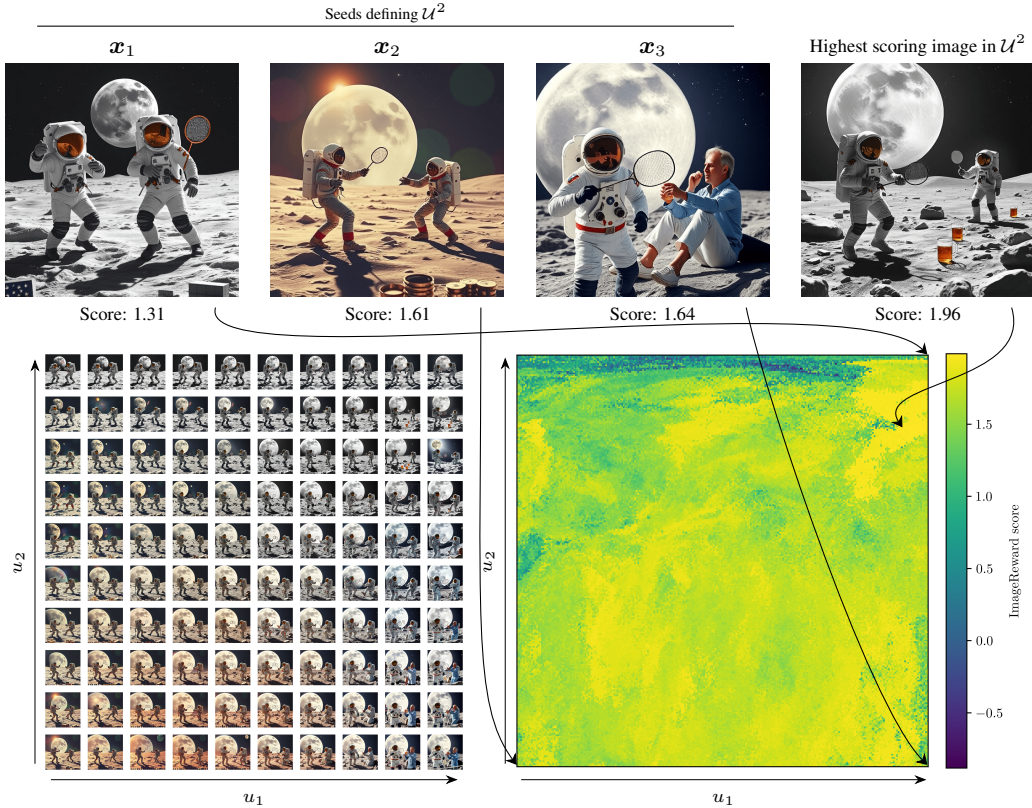


Figure 1: **Surrogate latent spaces.** (*bottom left*) Generations associated with a grid over a 2-dimensional surrogate latent space, formed using the latent vectors corresponding to the examples x_1 , x_2 , and x_3 for the FLUX.1-schnell [29] rectified flow model. (*bottom right*) The ImageReward score for a target prompt (the objective function) over a dense grid (256×256) of generations from our surrogate space show rich structure that can be exploited by standard optimisers. The example (‘seed’) images were obtained by sampling the model using the target prompt but they fail to follow it; by navigating our surrogate space, we can find images with better alignment.

2 Background

Black-box optimisation targets problems where the objective function is explicitly unknown *a-priori* but can be queried through function evaluations. These evaluations may be expensive, noisy, or derivative-free, and given that the function is not directly accessible such problems are referred to as “black-box”. A typical black-box optimisation problem can be written as $\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})$, where $f: \mathcal{X} \rightarrow \mathbb{R}$ is the objective function, and $\mathcal{X}$ is the domain of interest.

Many problems where we would like to apply black-box optimisation are characterised by high-dimensional and highly structured data, making it challenging to optimise f directly. One way to address this challenge is to optimise over a representation of the data that is better suited for search. This is commonly referred to as Latent Space Optimisation (LSO) [28, 17, 32, 31], and involves searching over a latent space $\mathcal{Z}$ instead of directly in $\mathcal{X}$:

$$\mathbf{z}^* = \arg \max_{\mathbf{z} \in \mathcal{Z}} f(g(\mathbf{z})), \quad (1)$$

where $g: \mathcal{Z} \rightarrow \mathcal{X}$ is a generative model. Traditionally, LSO has relied on low-dimensional generative latent-variable models, most prominently VAEs [24].

Diffusion [21, 47, 46] and flow matching [30] models have been shown to learn latent representations capable of modelling highly structured, high-dimensional data. When sampled deterministically, a latent variable $\mathbf{z} \sim p(\mathbf{z})$ fully specifies the generated sample $\mathbf{x}$, inducing a mapping $g: \mathcal{Z} \rightarrow \mathcal{X}$ that can in principle be used in Equation 1. However, the latent variables in these models typically have the same dimensionality as the data, making direct optimisation in $\mathcal{Z}$ challenging. A further challenge is that, until recently, it was unclear how to safely navigate these latent spaces beyond sampling. Bodin et al. [5] address this by showing that the latents are statistically structured: high-quality generations arise from latents whose statistics match typical draws from $p(\mathbf{z})$, rather than from arbitrary points that may have high density under $p(\mathbf{z})$ but are atypical in high dimensions. Consequently, optimisation algorithms querying the latent space must preserve these statistics to remain within the input regime where the generative model operates reliably. While Bodin et al. [5] provide a framework for preserving these statistics, they do not address optimisation or how to construct effective search spaces. This is the focus of our paper.

3 Surrogate latent spaces

Here we begin with a simplified derivation of our construction, restricted to cases where the latent distribution of the generative model is of the form $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Full details of how to apply to more general latent distributions are included in Appendix A. We start from existing random subspace methods, then proceed to identify and address desirable properties required for valid latent manipulation and effective search in the black-box setting.

A standard approach in high-dimensional optimisation is to restrict the search to a low-dimensional subspace. REMBO [51] follows this approach by introducing a random Gaussian embedding mapping low-dimensional ($K \ll D$) proposals $\mathbf{w}$ to the full space

$$\mathbf{z}^* = \mathbf{Z}\mathbf{w}, \quad \mathbf{w} \in \mathbb{R}^K, \quad (2)$$

where $\mathbf{Z} \in \mathbb{R}^{D \times K}$ has i.i.d. Gaussian entries, $\mathbf{Z}_{ij} \sim \mathcal{N}(0, 1)$, and $\mathbf{z}^* \in \mathbb{R}^D$ denotes the resulting proposal in the full D -dimensional space.

Ensuring high-quality generations. For general $\mathbf{w}$, the REMBO embedding induces

$$\mathbf{z}^* = \mathbf{Z}\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \|\mathbf{w}\|_2^2 \mathbf{I}_D),$$

which does not, in general, match the latent distribution $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$ expected by the generative model g in Equation 1. Consequently, the resulting latents may fall outside the typical input regime on which diffusion and flow-matching models operate reliably. As shown by Bodin et al. [5], such mismatched latent statistics can lead to poor or implausible generations. To avoid this, we use Latent Optimal Linear combinations (LOL) [5]. In the Gaussian setting, LOL reduces to,

$$\mathbf{z}^* = \frac{\mathbf{Z}\mathbf{w}}{\|\mathbf{w}\|_2}, \quad \mathbf{w} \in \mathbb{R}^K. \quad (3)$$

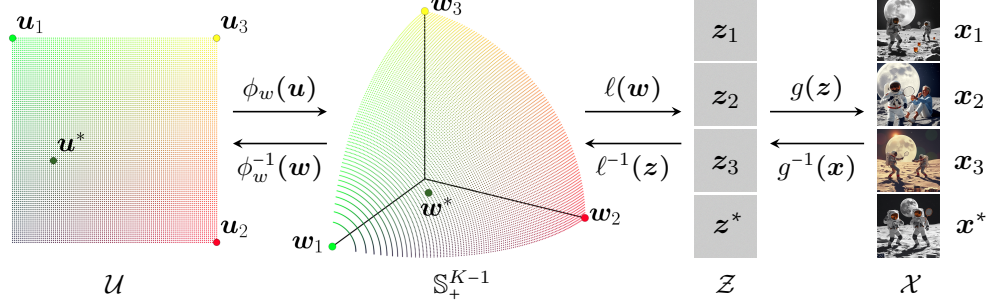


Figure 2: **Illustration of a surrogate latent space.** (left) A low-dimensional surrogate space $\mathcal{U}$ is mapped via the surrogate chart ϕ_w onto the positive orthant of the unit hypersphere (mid left), which is then mapped to latent variables z via ℓ (in the Gaussian case $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\ell(w) = \mathbf{Z}w$; see Appendix A for the general definition of ℓ). (mid right), before finally being decoded by the generative model g into objects x (right).

This construction ensures that $z^* \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$, matching the sampling statistics expected by the generative model and thereby enabling reliable generation [5].

Avoiding redundancy. While this construction ensures that the latent representation have the right statistical properties for the model, it introduces redundancy in the parametrisation. In particular, all weight vectors that differ by a positive scalar factor map to the same latent,

$$\frac{\mathbf{Z}w}{\|w\|_2} = \frac{\mathbf{Z}(\alpha w)}{\|\alpha w\|_2}, \quad \alpha > 0. \quad (4)$$

Thus, the mapping is many-to-one, and the search space contains multiple representations of the same solution. This redundancy is undesirable for optimisation, as it introduces flat directions along which the objective remains constant, causing the optimiser to waste evaluations exploring equivalent parametrisations. To remove this redundancy, we constrain the weights to have unit norm,

$$z^* = \mathbf{Z}w, \quad w \in \{w \in \mathbb{R}^K : \|w\|_2 = 1\} \quad (\text{i.e. } w \in \mathbb{S}^{K-1}). \quad (5)$$

If $\mathbf{Z}$ has full column rank, as occurs almost surely with independent Gaussian draws and $K \leq D$, then its null space is trivial. Consequently, distinct weight vectors induce distinct latents. This unit-norm constraint therefore removes the scaling redundancy while preserving the latent distribution, yielding a one-to-one, distribution-preserving parametrisation.

Removing curvature. While restricting w to the unit sphere removes redundancy, it introduces a new challenge: the search space is now a curved manifold $\mathbb{S}^{K-1}$. This curvature complicates optimisation, as standard black-box methods do not naturally handle manifold constraints. Optimising directly in this space would therefore require either constrained optimisation in $\mathbb{R}^K$ or methods adapted to spherical geometry.

We are thus motivated to construct a bounded, unconstrained Euclidean space $\mathcal{U}$ in which optimisation can be performed, together with a bijection,

$$\phi_w : \mathcal{U} \rightarrow \mathbb{S}^{K-1}, \quad \mathcal{U} = [0, 1]^{K-1}. \quad (6)$$

This allows standard optimisation algorithms to operate in $\mathcal{U}$, while ϕ_w maps their proposals to unit-norm weight vectors w ; bijectivity is discussed in Appendix A.

An overview of the resulting pipeline is shown in Figure 2. Here, ℓ denotes the map from weights w to latent variables z . In the Gaussian setting considered in this section, this map reduces to $\ell(w) = \mathbf{Z}w$. For more general latent distributions, ℓ may additionally involve transport maps; see Appendix A for the full derivation.

The map ϕ_w can be interpreted as flattening the hypersphere, analogous to constructing a map of the globe. Such a mapping cannot preserve all geometric properties and some distortion is unavoidable [16]. The key question is therefore: how do we design ϕ_w to minimise this distortion?

We address this challenge in two ways. First, we restrict the domain to the positive orthant of the hypersphere, $\mathbb{S}_+^{K-1} \subset \mathbb{S}^{K-1}$. By restricting the chart to this smaller region, the resulting Euclidean

parametrisation incurs less severe distortion in practice. The second part of our approach, discussed in the next section, is to choose ϕ_w so that Euclidean distances in $\mathcal{U}$ approximately preserve the similarity structure induced by the hypersphere.

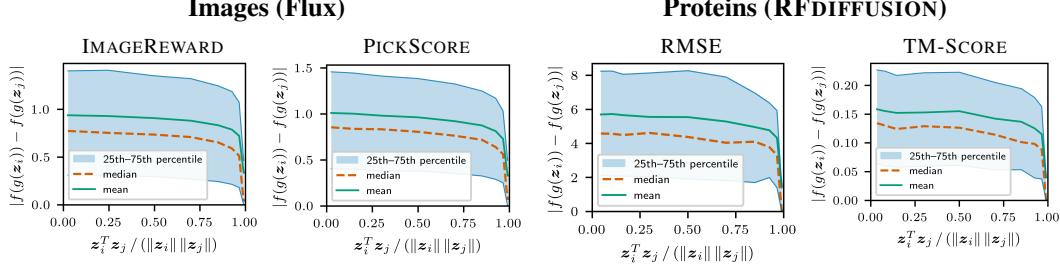


Figure 3: Relationship between latent cosine similarity and objective similarity across image and protein generation tasks. The first two panels show Flux [29] image generations evaluated with ImageReward and PickScore, while the last two panels show RFDIFFUSION [52] protein generations evaluated with RMSE and TM-score. For each metric, pairs of generated samples are grouped by the cosine similarity of their corresponding latents, and the plot shows the absolute difference in objective value between the paired samples. The black curve denotes the median, the green curve denotes the mean, and the blue shaded region denotes the 25th–75th percentile interval. Across modalities and metrics, higher latent cosine similarity corresponds to smaller differences in objective value, indicating that latent direction provides a useful notion of similarity for optimisation.

3.1 Hyperspherical geometry and cosine-preserving search spaces

The norm of samples from a high-dimensional unit Gaussian tightly concentrates around $\sqrt{D}$. Consequently, the variations in the latent representation is mainly encoded in the *direction* of the sample. Since the mapping g is a smooth neural network, it produces similar outputs for nearby directions. While smoothness guarantees this locally, it is not clear a priori that it extends to larger angles. However, as we show empirically in Figure 3, cosine similarity remains smoothly related to different objectives even at larger angular separations. We demonstrate this for images and prompt-following via ImageReward [54] and PickScore [25] in Flux [29], and for proteins using RMSE to a reference [52].

Motivated by this, we seek mappings ϕ_w such that Euclidean distance in $\mathcal{U}$ has a monotonic relationship with the cosine similarity of the corresponding points in latent space $\mathcal{Z}$,

$$v(\|\mathbf{u}_i - \mathbf{u}_j\|_2) \approx \frac{\mathbf{z}_i^\top \mathbf{z}_j}{\|\mathbf{z}_i\|_2 \|\mathbf{z}_j\|_2}, \quad \mathbf{z}_i = \mathbf{Z} \phi_w(\mathbf{u}_i), \quad (7)$$

where $v : \mathbb{R}_+ \rightarrow [0, 1]$ is a monotonically decreasing function. This induces a more regular and easier-to-optimize objective landscape over $\mathcal{U}$ when $f(\cdot)$ varies smoothly with cosine similarity.

In the Gaussian construction, there is a simple relationship between cosine similarity in latent space and inner products of the corresponding weight vectors. Let $\mathbf{z}_i = \mathbf{Z} \mathbf{w}_i$ and $\mathbf{z}_j = \mathbf{Z} \mathbf{w}_j$, where $\mathbf{Z} \in \mathbb{R}^{D \times K}$ has i.i.d. standard Gaussian entries and $\mathbf{w}_i, \mathbf{w}_j \in \mathbb{S}^{K-1}$. Then each marginal latent is distributed as $\mathcal{N}(\mathbf{0}, \mathbf{I}_D)$,

$$\mathbb{E}[\mathbf{z}_i^\top \mathbf{z}_j] = D \mathbf{w}_i^\top \mathbf{w}_j, \quad \text{so} \quad \mathbb{E}[D^{-1} \mathbf{z}_i^\top \mathbf{z}_j] = \mathbf{w}_i^\top \mathbf{w}_j.$$

Moreover, since $D^{-1} \mathbf{z}_i^\top \mathbf{z}_j$ concentrates around its expectation and $\|\mathbf{z}_i\|_2, \|\mathbf{z}_j\|_2$ concentrate around $\sqrt{D}$ in high dimensions, the latent cosine similarity satisfies,

$$\frac{\mathbf{z}_i^\top \mathbf{z}_j}{\|\mathbf{z}_i\|_2 \|\mathbf{z}_j\|_2} \approx \mathbf{w}_i^\top \mathbf{w}_j.$$

Thus, equation 7 simplifies to

$$v(\|\mathbf{u}_i - \mathbf{u}_j\|_2) \approx \mathbf{w}_i^\top \mathbf{w}_j, \quad \mathbf{w}_i = \phi_w(\mathbf{u}_i), \quad (8)$$

for some monotonically decreasing function $v : \mathbb{R}_+ \rightarrow [0, 1]$.

In Appendix D, we consider several choices for ϕ_w and empirically verify their suitability in Appendix F. We adopt a variant of the Knothe–Rosenblatt (KR) chart, which gives a smooth bijection from $\mathcal{U} = [0, 1]^{K-1}$ to $\mathbb{S}_+^{K-1}$, pushes $\mathbf{u} \sim \text{Unif}(\mathcal{U})$ to the uniform distribution on the hypersphere, and approximately preserves stationarity. By this, we mean that the induced weight-space similarity $\mathbf{w}_i^\top \mathbf{w}_j$ maintains an approximately monotonic relationship with the distance $\|\mathbf{u}_i - \mathbf{u}_j\|_2$.

3.2 Seed latents

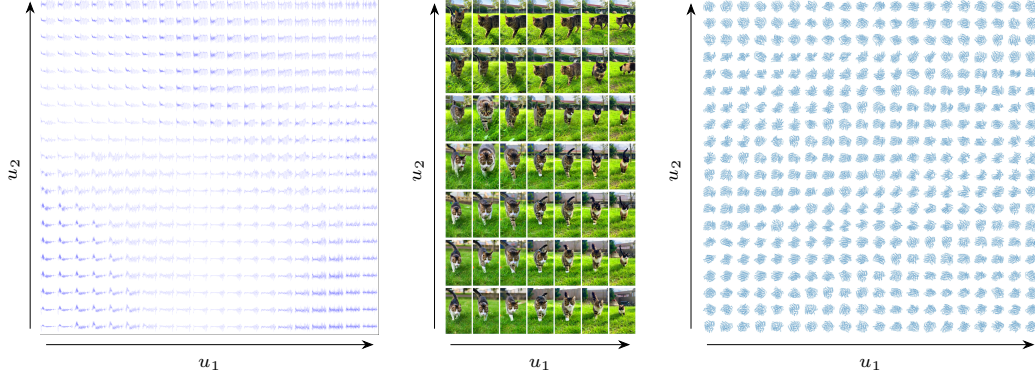


Figure 4: **We can build smooth surrogate spaces for any generative model.** (left) Waveform generations over a grid of a 2D slice of a 7D surrogate space formed from 8 seed latents and the 8256-dimensional StableAudio2.0 text-to-audio generation model [14]. (middle) The first frames of a similarly constructed grid of videos from the 4,308,480-dimensional HunyuanVideo text-to-video generation model [26]. (right) A grid of proteins over a 2D surrogate space formed from 3 seed latents corresponding to 3 proteins using RFDIFFUSION [52].

Building on the framework introduced by [5], we treat the columns of $\mathbf{Z}$ as *seed latents*, where $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_K]$ are valid samples from the model. As the generative model is a deterministic map each $\mathbf{z}_k$ corresponds to a generated object, which means our choice of $\mathbf{Z}$ defines a subspace formed of k generations from the model. New latent locations can now be formed as linear combinations of these seed latents. Crucially, the framework does not restrict the choice of seed latents beyond requiring that they exhibit sampling statistics consistent with the latent distribution [5]. This flexibility has two important consequences. First, the number of seed latents determines the dimensionality of the optimisation problem, and thus the expressivity of the search space. Second, seed latents can be chosen to correspond to favourable samples, inducing a subspace that interpolates between desirable regions of the data distribution.

We consider two practical strategies for selecting seed latents. The first is **filtering**, where we sample from the latent distribution and retain latents whose corresponding generations score highly under a given objective or scoring oracle. The second is **inversion** [46, 47], where we obtain latents corresponding to specific, known data points via the inverse of the generative model, $\mathbf{z}_i = g^{-1}(\mathbf{x}_i)$. Our method is agnostic to how these inversions are obtained, and can therefore make use of recent fast and accurate inversion methods, such as ExactDPM [22] and LightningInversionEdit [43].

The choice of seed latents therefore provides a high degree of control over the induced latent span, and consequently over the surrogate search space $\mathcal{U}$.

3.3 Summary

In summary, for $\mathbf{Z} \in \mathbb{R}^{D \times K}$ formed from i.i.d. Gaussian entries, the constructions considered in this work are:

REMBO $\mathbf{z}^* = \mathbf{Z}\mathbf{w}, \mathbf{w} \in \mathbb{R}^K$	LOL $\mathbf{z}^* = \frac{\mathbf{Z}\mathbf{w}}{\ \mathbf{w}\ _2}, \mathbf{w} \in \mathbb{R}^K$	O³ $\mathbf{z}^* = \mathbf{Z}\phi_w(\mathbf{u}), \mathbf{u} \in [0, 1]^{K-1}$
---	--	---

All three differ only in how the weights $\mathbf{w}$ are constrained or normalised. REMBO uses unconstrained weights, leading to a mismatch with the latent distribution. LOL corrects this by normalising the

linear combination weights w , but induces a many-to-one parametrisation. In contrast, we directly parametrise unit-norm weights via ϕ_w , yielding a one-to-one, distribution-preserving construction. Moreover, the map ϕ_w provides a bounded Euclidean search space $\mathcal{U} = [0, 1]^{K-1}$, enabling the use of standard black-box optimisation algorithms. We refer to this framework as O^3 .

4 Experiments

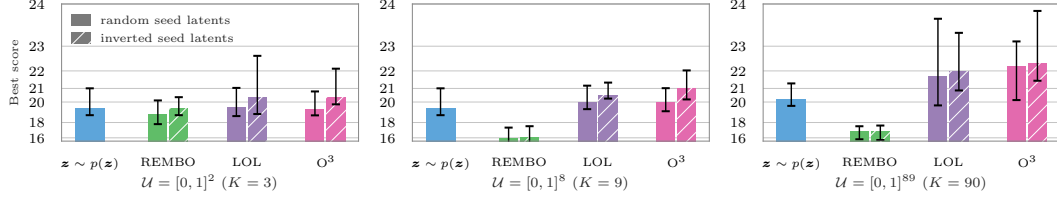


Figure 5: **Image optimisation with random and inversion-based seeds.** The three panels correspond to optimisation in surrogate spaces induced by 3, 9, and 90 seed latents, giving 2D, 8D, and 89D surrogate spaces, with optimisation budgets of 100, 100, and 1000 function evaluations respectively. Each bar shows the median and 90% confidence interval of the best PickScore achieved across 10 target prompts, comparing Bayesian Optimisation in search spaces formed by REMBO, LOL, and O^3 against Best-of- N — denoted as $z \sim p(z)$ — from the base model.

The experiments in Sections 4.1 – 4.3 are designed to verify basic properties: generality across modalities, sample quality and diversity within surrogate spaces, and the advantage of O^3 ’s surrogate space over alternative search space constructions when navigated by standard optimisers. Sections 4.4 – 4.5 then benchmark O^3 in more realistic protein-design settings: against existing guidance methods under matched oracle budgets, and in a regime where prior optimisation approaches are infeasible.

4.1 O^3 is model and data-modality agnostic

Our methodology acts on latent variables only and applies to any continuous-variable generative model with a latent distribution amenable to interpolation (Appendix A). We illustrate this on images and proteins (Figures 1, 7) and on audio and video in Figure 4 (see Appendix N). The formed spaces are smooth, low-dimensional, and retain full generative quality.

4.2 Good examples define spaces with better solutions

Goal. This experiment isolates a key requirement of surrogate search spaces: they must preserve both sample quality and diversity in order to support effective optimisation.

Setup. We follow the benchmark of Denker et al. [10], which evaluates generation from a diffusion model under an image prompt-following objective while maintaining sample diversity. This experiment tests whether high-scoring seed latents can define surrogate spaces that contain both high-scoring and diverse solutions (generations). We first sample a pool of random latents, select the highest-scoring latents as seeds, and construct surrogate spaces from these seed latents. We then evaluate the score and diversity of generations obtained from the resulting surrogate spaces, thereby assessing what regions of output space are made accessible by the chosen seeds. As reference points, we compare against state-of-the-art methods that explicitly train or fine-tune models for the same scoring function.

Results. Surrogate spaces constructed from high-scoring seed latents consistently yield generations with high mean scores while preserving substantial diversity. Their performance is comparable to methods that explicitly train or fine-tune models on the target objective, despite requiring no task-specific generative-model training. This suggests that high-scoring seed latents can define search spaces that retain access to high-quality and diverse solutions. Quantitative results and confidence intervals, together with comparisons to Importance Fine-tuning [10], DPOK [15], and Adjoint Matching [11], are reported in Table 1 together with experimental details in Appendix G.

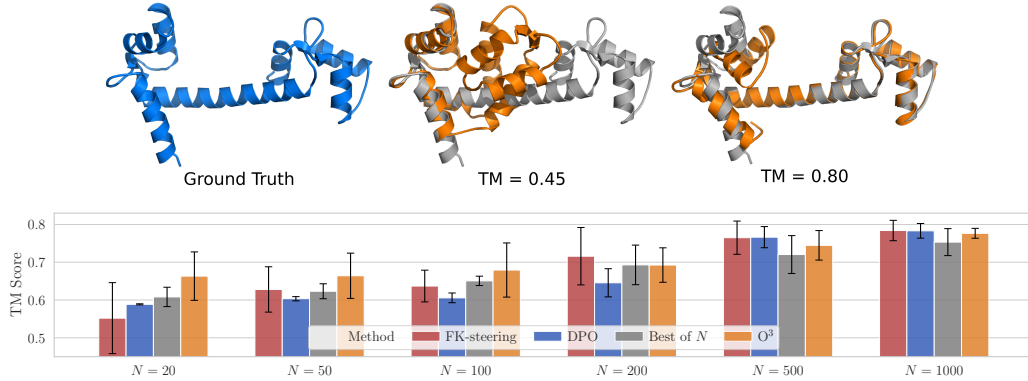


Figure 6: **Biasing protein structures of BOLTZ-2 toward a desired conformation.** *Top:* the desired 1CLL calmodulin conformation (left), a representative vanilla BOLTZ-2 generation in a different conformation (middle), and a O^3 -guided generation better matching the desired conformation (right); ground truth shown in grey for reference in the middle and right panels. *Bottom:* Comparison of O^3 +BO to FK-steering, DPO, and Best-of- N across six budgets of N oracle calls. Maximum TM-score to the 1CLL calmodulin structure is reported on the y-axis. Bars show the mean over 5 repetitions, and error bars ± 1 standard deviation. Higher is better.

4.3 Optimisation with O^3 versus existing search spaces

Goal. We construct a synthetic benchmark designed to be hard to solve by sampling alone: the model receives a generic prompt while being scored against a much more specific hidden target. The model is conditioned on a generic prompt ('A vehicle'), while the objective rewards prompt-following against a target sampled from a grammar of the form 'A <attribute> <vehicle type> <environment>'. Target prompts are not provided to the methods directly; they are revealed only through the prompt-following score used as the objective.

Setup. We measure prompt-following with PickScore [25]. PickScore is prompt-dependent and on this task occupies a roughly 16–24 range, so absolute values are not directly meaningful; we therefore compare methods within a prompt and average across 10 target prompts. To aid interpretation, scores less than 19 typically correspond to monochrome or noise-like outputs, 20 to coherent images that miss the target, and 23 to images matching two of three target attributes. Small numeric differences in this range can correspond to substantive visual differences; see Figure 19 for examples, as well as Appendix H for a full description of the experimental setup.

We compare Bayesian Optimisation in search spaces formed by REMBO, LOL, and O^3 against Best-of- N sampling from the base model. We evaluate at three search space dimensionalities (2D, 8D, and 89D, corresponding to 3, 9, and 90 seed latents respectively). For each method, the search space is formed either from random seed latents or from seed latents obtained by inverting an image that matches exactly one property of the target prompt. We report filtered-seed results to isolate optimiser performance from seed-acquisition cost; see Section 4.4 for matched-budget comparisons.

Results. Figure 5 shows three main findings. *First*, search spaces that produce samples matching the latent distribution of the model (LOL and O^3) substantially outperform Best-of- N sampling from the base model, while those that do not (REMBO) fail to do so — an example of the best REMBO generation is included in Figure 19. This supports the view that preserving the latent sampling statistics is what enables optimisation to succeed. We further corroborate this with CMA-ES in the native latent space $\mathcal{Z}$ (see Appendix H), whose proposals violate the necessary statistical properties identified by Bodin et al. [5], and produce only black images. *Second*, O^3 matches LOL in low-dimensional settings (2D, 8D) and pulls clearly ahead in the 89D search space. *Third*, inversion-based seeds improve performance for both LOL and O^3 across all settings, though the gain shrinks as the search space grows. Since our benchmark is constructed such that suitable seeds are readily available, this represents a best case for inversion; the diminishing returns at higher dimensions suggest the search-space structure increasingly carries the optimisation, with seed latent quality contributing less.

We report several ablations in the appendices. Appendix H extends the comparison to CMA-ES, both in a search space formed with O^3 and in the native space $\mathcal{Z}$. Appendix J reports results with an alternative choice of ϕ_w , alongside all combinations of optimisers — including random search in surrogate spaces. We find that in very low-dimensional surrogate spaces, random search in $\mathcal{U}$ is nearly as effective as BO and CMA-ES, since these spaces are simple to search but contain limited solutions. As dimensionality increases, BO and CMA-ES exploit the structure of the surrogate space and pull substantially ahead.

4.4 BOLTZ-2 guidance under matched oracle-call budgets

Goal. Protein structure prediction models such as BOLTZ-2 [39] predict 3D atomic structures from amino-acid sequences, but their outputs are not always aligned with task-specific objectives, for example when a particular conformation is preferred among several stable ones. In this experiment, we bias generation toward a specific reference conformation, with TM-score [58], a structural similarity score against a chosen reference, as our oracle.¹ While TM-score is cheap to compute, the matched-budget protocol is intended as a proxy for the expensive, often non-differentiable oracles common in protein design.

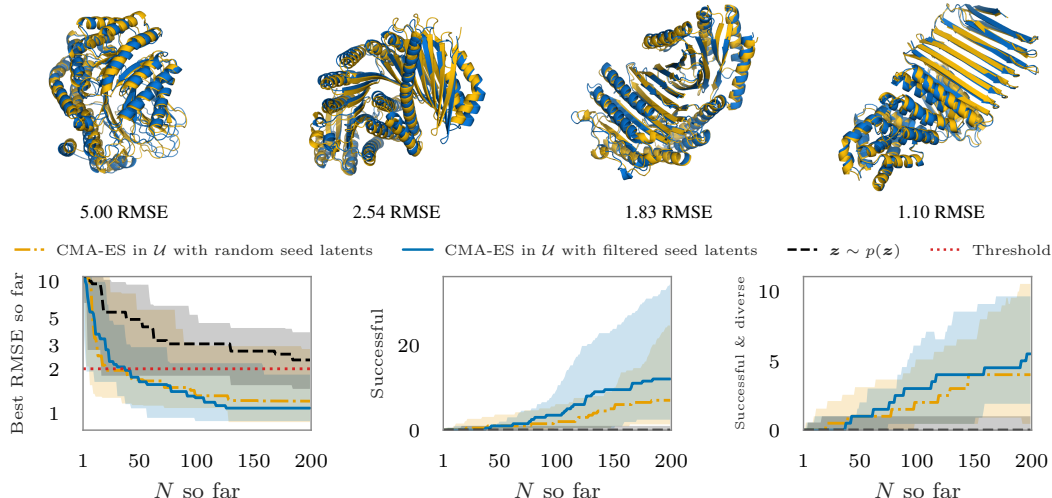


Figure 7: **Protein design with surrogate latent spaces.** *Top:* Representative generations showing the RMSE discrepancy between the RFDIFFUSION backbone (yellow) and their ALPHAFOLD2 regeneration (blue). *Bottom:* comparison of sample-and-filter in $\mathcal{Z}$ (Best of N) versus optimisation with O^3 using CMA-ES. Plots report the median and 90% confidence interval of the best RMSE per step as well as the number of successful and diverse designs.

Setup. Our reference is 1CLL [7], a known conformation of calmodulin. We adapt BOLTZ-2’s stochastic sampler to its corresponding probability-flow ODE so that latents map deterministically to structures, as required by the O^3 methodology. O^3 first samples a pool of m structures from BOLTZ-2, scores them with the oracle, and keeps the top K as seed latents defining the surrogate space $\mathcal{U}$. Bayesian optimisation then runs over $\mathcal{U}$ with the remaining $N - m$ oracle calls. The three baselines we compare against are: Best-of- N , which draws N samples and returns the highest-scoring one; FK-steering [45], which runs Sequential Monte Carlo and reweights particles by oracle scores during sampling for N total oracle calls; and DPO [41, 50], which fine-tunes BOLTZ-2 iteratively on preference pairs formed from N oracle-scored structures. Full implementation details, including the probability-flow ODE conversion, BO setup, baseline configurations, and ablations, are reported in Appendix K.

Results. Figure 6 reports max TM-score against the 1CLL reference across budgets. At low budgets ($N \leq 100$), O^3 outperforms all baselines: at $N = 100$ it achieves ~ 0.68 , against ~ 0.65 for Best-of- N and lower for FK-steering and DPO, both of which underperform O^3 at these budgets and also fail to beat Best-of- N . At intermediate budgets, FK-steering performs best at $N = 200$ and is on par

¹TM-score is in range of $(0, 1]$ with higher values indicating higher structural similarity. See Appendix M for more details.

with DPO at $N = 500$; DPO, however, requires fine-tuning of the base generative model, which is costly. At $N = 1000$, O^3 , FK-steering, and DPO all reach ~ 0.78 , with O^3 operating without any fine-tuning. Across the full budget range, O^3 is competitive with the strongest baseline at every N and outperforms all other methods at low oracle budgets.

4.5 Protein optimisation with RFDIFFUSION

Goal. Finally, we apply our method to a challenging setting in protein design: generating 600-residue proteins with RFDIFFUSION [52]. At this length, sampling directly from the model almost never produces designs meeting the RMSE recoverability criterion of Watson et al. [52]. The objective is expensive and non-differentiable, and the sample-and-filter strategy of Watson et al. [52] remains the only practical baseline.

Setup. We compare sample-and-filter in $\mathcal{Z}$ with CMA-ES in our surrogate space $\mathcal{U}$, using 24 seeds, either random or selected by lowest RMSE from 100 samples. The random-seed condition tests whether gains require informative seeds or follow from the search-space structure alone.

We optimise for RMSE only. The full criterion of Watson et al. [52] also requires $PAE < 5$; we relax this to avoid multi-objective optimisation. Appendix L reports PAE, where O^3 substantially improves over baseline despite not optimising it. Throughout, “successful” means passing the RMSE filter.

Results. Figure 7 shows that after 200 evaluations, sample-and-filter in $\mathcal{Z}$ often fails (median run: RMSE = 2.33, no successful designs). CMA-ES in $\mathcal{U}$ reliably finds successful designs with random seed latents (median run: RMSE = 1.19, 7 successes across 4 clusters), improving further with filtered seed latents (median run: RMSE = 1.08, 12 successes across 5.5 clusters). Substantial gains even from random seed latents indicate that surrogate-space structure is the dominant factor.

O^3 also reduces PAE substantially relative to baseline (Appendix L), though without targeting it directly we do not reach the $PAE < 5$ threshold; multi-objective optimisation is a natural extension. Unlike the matched-budget protocol of Section 4.4, here we do not charge seed-acquisition cost to the optimisation budget in the filtered-seed condition: this isolates the contribution of search-space structure from seed quality, and is often more representative of practice, where seeds are typically available from prior experiments or known designs; when they are not, the random-seed results show the search space alone delivers most of the benefit. Full experimental details are in Appendix L.

5 Conclusion

In this paper, we introduced *surrogate latent spaces*: a simple, general construction for expressive low-dimensional search spaces from high-dimensional generative models. These spaces define structured, deterministic manifolds over model outputs, making optimisation tractable for expensive or gradient-free black-box objectives. Our experiments show that surrogate latent spaces condition generative models through example-defined coordinates, with geometry suited to standard optimisation across modalities. Future work includes predicting seed-latent utility, adaptive seed selection, and human-in-the-loop objectives for creative and scientific applications.

References

- [1] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209): 1–80, 2025.
- [2] Sebastian Ament, Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy. Unexpected improvements to expected improvement for bayesian optimization. *Advances in neural information processing systems*, 36:20577–20612, 2023.
- [3] Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems* 33, 2020. URL <http://arxiv.org/abs/1910.06403>.

- [4] Erik Bodin, Zhenwen Dai, Neill Campbell, and Carl Henrik Ek. Black-box density function estimation using recursive partitioning. In *International Conference on Machine Learning*, pages 1015–1025. PMLR, 2021.
- [5] Erik Bodin, Alexandru Stere, Dragos D Margineantu, Carl Henrik Ek, and Henry Moss. Linear combinations of latents in generative models: subspaces and beyond. *arXiv preprint arXiv:2408.08558*, 2024.
- [6] Onur Boyar and Ichiro Takeuchi. Latent space bayesian optimization with latent data augmentation for enhanced exploration. *Neural Computation*, 36(11):2446–2478, 2024.
- [7] Rajagopal Chattopadhyaya, William E. Meador, Anthony R. Means, and Florante A. Quiocho. Calmodulin structure refined at 1.7 Å resolution. *Journal of Molecular Biology*, 228(4):1177–1192, December 1992. ISSN 0022-2836. URL [http://dx.doi.org/10.1016/0022-2836\(92\)90324-D](http://dx.doi.org/10.1016/0022-2836(92)90324-D).
- [8] Jaewon Chu, Jinyoung Park, Seunghun Lee, and Hyunwoo J Kim. Inversion-based latent bayesian optimization. *arXiv preprint arXiv:2411.05330*, 2024.
- [9] J Dauparas, I Anishchenko, N Bennett, H Bai, R J Ragotte, L F Milles, B I M Wicky, A Courbet, R J de Haas, N Bethel, P J Y Leung, T F Huddy, S Pellock, D Tischer, F Chan, B Koepnick, H Nguyen, A Kang, B Sankaran, A K Bera, N P King, and D Baker. Robust deep learning-based protein sequence design using ProteinMPNN. *Science*, 378(6615):49–56, October 2022.
- [10] Alexander Denker, Shreyas Padhy, Francisco Vargas, and Johannes Hertrich. Iterative importance fine-tuning of diffusion models. In *Frontiers in Probabilistic Inference: Learning meets Sampling*, 2025. URL <https://openreview.net/forum?id=HLaFozI6It>.
- [11] Carles Domingo-Enrich, Michal Drozdal, Brian Karrer, and Ricky TQ Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024.
- [12] David Eriksson and Martin Jankowiak. High-dimensional bayesian optimization with sparse axis-aligned subspaces. In *Uncertainty in Artificial Intelligence*, pages 493–503. PMLR, 2021.
- [13] David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. *Advances in neural information processing systems*, 32, 2019.
- [14] Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio open. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025.
- [15] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Reinforcement learning for fine-tuning text-to-image diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS) 2023*. Neural Information Processing Systems Foundation, 2023.
- [16] Carl Friedrich Gauss. Disquisitiones generales circa superficies curvas. *Commentationes Societatis Regiae Scientiarum Gottingensis Recentiores*, 1827.
- [17] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.
- [18] Antoine Grosnit, Rasul Tutunov, Alexandre Max Maraval, Ryan-Rhys Griffiths, Alexander I Cowen-Rivers, Lin Yang, Lin Zhu, Wenlong Lyu, Zhitang Chen, Jun Wang, et al. High-dimensional bayesian optimisation with variational autoencoders and deep metric learning. *arXiv preprint arXiv:2106.03609*, 2021.
- [19] Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36:12489–12517, 2023.

- [20] Nikolaus Hansen. The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016.
- [21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [22] Seongmin Hong, Kyeonghyun Lee, Suh Yoon Jeon, Hyewon Bae, and Se Young Chun. On exact inversion of dpm-solvers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7069–7078, 2024.
- [23] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the Design Space of Diffusion-Based Generative Models, 2022. URL <https://arxiv.org/abs/2206.00364>.
- [24] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *Proceedings of the International Conference on Learning Representations*, 2014.
- [25] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.
- [26] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.
- [27] Siddharth Krishnamoorthy, Satvik Mehul Mashkaria, and Aditya Grover. Diffusion models for black-box optimization. In *International Conference on Machine Learning*, pages 17842–17857. PMLR, 2023.
- [28] Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *International conference on machine learning*, pages 1945–1954. PMLR, 2017.
- [29] Black Forest Labs. Flux. <https://github.com/black-forest-labs/flux>, 2024.
- [30] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.
- [31] Xiaoyu Lu, Javier Gonzalez, Zhenwen Dai, and Neil Lawrence. Structured variationally auto-encoded optimization. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3267–3275. PMLR, 10–15 Jul 2018. URL <https://proceedings.mlr.press/v80/lu18c.html>.
- [32] Renqian Luo, Fei Tian, Tao Qin, Enhong Chen, and Tie-Yan Liu. Neural architecture optimization. *Advances in neural information processing systems*, 31, 2018.
- [33] Natalie Maus, Haydn Jones, Juston Moore, Matt J Kusner, John Bradshaw, and Jacob Gardner. Local latent space bayesian optimization over structured inputs. *Advances in neural information processing systems*, 35:34505–34518, 2022.
- [34] Henry Moss, Sebastian W Ober, and Tom Diethe. Return of the latent space cowboys: Rethinking the use of vaes for bayesian optimisation of structured spaces. In *International Conference on Machine Learning*, 2025.
- [35] Lam Ngo, Huong Ha, Jeffrey Chan, and Hongyu Zhang. Boids: High-dimensional bayesian optimization via incumbent-guided direction lines and subspace embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19659–19667, 2025.
- [36] Masahiro Nomura and Masashi Shibata. cmaes: A simple yet practical python library for cma-es. *arXiv preprint arXiv:2402.01373*, 2024.
- [37] Rafael Oliveira, Daniel M Steinberg, and Edwin V Bonilla. Generative bayesian optimization: Generative models as acquisition functions. *arXiv preprint arXiv:2510.25240*, 2025.

- [38] Marina A Pak, Karina A Markhieva, Mariia S Novikova, Dmitry S Petrov, Ilya S Vorobyev, Ekaterina S Maksimova, Fyodor A Kondrashov, and Dmitry N Ivankov. Using AlphaFold to predict the impact of single mutations on protein stability and function. *PLoS One*, 18(3): e0282689, March 2023.
- [39] Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, David Kwabi-Addo, Dominique Beaini, Tommi Jaakkola, and Regina Barzilay. Boltz-2: Towards Accurate and Efficient Binding Affinity Prediction. June 2025. URL <http://dx.doi.org/10.1101/2025.06.14.659707>.
- [40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.
- [41] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model, 2023. URL <https://arxiv.org/abs/2305.18290>.
- [42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.
- [43] Dvir Samuel, Barak Meiri, Haggai Maron, Yoad Tewel, Nir Darshan, Shai Avidan, Gal Chechik, and Rami Ben-Ari. Lightning-fast image inversion and editing for text-to-image diffusion models. *arXiv preprint arXiv:2312.12540*, 2023.
- [44] 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. *Proceedings of the IEEE*, 104(1): 148–175, 2015.
- [45] Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A General Framework for Inference-time Scaling and Steering of Diffusion Models, 2025. URL <https://arxiv.org/abs/2501.06848>.
- [46] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.
- [47] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.
- [48] Daniel M Steinberg, Rafael Oliveira, Cheng Soon Ong, and Edwin V Bonilla. Variational search distributions. *arXiv preprint arXiv:2409.06142*, 2024.
- [49] Austin Tripp, Erik Daxberger, and José Miguel Hernández-Lobato. Sample-efficient optimization in the latent space of deep generative models via weighted retraining. *Advances in Neural Information Processing Systems*, 33:11259–11272, 2020.
- [50] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion Model Alignment Using Direct Preference Optimization, 2023. URL <https://arxiv.org/abs/2311.12908>.
- [51] Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Freitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.
- [52] Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Nikita Hanikel, Samuel J. Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S. Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, July 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06415-8. URL <http://dx.doi.org/10.1038/s41586-023-06415-8>.

- [53] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.
- [54] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.
- [55] Yinghua Yao, Yuangang Pan, Jing Li, Ivor Tsang, and Xin Yao. Proud: Pareto-guided diffusion model for multi-objective generation. *Machine Learning*, 113(9):6511–6538, 2024.
- [56] Ye Yuan, Can Chen, Christopher Pal, and Xue Liu. Paretoflow: Guided flows in multi-objective optimization. *arXiv preprint arXiv:2412.03718*, 2024.
- [57] Taeyoung Yun, Kiyoun Om, Jaewoo Lee, Sujin Yun, and Jinkyoo Park. Posterior inference with diffusion models for high-dimensional black-box optimization. *arXiv preprint arXiv:2502.16824*, 2025.
- [58] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–710, December 2004.

Impact Statement

O^3 is a general, model-agnostic optimisation layer that operates on top of pretrained continuous-variable diffusion and flow-matching models without modifying or fine-tuning them. Its societal impact is therefore largely inherited from the underlying generative models and from the objective functions that practitioners choose to optimise.

On the positive side, by enabling sample-efficient black-box optimisation without retraining, O^3 can lower the compute and data costs of steering generative models toward task-specific objectives. This is particularly relevant in scientific and engineering applications such as protein design, where evaluation is expensive and gradients are typically unavailable, and where reducing the number of oracle calls translates directly into reduced experimental cost. More broadly, decoupling search from model training broadens access to controllable generation for users without the resources to fine-tune large generative models.

On the negative side, any method that improves the controllability of generative models can in principle be misused to more reliably elicit harmful outputs, for example to amplify biases, generate disinformation or non-consensual content from image, audio, or video models, or to design biological sequences with harmful function. O^3 does not introduce new generative capabilities beyond those already present in the base model, and does not bypass safety filters or alignment mechanisms built into those models; the appropriate locus for safeguards therefore remains the base model and the choice of objective. We encourage practitioners to apply O^3 only on top of generative models whose intended use and safety properties are appropriate for the task, and, in high-stakes domains such as biological design, to pair it with task-specific review and screening procedures.

A Non-Gaussian latents and bijectivity of the surrogate chart

The construction in Section 3 was presented for standard Gaussian latents, $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, for which unit-norm linear combinations of independent latent samples remain distributed according to the same law. This property makes it possible to form valid proposals directly by combining seed latents with weights on the unit sphere. In this appendix, we extend the construction to latent distributions beyond the standard Gaussian case. The central requirement is not Gaussianity itself, but the existence of an “inner” latent representation in which unit- ℓ_2 linear combinations can be transformed back into valid samples from the target latent distribution. When the original latent distribution does not satisfy this property directly, we introduce transport maps to and from such an inner representation, perform the linear combination there, and map the result back to the original latent space. This recovers the general Latent Optimal Linear combinations (LOL) construction of Bodin et al. [5], which we further extend with additional transport maps to broaden the class of supported latent distributions. In the Gaussian setting used throughout the main text, these transports reduce to the identity.

We also show that the resulting chart is bijective *on its image*: every latent generated by the chart has a unique coordinate $\mathbf{u} \in \mathcal{U}$, and this coordinate can be recovered exactly. Equivalently, the surrogate space introduces no redundant parametrisations within the transported span induced by the chosen seed latents: distinct surrogate coordinates map to distinct latent proposals, and any proposal generated by the chart admits an exact inverse back to $\mathcal{U}$.

The key idea behind our method is to construct a search space $\mathcal{U}$ from a collection of K seed latents, yielding a coordinate system whose generated objects inherit properties from these seeds. Let $\{\mathbf{z}_k\}_{k=1}^K$, with $\mathbf{z}_k \in \mathcal{Z}$, denote the seed latents. We define a surrogate chart

$$\phi(\cdot; \{\mathbf{z}_k\}_{k=1}^K) : \mathcal{U} \rightarrow \mathcal{Z}$$

by

$$\phi(\mathbf{u}; \{\mathbf{z}_k\}_{k=1}^K) = \mathbf{z}, \quad \mathbf{z} := \ell(\mathbf{w}; \{\mathbf{z}_k\}_{k=1}^K), \quad \mathbf{w} := \phi_w(\mathbf{u}), \quad (9)$$

where $\phi_w : \mathcal{U} \rightarrow \mathbb{S}_+^{K-1}$ is a bijection from the surrogate space to the positive orthant of the unit sphere, and

$$\ell(\cdot; \{\mathbf{z}_k\}_{k=1}^K) : \mathbb{S}_+^{K-1} \rightarrow \mathcal{Z}$$

maps spherical weights to latent variables. Here,

$$\mathbb{S}_+^{K-1} := \left\{ \mathbf{w} \in \mathbb{R}^K \mid \|\mathbf{w}\|_2 = 1, w_i \geq 0 \ \forall i \right\}.$$

On its image, the chart admits the inverse

$$\phi^{-1}(z; \{z_k\}_{k=1}^K) = \mathbf{u}, \quad \mathbf{u} := \phi_w^{-1}(\mathbf{w}), \quad \mathbf{w} := \ell^{-1}(z; \{z_k\}_{k=1}^K). \quad (10)$$

To ensure validity, each coordinate $\mathbf{u} \in \mathcal{U}$ must map to a latent realisation $\mathbf{z} \in \mathcal{Z}$ with the model's latent distribution p . We achieve this by defining ℓ using a transported aggregation. Let $\mathcal{T}_\rightarrow$ map model latents into an inner representation, and let $\mathcal{T}_\leftarrow$ map aggregated inner vectors back to valid samples from p . Given seed latents $\{z_k\}_{k=1}^K$, define

$$\mathbf{z} = \ell(\mathbf{w}; \{z_k\}_{k=1}^K) := \mathcal{T}_\leftarrow(\boldsymbol{\epsilon}), \quad \boldsymbol{\epsilon} := \boldsymbol{\xi}\mathbf{w}, \quad \boldsymbol{\xi} := [\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_K], \quad \boldsymbol{\epsilon}_k := \mathcal{T}_\rightarrow(z_k). \quad (11)$$

The required validity condition is that, for independent seed latents $z_k \sim p$ and any $\mathbf{w} \in \mathbb{S}^{K-1}$,

$$\mathcal{T}_\leftarrow(\boldsymbol{\xi}\mathbf{w}) \sim p.$$

In the Gaussian case, this holds because the inner aggregate itself has the target Gaussian law. In the hyperspherical case, the aggregate is not uniformly distributed on the sphere, but its normalised direction is.

For points generated by the forward map $\mathbf{u} \mapsto \mathbf{w} \mapsto \mathbf{z}$, the inverse can be computed exactly on the image of the chart under an additional directional-consistency condition on the transports. In particular, if $\mathcal{T}_\rightarrow(\mathcal{T}_\leftarrow(\boldsymbol{\epsilon}))$ is a positive scalar multiple of $\boldsymbol{\epsilon}$ and the seed matrix $\boldsymbol{\xi}$ has full column rank, then

$$\mathbf{w} = \ell^{-1}(z; \{z_k\}_{k=1}^K) = \frac{\boldsymbol{\xi}^+ \mathcal{T}_\rightarrow(z)}{\|\boldsymbol{\xi}^+ \mathcal{T}_\rightarrow(z)\|_2},$$

where $\boldsymbol{\xi}^+$ denotes the Moore–Penrose inverse of $\boldsymbol{\xi}$; see Appendix B. Typically $\dim(\boldsymbol{\epsilon}) = \dim(\mathbf{z})$, although this depends on the chosen transport map.

Inner aggregations preserving the target distribution. An inner representation is amenable to our methodology if unit- ℓ_2 aggregation of independent transformed seed latents, followed by the backward transport, preserves the target latent distribution. Formally, if $z_1, \dots, z_K \stackrel{\text{i.i.d.}}{\sim} p$, $\boldsymbol{\epsilon}_k = \mathcal{T}_\rightarrow(z_k)$, and $\mathbf{w} \in \mathbb{S}^{K-1}$, then the requirement is

$$\mathcal{T}_\leftarrow\left(\sum_{k=1}^K w_k \boldsymbol{\epsilon}_k\right) \sim p.$$

Equivalently, with $\boldsymbol{\xi} = [\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_K]$ and $\boldsymbol{\epsilon} = \boldsymbol{\xi}\mathbf{w}$, we require

$$\mathbf{z} = \mathcal{T}_\leftarrow(\boldsymbol{\epsilon}) \sim p.$$

In some cases, such as zero-mean Gaussian inner latents, the aggregate $\boldsymbol{\epsilon}$ itself has a fixed inner distribution. In other cases, such as hyperspherical latents, $\boldsymbol{\epsilon}$ need only have a distribution whose image under $\mathcal{T}_\leftarrow$ is the desired target distribution. Specifying the maps $\mathcal{T}_\rightarrow$ and $\mathcal{T}_\leftarrow$ therefore gives a general framework for applying our method to different latent distributions:

- **Gaussian latents.** If $p = \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, then closure holds directly under $\|\mathbf{w}\|_2 = 1$, and we may set $\mathcal{T}_\rightarrow = \mathcal{T}_\leftarrow = \text{id}$. If $p = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with $\boldsymbol{\mu} \neq \mathbf{0}$, then

$$\mathcal{T}_\rightarrow(\mathbf{z}) = \mathbf{z} - \boldsymbol{\mu}, \quad \mathcal{T}_\leftarrow(\boldsymbol{\epsilon}) = \boldsymbol{\epsilon} + \boldsymbol{\mu},$$

which centre the distribution before aggregation and restore the mean afterwards. This case was treated in Bodin et al. [5].

- **Hyperspherical latents.** If $p = \text{Unif}(\mathbb{S}^{D-1})$, where $\mathbf{z} \in \mathbb{R}^D$, then we may set $\mathcal{T}_\rightarrow = \text{id}$ and let $\mathcal{T}_\leftarrow$ normalise the aggregated vector back onto the sphere,

$$\mathcal{T}_\leftarrow(\boldsymbol{\epsilon}) = \frac{\boldsymbol{\epsilon}}{\|\boldsymbol{\epsilon}\|_2}.$$

The aggregate $\boldsymbol{\xi}\mathbf{w}$ is not itself uniformly distributed on the sphere. However, for any fixed unit-norm $\mathbf{w}$, its distribution is rotation invariant, and therefore its normalised direction is uniformly distributed on $\mathbb{S}^{D-1}$.

- **Composite latents.** If the latent variable decomposes into M statistically independent components, $\mathbf{z} = \{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(M)}\}$, with

$$p(\mathbf{z}) = \prod_{m=1}^M p_m(\mathbf{z}^{(m)}),$$

then each component can be transported separately. This allows models with multiple latent variables, or latents with heterogeneous distributions, to be handled by applying component-wise transports and concatenating the resulting inner representations before computing linear combinations.

- **General scalar distributions.** If the dimensions of $\mathbf{z}$ are statistically independent scalar variables, each dimension can be transported to an amenable scalar distribution, such as $\mathcal{N}(0, 1)$, using the probability integral transform. For example, if F_i is the cumulative distribution function of z_i and Φ is the standard Gaussian cumulative distribution function, one may use

$$\epsilon_i = \Phi^{-1}(F_i(z_i)).$$

This corresponds to the scalar transport construction proposed in Bodin et al. [5].

B Inverse of the ℓ map

In this section, we derive the inverse of the map ℓ introduced in Section A. The inverse is defined on the image of ℓ , i.e. for latents generated by the forward map from a fixed set of seed latents.

Lemma B.1. *Let $\{\mathbf{z}_k\}_{k=1}^K \subset \mathcal{Z}$ be seed latents, and define their corresponding inner latents by*

$$\boldsymbol{\epsilon}_k = \mathcal{T}_{\rightarrow}(\mathbf{z}_k).$$

Let

$$\boldsymbol{\xi} = [\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_K] \in \mathbb{R}^{D \times K}.$$

Assume:

(A1) $\boldsymbol{\xi}$ has full column rank, so that $\boldsymbol{\xi}^+ \boldsymbol{\xi} = I_K$;

(A2) for all relevant $\boldsymbol{\epsilon} \in \mathbb{R}^D$,

$$\mathcal{T}_{\rightarrow}(\mathcal{T}_{\leftarrow}(\boldsymbol{\epsilon})) = \alpha(\boldsymbol{\epsilon}) \boldsymbol{\epsilon}, \quad \alpha(\boldsymbol{\epsilon}) > 0.$$

Define

$$\ell(\mathbf{w}; \{\mathbf{z}_k\}_{k=1}^K) = \mathcal{T}_{\leftarrow}(\boldsymbol{\xi} \mathbf{w}), \quad \mathbf{w} \in \mathbb{S}_+^{K-1}.$$

Then ℓ is injective on $\mathbb{S}_+^{K-1}$ and admits the following inverse on its image:

$$\ell^{-1}(\mathbf{z}; \{\mathbf{z}_k\}_{k=1}^K) = \frac{\boldsymbol{\xi}^+ \mathcal{T}_{\rightarrow}(\mathbf{z})}{\|\boldsymbol{\xi}^+ \mathcal{T}_{\rightarrow}(\mathbf{z})\|_2}.$$

Proof. Let

$$\mathbf{z} = \ell(\mathbf{w}; \{\mathbf{z}_k\}_{k=1}^K) = \mathcal{T}_{\leftarrow}(\boldsymbol{\xi} \mathbf{w})$$

for some $\mathbf{w} \in \mathbb{S}_+^{K-1}$. Applying $\mathcal{T}_{\rightarrow}$ and using (A2) gives

$$\mathcal{T}_{\rightarrow}(\mathbf{z}) = \mathcal{T}_{\rightarrow}(\mathcal{T}_{\leftarrow}(\boldsymbol{\xi} \mathbf{w})) = \alpha(\boldsymbol{\xi} \mathbf{w}) \boldsymbol{\xi} \mathbf{w}.$$

Multiplying by $\boldsymbol{\xi}^+$ and using (A1), we obtain

$$\boldsymbol{\xi}^+ \mathcal{T}_{\rightarrow}(\mathbf{z}) = \alpha(\boldsymbol{\xi} \mathbf{w}) (\boldsymbol{\xi}^+ \boldsymbol{\xi}) \mathbf{w} = \alpha(\boldsymbol{\xi} \mathbf{w}) \mathbf{w}.$$

Thus $\boldsymbol{\xi}^+ \mathcal{T}_{\rightarrow}(\mathbf{z})$ is a positive scalar multiple of $\mathbf{w}$. Since $\|\mathbf{w}\|_2 = 1$, normalising removes the unknown positive scale factor:

$$\frac{\boldsymbol{\xi}^+ \mathcal{T}_{\rightarrow}(\mathbf{z})}{\|\boldsymbol{\xi}^+ \mathcal{T}_{\rightarrow}(\mathbf{z})\|_2} = \mathbf{w}.$$

Therefore the stated expression recovers the unique weight vector $\mathbf{w}$ that generated $\mathbf{z}$, establishing injectivity of ℓ on $\mathbb{S}_+^{K-1}$ and giving the inverse on the image of ℓ . $\square$

Corollary B.2 (When normalisation is redundant). *For generated points $\mathbf{z} = \ell(\mathbf{w}; \{\mathbf{z}_k\}_{k=1}^K)$, the normalisation in the inverse formula is redundant whenever*

$$\mathcal{T}_{\rightarrow} \circ \mathcal{T}_{\leftarrow} = \text{id}$$

on the relevant inner latent subspace. In this case, $\alpha(\epsilon) \equiv 1$, and

$$\ell^{-1}(\mathbf{z}; \{\mathbf{z}_k\}_{k=1}^K) = \xi^+ \mathcal{T}_{\rightarrow}(\mathbf{z}).$$

- **Gaussian latents.** *The maps $\mathcal{T}_{\rightarrow}$ and $\mathcal{T}_{\leftarrow}$ are exact inverses, so $\alpha(\epsilon) = 1$ and normalisation is not required.*
- **Hyperspherical latents.** *With $\mathcal{T}_{\leftarrow}(\epsilon) = \epsilon / \|\epsilon\|_2$ and $\mathcal{T}_{\rightarrow} = \text{id}$,*

$$\mathcal{T}_{\rightarrow}(\mathcal{T}_{\leftarrow}(\epsilon)) = \frac{\epsilon}{\|\epsilon\|_2},$$

so $\alpha(\epsilon) = 1 / \|\epsilon\|_2$. Normalisation is therefore required to recover $\mathbf{w}$.

- **Independent scalar latents mapped via CDF transports.** *When the scalar CDF transports and their inverses are exact, $\mathcal{T}_{\rightarrow}$ and $\mathcal{T}_{\leftarrow}$ are exact inverses, so $\alpha(\epsilon) = 1$ and normalisation is redundant.*

C The unit hypersphere is a sufficient index set for Latent Optimal Linear combinations

In this section, we show that Latent Optimal Linear combinations [5] (LOL) can be indexed by weights on the unit hypersphere. The essential point is that valid transported aggregation requires only a unit-norm direction among the seed latents. In the zero-mean Gaussian case, this can also be viewed as removing radial redundancy in an arbitrary nonzero weight vector: scaling all weights by a positive constant does not change the corrected latent. We first describe this in the Gaussian case and then extend the argument to the transported, non-Gaussian setting.

Gaussian latents. Let

$$\mathbf{y} = \mathbf{Z}\mathbf{a}, \quad \mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_K], \quad \mathbf{a} \in \mathbb{R}^K,$$

where the seed latents are independent draws $\mathbf{z}_k \sim p$ with

$$p = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}).$$

Then

$$\mathbf{y} \sim \mathcal{N}(\alpha\boldsymbol{\mu}, \beta\boldsymbol{\Sigma}), \quad \alpha = \sum_{k=1}^K a_k, \quad \beta = \sum_{k=1}^K a_k^2 = \|\mathbf{a}\|_2^2.$$

Unless $\alpha = 1$ (or $\boldsymbol{\mu} = \mathbf{0}$) and $\beta = 1$, the linear combination $\mathbf{y}$ is not guaranteed to follow the target latent distribution p . In the Gaussian case, Latent Optimal Linear combinations correct this by applying the Monge optimal transport map from $\mathcal{N}(\alpha\boldsymbol{\mu}, \beta\boldsymbol{\Sigma})$ to $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$,

$$\mathcal{T}(\mathbf{y}) = \boldsymbol{\mu} + \frac{\mathbf{y} - \alpha\boldsymbol{\mu}}{\sqrt{\beta}} = \left(1 - \frac{\alpha}{\|\mathbf{a}\|_2}\right)\boldsymbol{\mu} + \frac{\mathbf{y}}{\|\mathbf{a}\|_2}.$$

For zero-mean Gaussian latents, this simplifies to

$$\mathcal{T}(\mathbf{y}) = \frac{\mathbf{Z}\mathbf{a}}{\|\mathbf{a}\|_2}.$$

Thus, for any nonzero $\mathbf{a} \in \mathbb{R}^K$, defining

$$\mathbf{w} = \frac{\mathbf{a}}{\|\mathbf{a}\|_2} \in \mathbb{S}^{K-1}$$

gives

$$\mathcal{T}(\mathbf{Z}\mathbf{a}) = \mathbf{Z}\mathbf{w}.$$

Therefore the norm of $\mathbf{a}$ is redundant in the zero-mean Gaussian case: all positive rescalings of $\mathbf{a}$ yield the same corrected latent. Consequently, weights on the unit hypersphere $\mathbb{S}^{K-1}$ are sufficient to index all zero-mean Gaussian Latent Optimal Linear combinations.

For nonzero-mean Gaussian latents, the same conclusion holds after centring. Writing

$$\boldsymbol{\epsilon}_k = \mathbf{z}_k - \boldsymbol{\mu},$$

we form the centred combination

$$\boldsymbol{\epsilon} = \sum_{k=1}^K w_k \boldsymbol{\epsilon}_k, \quad \mathbf{w} \in \mathbb{S}^{K-1},$$

and map back via

$$\mathbf{z} = \boldsymbol{\epsilon} + \boldsymbol{\mu}.$$

Since $\|\mathbf{w}\|_2 = 1$, the centred combination satisfies $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, and therefore $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Thus, centring removes the dependence on the mean, and unit-norm weights again provide a sufficient index set.

General transported latents. The same principle applies in the non-Gaussian setting whenever the latent distribution admits an inner representation whose unit- ℓ_2 aggregations can be mapped back to valid samples from the target distribution. Let $\mathcal{T}_\rightarrow$ map model latents into an inner representation, and let $\mathcal{T}_\leftarrow$ map aggregated inner vectors back to samples from p . Given seed latents $\{\mathbf{z}_k\}_{k=1}^K$, define

$$\boldsymbol{\epsilon}_k = \mathcal{T}_\rightarrow(\mathbf{z}_k), \quad \boldsymbol{\xi} = [\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_K].$$

The required condition is that, for independent seed latents $\mathbf{z}_k \sim p$ and any

$$\mathbf{w} \in \mathbb{S}^{K-1}, \quad \boldsymbol{\epsilon} = \boldsymbol{\xi} \mathbf{w},$$

the transported aggregate satisfies

$$\mathbf{z} = \mathcal{T}_\leftarrow(\boldsymbol{\epsilon}) \sim p.$$

Thus, the unit hypersphere provides a sufficient index set for valid Latent Optimal Linear combinations: the weights specify a unit-norm aggregation direction among the transformed seed latents, while the backward transport ensures that the resulting latent has the correct target distribution.

Summary. Both the Gaussian and transported non-Gaussian constructions rely on the same mechanism: aggregation is performed in an inner representation using unit- ℓ_2 weights, and the resulting aggregate is mapped back to the target latent distribution. Therefore, valid Latent Optimal Linear combinations can be indexed by weights on the unit hypersphere $\mathbb{S}^{K-1}$. In the zero-mean Gaussian case, this additionally removes the radial redundancy of arbitrary nonzero weights in $\mathbb{R}^K$, since the corrected latent depends only on the normalised direction of the weight vector.

D Weight charts ϕ_w

Let K denote the number of seed latents. A *weight chart* is a map

$$\phi_w : [0, 1]^{K-1} \rightarrow \mathbb{S}_+^{K-1},$$

where

$$\mathbb{S}_+^{K-1} = \{\mathbf{w} \in \mathbb{R}^K \mid \|\mathbf{w}\|_2 = 1, w_i \geq 0 \ \forall i\}$$

is the positive orthant of the unit hypersphere. The charts below define smooth bijections from the open cube $(0, 1)^{K-1}$ to the interior of $\mathbb{S}_+^{K-1}$, and extend continuously to the boundary.

D.1 Angular-coordinate chart

The standard spherical-coordinate parametrisation gives a simple chart for $\mathbb{S}_+^{K-1}$. For $\mathbf{u} \in (0, 1)^{K-1}$, set

$$\theta_i = \frac{\pi}{2} u_i, \quad i = 1, \dots, K-1,$$

and define

$$w_1 = \cos \theta_1, \quad (12)$$

$$w_k = \left(\prod_{i=1}^{k-1} \sin \theta_i \right) \cos \theta_k, \quad k = 2, \dots, K-1, \quad (13)$$

$$w_K = \prod_{i=1}^{K-1} \sin \theta_i. \quad (14)$$

This gives $\mathbf{w} \in \mathbb{S}_+^{K-1}$.

The inverse is obtained recursively. First,

$$\theta_1 = \arccos(w_1), \quad u_1 = \frac{2}{\pi} \theta_1.$$

For $k = 2, \dots, K-1$,

$$\theta_k = \arccos\left(\frac{w_k}{\prod_{i=1}^{k-1} \sin \theta_i}\right), \quad u_k = \frac{2}{\pi} \theta_k.$$

This chart is simple and smooth on the interior, but it is not measure-preserving: if $\mathbf{u} \sim \text{Unif}([0, 1]^{K-1})$, the induced distribution on $\mathbb{S}_+^{K-1}$ is not the uniform surface measure. The inverse can also be numerically unstable near the boundary, where some factors $\sin \theta_i$ are close to zero.

D.2 Knothe–Rosenblatt stick-breaking chart

We next define a measure-preserving chart based on the Knothe–Rosenblatt rearrangement for the Dirichlet distribution. Let $\mathbf{u} \in (0, 1)^{K-1}$ and define independent stick-breaking variables

$$v_k = I_{u_k}^{-1}\left(\frac{1}{2}, \frac{K-k}{2}\right), \quad k = 1, \dots, K-1,$$

where $I_u^{-1}(a, b)$ denotes the inverse regularised incomplete beta function. Define

$$q_1 = v_1, \quad (15)$$

$$q_k = v_k \prod_{i=1}^{k-1} (1 - v_i), \quad k = 2, \dots, K-1, \quad (16)$$

$$q_K = \prod_{i=1}^{K-1} (1 - v_i), \quad (17)$$

and set

$$w_i = \sqrt{q_i}, \quad i = 1, \dots, K.$$

Since $q_i \geq 0$ and $\sum_{i=1}^K q_i = 1$, this gives $\mathbf{w} \in \mathbb{S}_+^{K-1}$.

The inverse is obtained by reversing the stick-breaking transform. Let

$$q_i = w_i^2, \quad s_k = \sum_{j=k}^K q_j.$$

Then, for $k = 1, \dots, K-1$,

$$v_k = \frac{q_k}{s_k}, \quad u_k = I_{v_k}\left(\frac{1}{2}, \frac{K-k}{2}\right),$$

where $I_v(a, b)$ is the regularised incomplete beta function.

This chart pushes $\text{Unif}([0, 1]^{K-1})$ forward to the uniform surface measure on $\mathbb{S}_+^{K-1}$. In this measure-preserving sense, it is equal-area. In practice, stable implementations of `betainc` and `betaincinv` should be used, and inputs may be clipped away from exactly 0 and 1 to avoid numerical singularities at the boundary.

Uniformity of the Knothe–Rosenblatt chart. Let $\mathbf{u} \sim \text{Unif}((0, 1)^{K-1})$. By the inverse-CDF construction,

$$v_k = I_{u_k}^{-1}\left(\frac{1}{2}, \frac{K-k}{2}\right) \implies v_k \sim \text{Beta}\left(\frac{1}{2}, \frac{K-k}{2}\right),$$

independently for $k = 1, \dots, K-1$. The stick-breaking transform then gives

$$\mathbf{q} = (q_1, \dots, q_K) \sim \text{Dirichlet}\left(\frac{1}{2}, \dots, \frac{1}{2}\right).$$

For a point $\mathbf{w}$ uniformly distributed on $\mathbb{S}_+^{K-1}$, the squared coordinates satisfy

$$\mathbf{w} \odot \mathbf{w} \sim \text{Dirichlet}\left(\frac{1}{2}, \dots, \frac{1}{2}\right).$$

Conversely, if

$$\mathbf{q} \sim \text{Dirichlet}\left(\frac{1}{2}, \dots, \frac{1}{2}\right)$$

and $w_i = \sqrt{q_i}$, then $\mathbf{w}$ is uniformly distributed on $\mathbb{S}_+^{K-1}$. Therefore, the Knothe–Rosenblatt stick-breaking chart pushes the uniform distribution on $[0, 1]^{K-1}$ to the uniform surface measure on the positive orthant of the hypersphere.

E The weight chart ϕ_w sets the similarity structure

In Figure 8 we demonstrate numerical evidence for the claim in Section 3 that the dot product $\mathbf{w}_i^\top \mathbf{w}_j$ is the dominant factor in determining the cosine similarity between two latent variables $\epsilon_i, \epsilon_j \in \mathbb{R}^D$ indexed by a surrogate latent space $\mathcal{U}$. We see that already $D \approx 100, K \leq 10$ yields a dominating $\mathbf{w}_i^\top \mathbf{w}_j$, as shown by Pearson correlations of more than 0.95, and correlations very close to 1 for higher dimensionalities of D (at a rate dependent on K). For reference, typical diffusion and flow matching models [42, 29, 30] have a D of *tens of thousands to hundreds of thousands*, and [26] has a dimensionality of several *million*.

F Empirical stationarity assessments of ϕ_w

In Section 3, we discuss the notion of stationarity and explain why, in practice, it can only be achieved approximately. In our setting, the relevant similarity structure is induced by the weight chart ϕ_w , which maps surrogate coordinates $\mathbf{u} \in \mathcal{U}$ to weights $\mathbf{w} \in \mathbb{S}_+^{K-1}$. We would therefore like the chosen chart to preserve, as far as possible, a simple relationship between Euclidean distances in $\mathcal{U}$ and similarities between the corresponding weights. In this section, we make this notion precise and empirically evaluate the candidate charts introduced in Section D. These candidates are not intended to be exhaustive; rather, our goal is to identify a sufficiently well-behaved chart for use in our methodology.

Define the chart-induced similarity kernel

$$k_\phi(\mathbf{u}, \mathbf{u}') = \phi_w(\mathbf{u})^\top \phi_w(\mathbf{u}').$$

A stationary, or translation-invariant, kernel depends only on the displacement between its inputs,

$$k_\phi(\mathbf{u}, \mathbf{u}') = \kappa(\mathbf{u} - \mathbf{u}').$$

In this work, we consider the stronger isotropic form, where the dependence is only through Euclidean distance:

$$k_\phi(\mathbf{u}, \mathbf{u}') = v(\|\mathbf{u} - \mathbf{u}'\|_2).$$

Equivalently, for any two pairs of points $(\mathbf{u}_1, \mathbf{u}_2)$ and $(\mathbf{u}_3, \mathbf{u}_4)$ in $\mathcal{U}$ with the same Euclidean distance,

$$\|\mathbf{u}_1 - \mathbf{u}_2\|_2 = \|\mathbf{u}_3 - \mathbf{u}_4\|_2,$$

isotropic stationarity would require

$$\phi_w(\mathbf{u}_1)^\top \phi_w(\mathbf{u}_2) = \phi_w(\mathbf{u}_3)^\top \phi_w(\mathbf{u}_4).$$

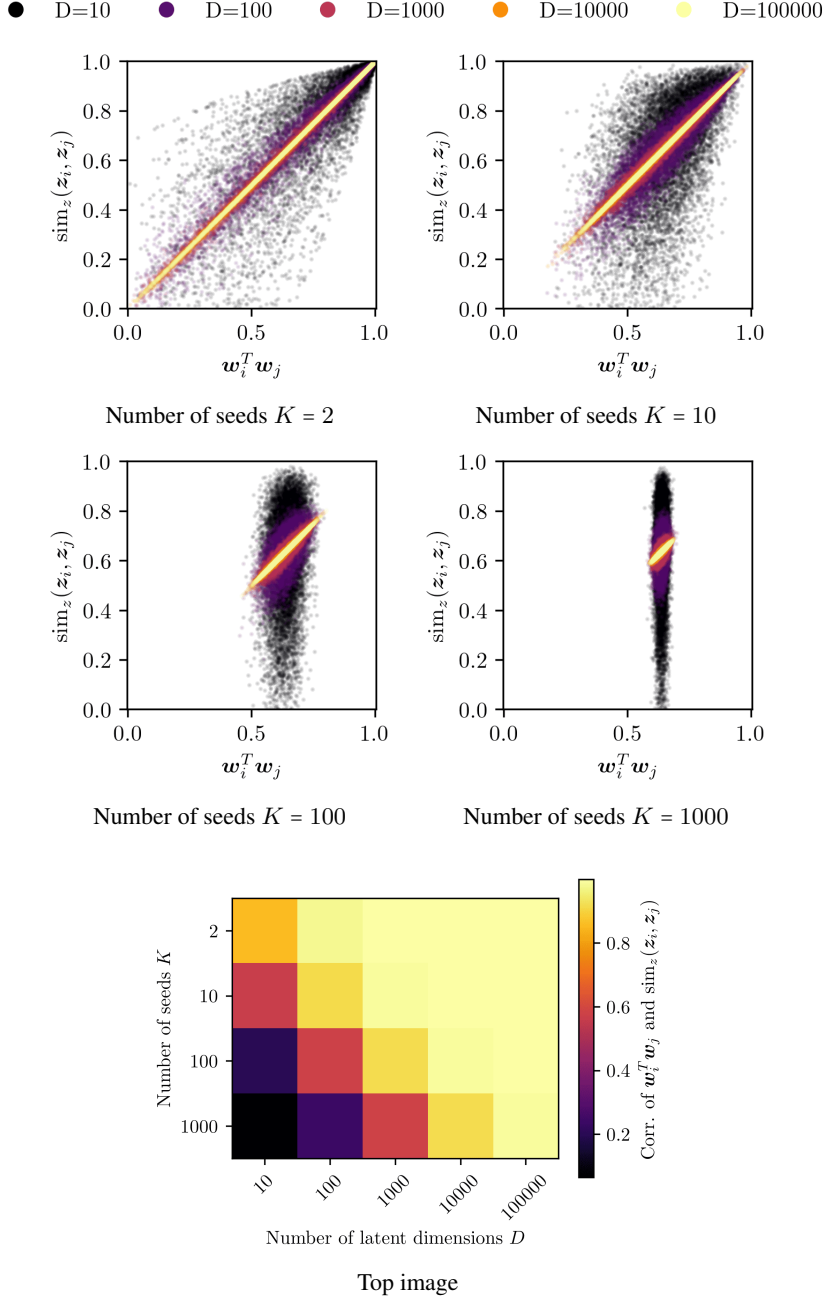


Figure 8: (top) Dot products of the weights $w_i^T w_j$ and the cosine similarity $\text{sim}_z(z_i, z_j)$ for uniformly drawn samples in $\mathcal{U}$ for $K = 2, 10, 100, 1000$, respectively for various dimensions D , where $z \in \mathbb{R}^D$. The number of samples per setting is 10,000, with 100 realisations of the seeds drawn from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and 100 uniformly sampled $\mathbf{u}$ per sampled seeds realisation. (bottom) Estimated correlations between $w_i^T w_j$ and $\text{sim}_z(z_i, z_j)$ uses all the samples per setting.

Since such an isotropic stationary relationship cannot hold exactly for a low-distortion chart from a Euclidean cube to the hypersphere, we instead assess *approximate stationarity*: the induced dot products need not be identical for equal-distance pairs, but should vary predictably and monotonically with Euclidean distance across most of $\mathcal{U}$.

Figure 9 evaluates this property using one million pairs of points in $\mathcal{U}$ for both the Knothe–Rosenblatt (KR) chart and the angular-coordinate chart. Each pair is obtained by sampling two points uniformly along a randomly chosen line through $\mathcal{U}$. For comparison, Figure 10 presents the same analysis when both points are sampled independently and uniformly from $\mathcal{U}$. These two sampling schemes behave differently in high dimensions. Under independent sampling, pairwise distances in the hypercube concentrate sharply, making it difficult to assess the dependence of induced dot products across a broad range of distances. In contrast, line-based sampling produces a wider and more informative range of separations, and is more relevant for optimisation because it probes behaviour along local search directions.

Across the dimensionalities tested, the KR chart exhibits a strong relationship between Euclidean distance in $\mathcal{U}$ and the corresponding weight-space dot product, with this relationship becoming clearer as the dimension increases. By contrast, the angular-coordinate chart shows substantially weaker distance–similarity alignment in higher dimensions, particularly for pairs that are farther apart. These empirical results motivate our use of the KR chart as the default choice for ϕ_w .

G Good examples define spaces with better solutions: details

For our surrogate spaces to be useful they need to contain varied objects that share characteristics with the seed latents; especially those attributes that impact targeted objective functions. Figure 1 presented an illustrative example, where a 2D space formed from three seeds indeed contained large areas of solutions better than the seeds. We now assess this property qualitatively on the benchmark presented in [10], where we seek generations from the diffusion model Stable Diffusion 1.5 [42] that score highly according to ImageReward [54], a measure of alignment with a target prompt. We also report diversity scores by computing one minus the mean cosine similarity of the CLIP [40] embeddings of the images.

Table 1: Scores and diversities of generated images Reported is the median and the 90% confidence interval of the mean score and diversity, respectively, for each method (row) and prompt (shown in the top). Each section of rows correspond to methods using different compute budgets; with standard model sampling, 100 samples are drawn randomly from the (base) model (SD 1.5), while the bottom section are methods requiring training beyond the base model. The ‘1/2 efficiency’ are methods with 100 initial random samples at its disposal before then producing the final 100 samples, and ‘1/6 efficiency’ is the same but using 500 initial samples. The ‘Best-of’-methods use the budget as per their name, and the ‘Grid in $\mathcal{U}$ ’-methods forms surrogate spaces from seeds being the highest scoring samples among the initial random samples, followed by producing a grid of 100 points.

	“A green colored rabbit.”		“Two roses in a vase.”		“Two dogs in the park.”		GPU hrs (training)
	Reward (†)	Diversity (†)	Reward (†)	Diversity (†)	Reward (†)	Diversity (†)	
Standard model sampling	-0.16 [-0.35, 0.03]	0.17 [0.15, 0.18]	0.80 [0.66, 0.90]	0.12 [0.10, 0.13]	0.36 [0.29, 0.43]	0.16 [0.16, 0.17]	N/A
<i>1/2 efficiency</i>							
Best-1-of-2, 100 times	0.54 [0.39, 0.73]	0.17 [0.16, 0.18]	1.22 [1.15, 1.30]	0.09 [0.09, 0.10]	0.63 [0.58, 0.69]	0.16 [0.15, 0.16]	N/A
Best-100-of-200	1.01 [0.78, 1.17]	0.17 [0.15, 0.18]	1.40 [1.34, 1.45]	0.09 [0.09, 0.10]	0.73 [0.68, 0.78]	0.15 [0.14, 0.16]	N/A
Grid in $\mathcal{U}^1$ (2 seeds)	1.67 [1.08, 1.84]	0.06 [0.03, 0.12]	1.43 [0.70, 1.77]	0.05 [0.04, 0.11]	0.83 [0.32, 1.16]	0.09 [0.07, 0.14]	N/A
Grid in $\mathcal{U}^3$ (4 seeds)	1.55 [1.27, 1.77]	0.08 [0.06, 0.13]	1.29 [0.96, 1.53]	0.08 [0.07, 0.11]	0.74 [0.49, 0.93]	0.13 [0.11, 0.15]	N/A
Grid in $\mathcal{U}^5$ (6 seeds)	1.56 [1.28, 1.74]	0.09 [0.07, 0.13]	1.34 [1.01, 1.58]	0.09 [0.07, 0.12]	0.68 [0.53, 0.83]	0.13 [0.11, 0.15]	N/A
<i>1/6 efficiency</i>							
Best-1-of-6, 100 times	1.42 [1.35, 1.50]	0.13 [0.12, 0.14]	1.55 [1.52, 1.58]	0.09 [0.09, 0.10]	0.91 [0.87, 0.95]	0.15 [0.14, 0.15]	N/A
Best-100-of-600	1.65 [1.61, 1.69]	0.11 [0.10, 0.12]	1.64 [1.61, 1.65]	0.09 [0.08, 0.10]	1.00 [0.97, 1.03]	0.15 [0.14, 0.15]	N/A
Grid in $\mathcal{U}^1$ (2 seeds)	1.75 [1.50, 1.86]	0.04 [0.03, 0.08]	1.55 [1.19, 1.78]	0.06 [0.04, 0.11]	0.84 [0.35, 1.13]	0.09 [0.07, 0.11]	N/A
Grid in $\mathcal{U}^3$ (4 seeds)	1.73 [1.31, 1.83]	0.07 [0.05, 0.08]	1.40 [1.21, 1.65]	0.08 [0.06, 0.10]	0.69 [0.32, 0.94]	0.13 [0.11, 0.16]	N/A
Grid in $\mathcal{U}^5$ (6 seeds)	1.71 [1.45, 1.79]	0.07 [0.06, 0.09]	1.32 [1.15, 1.55]	0.09 [0.07, 0.12]	0.68 [0.40, 0.86]	0.14 [0.12, 0.17]	N/A
DPOK	1.62	0.07	1.59	0.11	1.01	0.14	28 (A100)
Adjoint Matching	1.71	0.05	1.50	0.09	1.33	0.13	4 (A100)
Importance Fine-tuning	1.46	0.05	1.53	0.08	1.01	0.12	7 (RTX 4090)

Results: We test our surrogate spaces by seeing if we can form 100 generations (gridded over the surrogate space) that are both good (well-aligned with the target prompt) and diverse. To generate the seed latents that define our surrogate space, we use a budget of S random generations and pick the top K (a stand-in for having a-priori access to ‘good’ seeds). We also report the canonical baselines

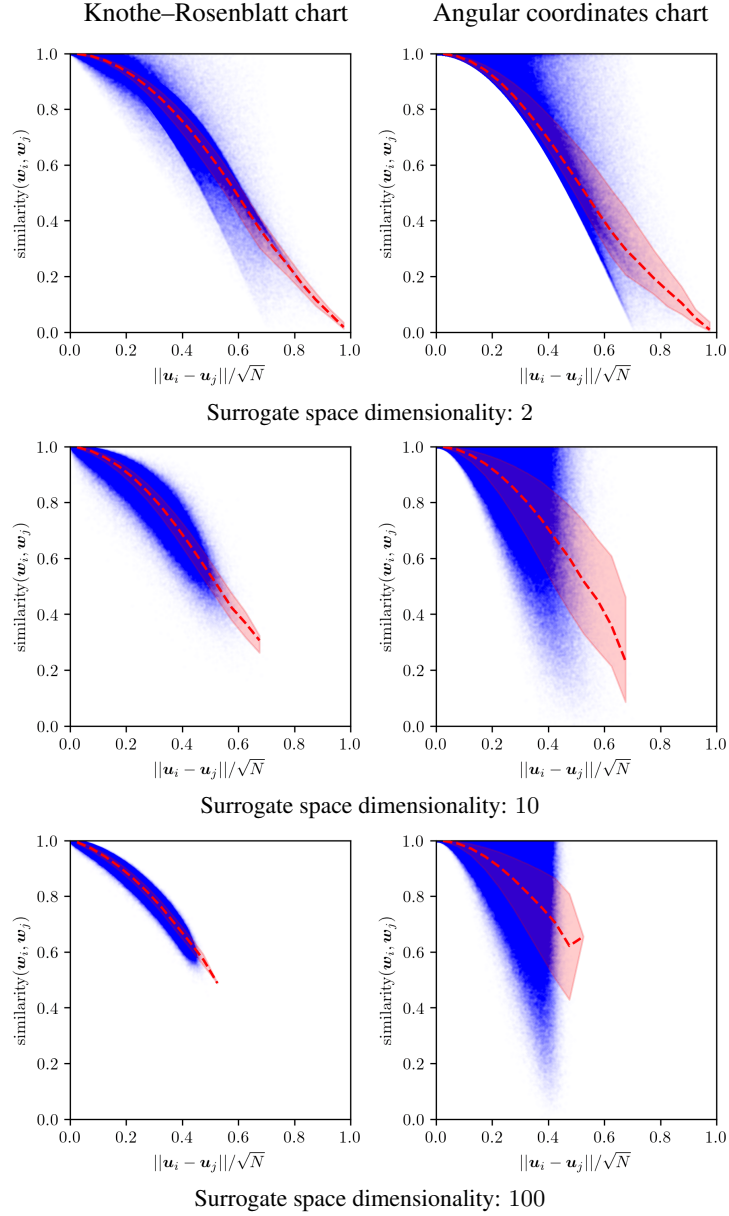


Figure 9: Empirical assessment of approximate stationarity for the Knothe–Rosenblatt chart and the angular-coordinate chart. Blue points show one million sampled pairs of surrogate coordinates in $\mathcal{U}$, plotting their normalised Euclidean distance against the dot product of their corresponding weight vectors. The left column uses the Knothe–Rosenblatt chart and the right column uses the angular-coordinate chart; see Appendix D. The red dashed line shows the mean similarity, and the red shaded region shows the central 50% interval. The top, middle, and bottom rows correspond to surrogate dimensions 2, 10, and 100, respectively, or equivalently to 3, 11, and 101 seed latents. Pairs are generated by repeating the following procedure one million times: (1) sample a point p_1 uniformly in $\mathcal{U}$; (2) form a line segment from p_1 to a point sampled uniformly on the boundary of $\mathcal{U}$; and (3) sample p_2 uniformly along this line segment.

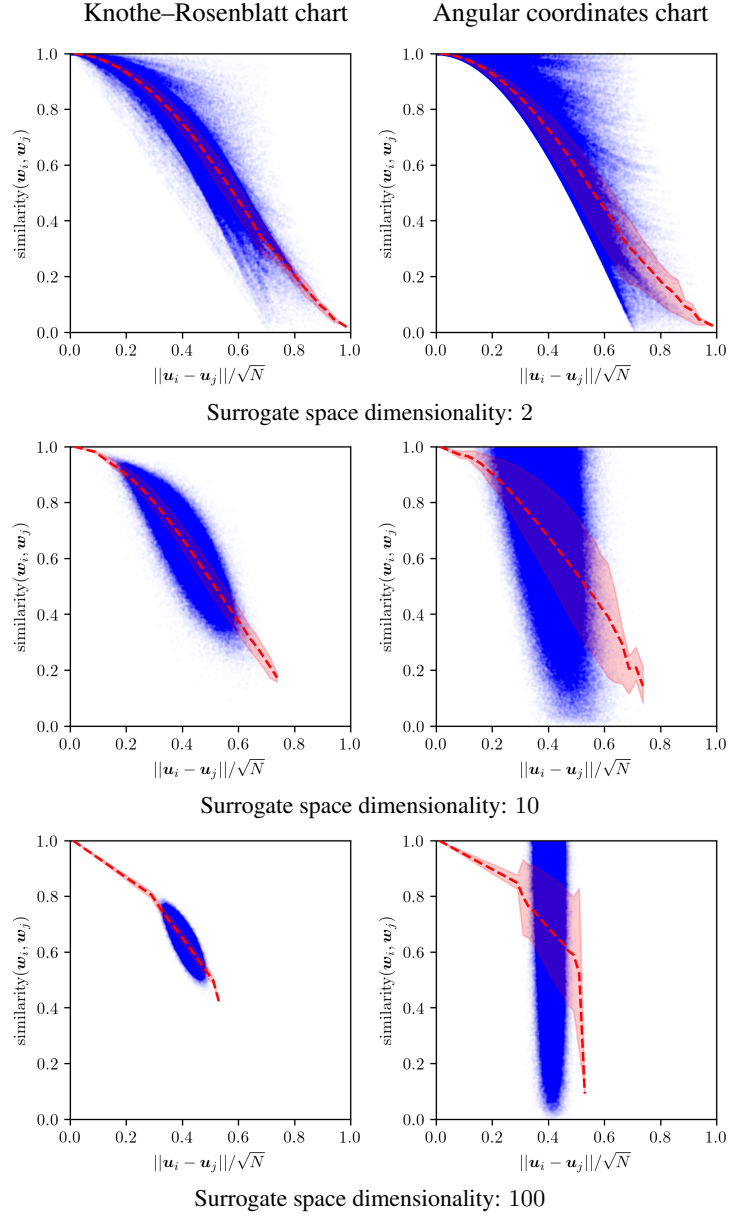


Figure 10: Empirical assessment of approximate stationarity under independent uniform sampling in the surrogate space $\mathcal{U}$. Blue points show one million independently sampled pairs of surrogate coordinates, plotting their normalised Euclidean distance against the dot product of their corresponding weight vectors. The left column uses the Knothe–Rosenblatt chart and the right column uses the angular-coordinate chart; see Appendix D. The red dashed line shows the mean similarity, and the red shaded region shows the central 50% interval. The top, middle, and bottom rows correspond to surrogate dimensions 2, 10, and 100, respectively, or equivalently to 3, 11, and 101 seed latents. The Knothe–Rosenblatt chart maintains a strong negative relationship between Euclidean distance and weight-space dot product, whereas this relationship is substantially weaker for the angular-coordinate chart.

of taking the top 100 directly from the S random generations on each run. Table 1 reports the median and 90% confidence interval of the mean score of the 100 generated images per method over 30 repetitions. To provide context for these scores, we also include the results from [10] who use the same benchmarking setup to compare the performance of algorithms that require many GPU hours of training on the particular score function in order to produce generations with high scores. We include their reported scores (they only provide one repetition) for Importance Fine-tuning [10], DPOK [15], Adjoint Matching [11], demonstrating that, on two out of three prompts, our surrogate spaces produce higher or same scoring generations than the expensive fine-tuning approaches, and similarly diverse. A larger relatively volume of high scoring solutions was observed for the lower dimensional surrogate spaces (but slightly less diverse) — which can be told by the grid yielding high mean scores — which is expected as the lower dimensionalities were produced from better seeds (with more random samples per seed to determine them).

H Surrogate spaces searched by standard optimisation algorithms: details

Goal. We will now confirm that surrogate spaces enable effective LSO in high-dimensional latent variable models by deploying popular optimisation algorithms and compare how they perform in our surrogate latent spaces against the original latent space. Our methodology enables good or informative solutions to guide the search by defining a targeted space, which is to be reflected in the test task. We use the popular methods of CMA-ES [20], BO [44], as well as random search. The objective function is to optimise the Pick score [25] for generations of the Stable Diffusion (SD) 2.1 [42] model. Specifically, the model is given a general prompt (‘A vehicle’) and the objective is to obtain high Pick-scores for a prompt sampled randomly from a grammar composed of three parts as ‘A <attribute> <vehicle type> <environment>’, forming a million possible combinations. The grammar is given in Section O. The sampled target prompts are hidden from the methods, but implicitly conveyed via the objective function; the score as a function of the generated image. At our disposal we have M examples for each part where the attribute, type, or environment match the target, but where the other parts are sampled randomly. This is to simulate the scenario where the practitioner has access to informative but incomplete solutions a-priori, having some of the target characteristics, but not all. This, per setup and sampled prompt, yields a number of seeds of $K = 3M$. The experiment is repeated 10 times; i.e. we sample 10 targets and their corresponding seed examples independently, and apply each optimisation algorithm and weight chart combination to these targets, producing 10 corresponding runs for which we report the median and the 90th confidence interval. For optimiser setups, see I.

Results. Figure 11 demonstrates that optimisers perform better within our surrogate spaces than in the full, original latent space, typically outperforming the best solutions found over a whole run of random search in the full space (i.e. standard sampling from the generative model) in just handful of evaluations. CMA-ES deployed in the original latent space (by specifying points in $\mathbf{u} \in [0, 1]^D$ which are subsequently mapped to latent distribution samples via the inverse Gaussian CDF) failed to produce anything but black images (see Figure 19), which is not surprising as it is unlikely to find a point on the manifold of realistic latent realisations (see [5]). In Figure 12 we report results using surrogate spaces with alternative choices of ϕ_w (see Section J), as well as all combinations of optimisers, including random search in surrogate spaces. Within very low-dimensional surrogate spaces (not the full space), random search was nearly as effective as BO and CMA-ES, but as the dimensionality increased (by providing more seeds) CMA-ES and BO performed substantially better.

I Optimiser setups

For the image and RFDIFFUSION experiments:

- **CMA-ES.** We use the implementation from [36] with population size 4 and $\sigma = 0.2$.
- **BO.** For $K = 3$ and $K = 9$ (i.e. 2D and 8D search problems), we use a Gaussian Process prior with a (3/2)-Matérn kernel and DEFER [4] — with a budget of 300 density function evaluations and 30 hyperparameter posterior samples — for Bayesian inference for the kernel scale, lengthscale, and Gaussian (homoscedastic) noise variance parameters. For $K = 90$ (i.e. 89D search problems), we use Turbo [13] and the author’s official implementation.
- **Random search in $\mathcal{U}$.** Uniform, independent sampling in $\mathcal{U}$.

- **Random search in $\mathcal{Z}$.** Standard random (and independent) sampling from the latent distribution.
- **CMA-ES in $\mathcal{Z}$.** CMA-ES deployed on $[0, 1]^D$, where evaluations are mapped to latent distribution samples via the inverse Gaussian CDF.

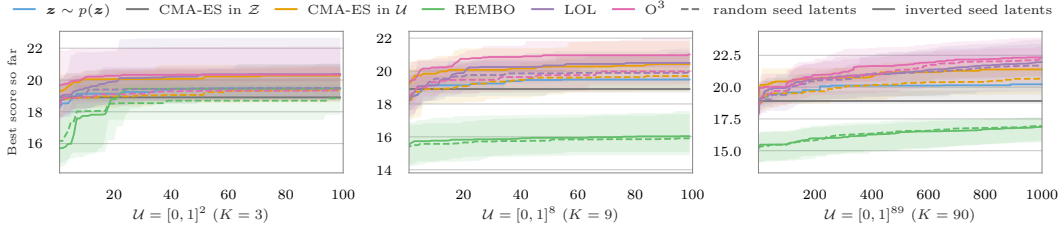


Figure 11: **Image optimisation: best score so far across function evaluations.** Per-iteration median (lines) and 90% confidence interval (shaded, 5th–95th percentile across 10 runs) of the best score reached so far. The three panels correspond to surrogate spaces of increasing dimension induced by $K \in \{3, 9, 90\}$ seed latents, giving $\mathcal{U} = [0, 1]^2$, $[0, 1]^8$, and $[0, 1]^{89}$, respectively. Methods compared: random sampling $z \sim p(z)$, CMA-ES applied directly in $\mathcal{Z}$, CMA-ES in $\mathcal{U}$, and BO in REMBO, LOL, and $\mathcal{U}$. For the latter four, dashed and solid lines indicate random and filtered seed latents, respectively. The optimisation budget is 100 function evaluations for the two low-dimensional panels and 1000 for the high-dimensional one.

J Weight chart optimisation comparison

In this section we report results for combinations of choices of ϕ_w (see Section D).

In Figure 12 we see optimisation results for two different choices of ϕ_w ; the KR and the Angular chart, respectively, in the context of each of BO, CMA-ES, and random search within the formed surrogate space, and include random search (standard sampling) in the full latent space for reference. For optimiser setups, see I. In the context of each combination of choice for ϕ_w , optimiser, and number of seeds, the surrogate spaces substantially outperform random sampling in the full latent space. Using few seeds, i.e. low-dimensional surrogate spaces, both choices of ϕ_w perform similarly, while for the relatively high-dimensional surrogate space ($89D$, formed from 90 seeds) the KR chart substantially outperform the Angular chart when in the context of an optimisation algorithm (BO and CMA-ES) instead of uniform sampling within the surrogate space. Although some differences in final score appear modest, they correspond to large differences in evaluation budget: in this low-budget regime, achieving even small additional improvements by random sampling would require many more model evaluations.

K BOLTZ-2 experiment details

Base model. BOLTZ-2 [39] is a protein-structure prediction model that maps an amino-acid sequence to atomic 3D coordinates via a Pairformer trunk followed by a diffusion decoder, trained in the EDM framework [23]. By default, the decoder samples from a stochastic reverse SDE with per-step noise injection controlled by the γ_0 hyperparameter and stochastic SE(3) coordinate augmentation at each step.

Probability-flow ODE conversion. O^3 requires a deterministic generative map so that latent samples can serve as a basis for the surrogate space $\mathcal{U}$. We adapt BOLTZ-2’s stochastic sampler to its equivalent probability-flow ODE within the EDM framework by setting $\gamma_0 = 0$ and disabling stochastic SE(3) augmentation, yielding a single deterministic Euler integrator from $\sigma_{\max}$ to $\sigma_{\min}$. We further reparametrise the latent prior to a unit Gaussian $z \sim \mathcal{N}(0, I)$; the composite-latents construction $z^* = Z\phi_w(u)$ then applies directly with no additional transport map. The resulting generative map $g: \mathcal{Z} \rightarrow \mathcal{X}$ is deterministic and bijective on its image.

Oracle. We score generated structures with the TM-score [58] against the PDB 1CLL crystal structure of calmodulin [7], computed on C_α atoms using the `tmtools` package [58]. The score is

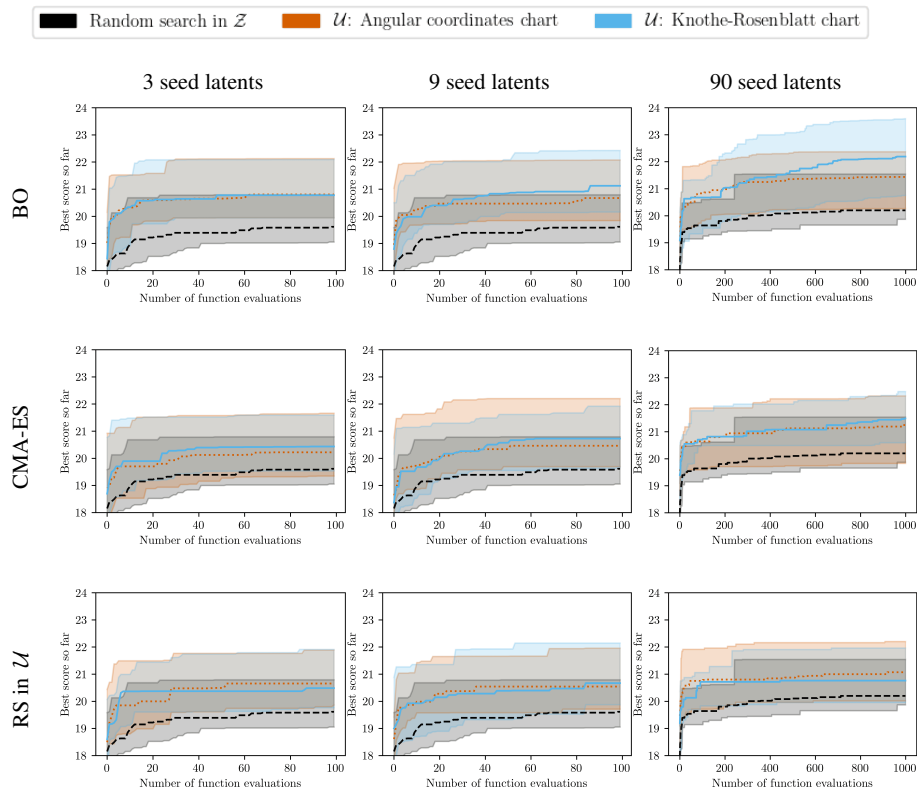


Figure 12: **Chart comparison across optimisers and surrogate space dimensionalities** Shown is the median and 90% confidence interval of the best-so-far score found per step across runs, on the task described in Figure 5 and Section H, for surrogate spaces with dimensionality $(K - 1)$, where K is the number of seeds (examples) provided.

normalised by the reference chain length ($n = 144$ for 1CLL). See Appendix M for the TM-score definition.

Bayesian optimisation setup. At each Bayesian optimisation (BO) acquisition round, we fit a Gaussian process (GP) surrogate [53] on the points acquired so far in $\mathcal{U}$. We use an RBF kernel, a constant mean function, and a single-task GP from BoTorch [3]. Acquisitions are driven by Log Expected Improvement [2], sampled via BoTorch’s Monte Carlo acquisition sampler. We project the K seed latents into $\mathcal{U}$ via the surrogate chart ϕ_w and include them as additional training points alongside an initial set of j uniformly drawn points in $\mathcal{U}$.

Visualisations. Figure 13 shows a LOL interpolation between two seed structures, illustrating that the surrogate space supports smooth structural variation over BOLTZ-2 generations. Figure 14 shows a 2D surrogate space constructed from $K = 3$ seed latents, with the GP posterior mean and the acquired points after two BO rounds.

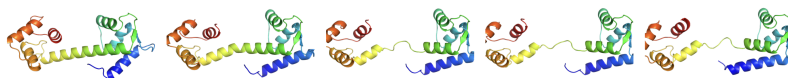


Figure 13: **LOL interpolation in a BOLTZ-2 surrogate space.** Five intermediate structures along the LOL line between two calmodulin seeds (i.e. $K = 2$ seed latents, surrogate space $\mathcal{U}^1$). Structures vary smoothly while remaining within the support of BOLTZ-2.

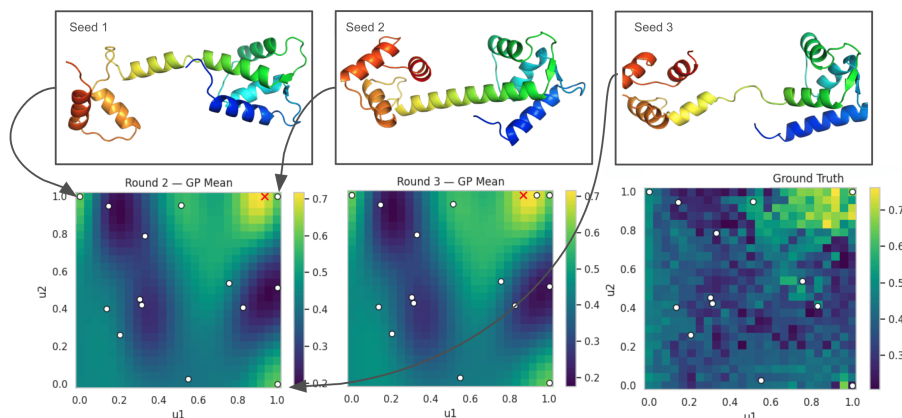


Figure 14: **Bayesian optimisation in a BOLTZ-2 surrogate space defined by $K = 3$ seed latents.** Three calmodulin seed structures (top), the resulting 2D surrogate space $\mathcal{U}^2$ (bottom), the GP training data (white dots) and predictive posterior mean over $\mathcal{U}^2$, the chosen acquisition point (red cross) for two BO rounds, and a 25×25 exhaustive ground-truth grid (bottom right).

Ablations. Below we report three ablations of the $\mathcal{O}^3$ configuration on the 1CLL conformation-biasing task.

Figure 15 sweeps the number of seed latents K (which yields a surrogate space of dimension $K - 1$) for each oracle-call budget N . The best-performing K varies with the budget: at $N = 20$ the strongest configuration uses $K = 3$ seed latents, while at $N = 500$ the strongest is $K = 7$. Different oracle-call regimes favour different numbers of seed latents.

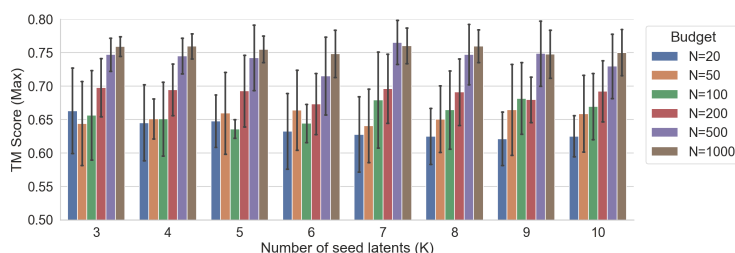


Figure 15: **Number of seed latents K across budgets.** Max TM-score for $\mathcal{O}^3$ with Bayesian optimisation, across number of seed latents K for different budgets N , on 1CLL. The best-performing K varies with N . Bars are means over 5 repetitions; error bars are ± 1 std.

Figure 16 compares $\mathcal{O}^3$ with Bayesian optimisation against $\mathcal{O}^3$ with uniform random sampling in the same surrogate space at $N = 100$. Random sampling alone fails to outperform Best-of- N , while BO does. The gain over Best-of- N comes from the optimisation step in $\mathcal{U}$ rather than from the search space construction itself.

Figure 17 compares two strategies for choosing the K seed latents that define $\mathcal{U}$: *filtered*, which samples m initial structures, scores them, and keeps the top K ; and *random*, which draws K latents directly from the prior. Filtered seed latents give a small but consistent gain at fixed K ; random seed latents remain a useful default when the additional m initial scorings are too expensive.

FK-steering configuration. We use K particles and run Sequential Monte Carlo with reward-based weights computed at intermediate denoising steps where BOLTZ-2 injects positive noise; the final, deterministic step is excluded since reward differences there carry no resampling signal. The reward-scale hyperparameter is $\lambda = 50$. We set the proposal-kernel to the original BOLTZ-2 transition kernel to leverage gradient-free nature of FK-steering, since most biological oracles are non-differentiable. The total oracle-call budget N is distributed throughout denoising: one call at the start of denoising, one at the end, and $N-2$ calls distributed across resampling steps in the first two-thirds of the noise-injecting trajectory.

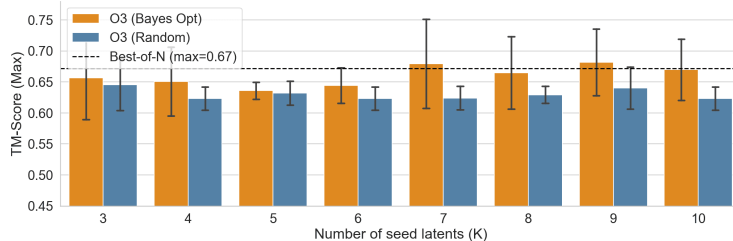


Figure 16: **Bayesian optimisation versus random sampling in the surrogate space.** Max TM-score at $N = 100$ for O^3 with BO and O^3 with uniform random sampling in $\mathcal{U}$, across number of seed latents K . The Best-of- N baseline is shown as a horizontal line. Bars are means over 5 repetitions; error bars are ± 1 std.

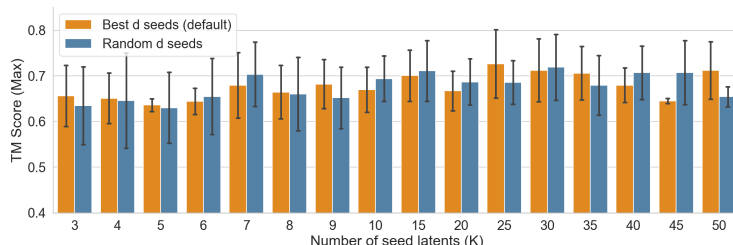


Figure 17: **Seed-selection strategy.** Max TM-score for two seed-selection strategies (filtered and random) across number of seed latents K at $N = 100$. Bars are means over 3 repetitions; error bars are ± 1 std.

Online DPO configuration. At each of E epochs we generate N/E structures from the current model, score them with the oracle, form preference pairs by opposing structures with above-median scores against those with below-median scores, and update the model for one epoch on the resulting pairs. The reference model p_{ref} is reset to the current policy at the start of each epoch, which softens the KL constraint and lets the policy drift further from the pre-trained BOLTZ-2 checkpoint. We use the Diffusion-DPO loss [50] with denoising-score-matching loss differences as a tractable surrogate for the log-likelihood ratio.

L RFDIFFUSION

We adopt the pipeline of Watson et al. [52], consisting of: (1) backbone generation with RFDIFFUSION, (2) sequence design with PROTEINMPNN [9], (3) structure reconstruction with ALPHAFOLD2 [38], and (4) evaluation by C_α -frame RMSE. Lower RMSE indicates closer agreement between the generated and reconstructed backbones.

In step 1, candidate backbones are sampled from the original RFDIFFUSION model using DDIM. A backbone is defined as the set of C_α coordinates and residue-wise rotations, but does not include categorical amino acid identities. In step 2, each backbone is completed with $M = 8$ amino acid sequences predicted by PROTEINMPNN. This introduces the missing categorical information; however, the predictions are noisy, motivating multiple samples. In step 3, the sequences are passed to ALPHAFOLD2, which reconstructs 3D structures from sequence alone, testing whether the backbone proposed by RFDIFFUSION is compatible with realistic sequences. In step 4, reconstructed proteins are aligned to the original backbones, and C_α RMSE is computed. For each backbone we report the best sequence (minimum RMSE over $M = 8$), following the evaluation protocol of Watson et al. [52]. For optimiser setups, see I

As in Watson et al. [52] we adopt a threshold of $T = 2.0 \text{ \AA}$ RMSE to define successful recovery, however we drop their secondary filtering metric of designs having $\text{PAE} < 5.0$ to focus on proof of principle, although in future this could naturally be supported by considering multi-objective optimisation. For fairness, all baselines were recomputed under our evaluation. Each optimisation

run used 200 function evaluations (50 CMA-ES steps of population 4), twice the 100 generations of the original paper.

RFDIFFUSION parametrises a backbone of length N by residue-wise frames $(\mathbf{x}_{pos}^{(t)}, \mathbf{x}_{rot}^{(t)}) \in \mathbb{R}^{3N} \times \text{SO}(3)^N$, where $\mathbf{x}_{pos}^{(t)}$ are C_α coordinates and $\mathbf{x}_{rot}^{(t)}$ are orientations derived from N-C $_\alpha$ -C triplets, measured from a reference frame. The forward diffusion process applies Gaussian noise to $\mathbf{x}_{pos}$ and Brownian motion on $\mathbf{x}_{rot}$; generation is by reverse integration of the probability-flow ODE. The resulting latent is

$$\mathbf{z} = (\mathbf{z}_{pos}, \mathbf{z}_{rot}) = (\mathbf{x}_{pos}^{(T)}, \mathbf{x}_{rot}^{(T)}), \quad \mathbf{z}_{pos} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}^{3N}), \quad \mathbf{z}_{rot} \sim \text{Unif}(\text{SO}(3)^N).$$

Because $\mathbf{z}_{pos}$ follows a Gaussian distribution and we parametrise $\mathbf{z}_{rot}$ as quaternions which are uniformly distributed on $\mathbf{z}_{rot} \sim \text{Unif}(\mathbb{S}^{3N})$, we can directly apply composite latents from Section A to construct surrogate latent spaces $\mathcal{U}$.

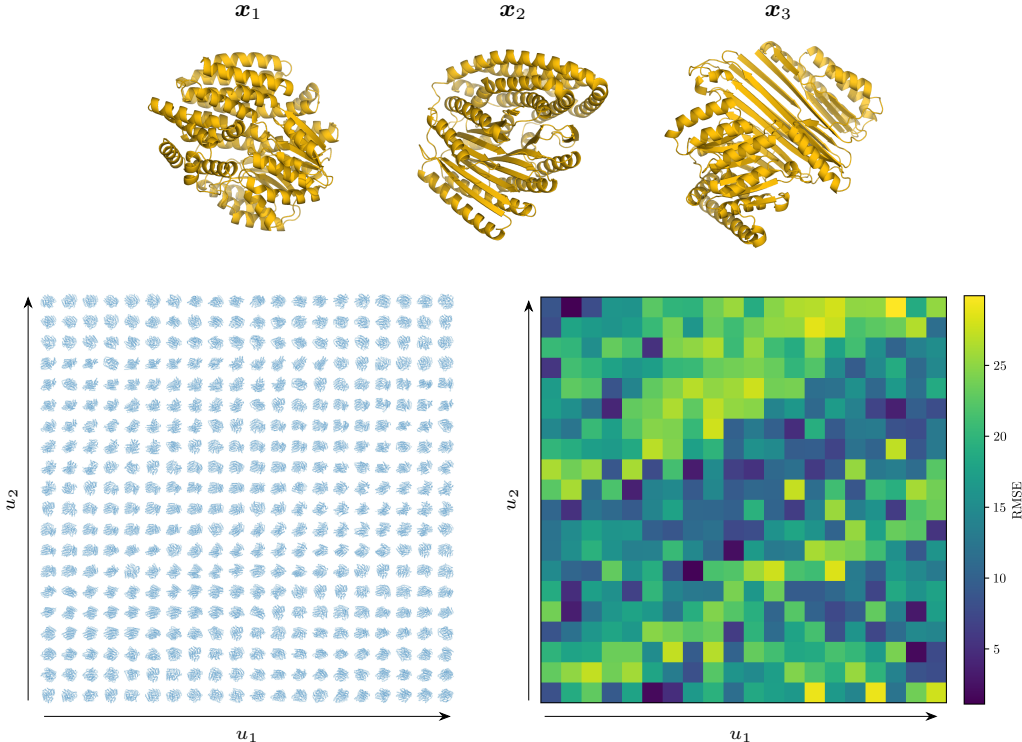


Figure 18: **2D surrogate space for proteins.** A surrogate space $\mathcal{U}^2$ defined by $K = 3$ seed latents (top) yields a structured objective landscape. *Left:* grid of generated backbones across $\mathcal{U}^2$. *Right:* corresponding evaluation scores (C_α -frame RMSE).

Figure 18 shows a 2-dimensional latent space formed from $K=3$ seed latents, over which a grid of protein structures have been generated and evaluated according to the target objective. Clear structure is shown in the objective space which makes this objective amenable to optimisation.

For the protein optimisation experiments we set the number of seed latents to $K = 24$. Optimisation is performed in $\mathcal{U}$ via CMA-ES, with candidates mapped back into $\mathcal{Z}$ for decoding and evaluation. Two seed selection strategies were used. *Random seed latents:* sampled directly from the latent distributions, incurring no additional cost. *Filtered seed latents:* obtained by first generating 100 backbones from the base model, ranking them by RMSE, and selecting the top $K = 24$ latents as seeds. None of these passed the $T = 2.0$ threshold, but they provided a stronger starting point than random seed latents. The extra cost relative to random seed latents is generating and evaluating the pipeline 100 times.

For reference, the top row of Figure 7 provides qualitative context for the C_α -frame RMSE metric. Errors around 5 Å reflect substantial global structural discrepancies, while RMSE values below 2 Å

correspond to largely overlapping backbone geometries, justifying the recovery threshold adopted by Watson et al. [52] and used throughout our experiments.

PAE results. While we optimise and filter on RMSE only, we report the predicted aligned error (PAE) of the best designs across all conditions, allowing comparison against the full recoverability criterion of Watson et al. [52] (which also requires $\text{PAE} < 5$). Table 2 reports best-per-run RMSE and PAE, aggregated across 10 runs of 200 function evaluations.

O^3 substantially improves both metrics over random sampling in $\mathcal{Z}$. Median best-per-run PAE drops from 9.07 for random sampling to 6.34 with random seed latents and 6.84 with filtered seed latents. RMSE drops from 2.33 to 1.19 and 1.08 respectively. Filtered seed latents tighten the run-to-run variance considerably: across 10 runs, the 5th–95th percentile range of best RMSE is $[0.89, 2.82]$ with random seed latents versus $[0.90, 1.81]$ with filtered seed latents, indicating that informative initialisation produces consistently good designs rather than merely a better average.

However, no run achieves the $\text{PAE} < 5$ threshold across any condition, so none of the designs satisfy the full recoverability criterion. Notably, filtered seed latents achieve worse median PAE than random seed latents despite better median RMSE; this is consistent with the single-objective formulation pushing harder on RMSE at slight cost to PAE when given a stronger starting point. Multi-objective extensions to O^3 , which our framework readily supports, are a natural direction for addressing both criteria jointly.

Table 2: Best per-run RMSE and PAE for each method on the 600-residue RFDIFFUSION task, reported as median with 5th–95th percentile range across 10 runs of 200 evaluations. Lower is better for both metrics.

Method	Best RMSE (Å)	Best PAE
Random in $\mathcal{Z}$	2.33 [1.44, 3.82]	9.07 [6.92, 11.87]
CMA-ES, random seed latents	1.19 [0.89, 2.82]	6.34 [5.38, 10.83]
CMA-ES, filtered seed latents	1.08 [0.90, 1.81]	6.84 [5.43, 8.20]

M Template Modelling Score (TM-score)

The Template Modelling score (TM-score) is a widely used measure of structural similarity between two protein backbones. Unlike RMSE, which is sensitive to local deviations and scales poorly with chain length, the TM-score is normalised to the length of the target protein and therefore more suitable for comparing proteins of different sizes [58].

Given a target structure of length L and a comparison structure, the TM-score is defined as

$$\text{TM-score} = \max_{\text{alignments}} \frac{1}{L} \sum_{i=1}^L \frac{1}{1 + \left(\frac{d_i}{d_0(L)}\right)^2}, \quad (18)$$

where d_i is the distance between the i th pair of aligned C_α atoms under a given alignment, and $d_0(L) = 1.24 \sqrt[3]{L - 15} - 1.8$ is a normalisation factor that accounts for protein length. The score lies in $[0, 1]$, with higher values indicating greater structural similarity.

As a rule of thumb, $\text{TM-score} > 0.5$ indicates that two structures share the same fold, while $\text{TM-score} < 0.17$ corresponds to similarity expected by chance.

In our case, all generations are of equal length, so RMSE remains valid; however, using TM-score not only allows us to apply established interpretative thresholds, but also lets us follow Watson et al. [52] in treating two designs as *non-diverse* if their pairwise TM-score exceeds 0.6.

Diversity counting. To compute the number of diverse generations reported in Section 4.5, we apply the following greedy procedure: 1. Sort generated proteins by reconstruction accuracy (lowest RMSE first). 2. Initialise the diverse set with the best structure. 3. For each subsequent protein, compute its TM-score against all members of the current diverse set. 4. Add it to the diverse set if its TM-score is ≤ 0.6 with respect to all previously accepted members; otherwise, discard it.

This ensures that each counted generation is both accurate (passes the RMSE threshold) and structurally distinct under TM-score. *Note:* because a newly generated protein may achieve lower RMSE than existing members of the diverse set while simultaneously being non-diverse with respect to several of them, the overall count of diverse structures can decrease across iterations.

N Audio/Video

Figure 4 illustrates that the surrogate latent space construction applies beyond images, and can be used with generative models operating on high-dimensional continuous representations across diverse modalities. In each case, we form a low-dimensional surrogate space using a small set of seed latents corresponding to a deterministic inversion of targets from the data distribution, and evaluate a dense 2D grid within this space to visualise the resulting generations. For all examples we observe smooth variation of the output across the surrogate coordinates, indicating that the surrogate chart preserves locality even in extremely high-dimensional latent spaces. These qualitative results support the claim that surrogate latent spaces provide well-behaved, low-dimensional manifolds suitable for optimisation across a wide range of generative models and data modalities.

O Image Feature Composition Benchmark grammar

Table 3: **Image Feature Composition Benchmark** vehicle grammar. Each combination of attributes, type, and environment strings are uniformly and independently sampled to form target prompts of the form: ‘A <attributes> <type> <environment>’ for the generation prompt ‘A vehicle’.

Attributes	Types	Environments
"red, shiny"	"sedan car"	"on a mountain road"
"blue, glossy"	"hatchback car"	"by the ocean beach"
"green, matte"	"coupe car"	"in a desert with sand dunes"
"black, reflective"	"convertible car"	"through a forest trail"
"white, clean"	"station wagon car"	"on a snowy mountain peak"
"silver, metallic"	"SUV"	"beside a flowing river"
"gold, polished"	"pickup truck"	"in a dense jungle"
"yellow, bright"	"minivan"	"on a frozen lake"
"orange, vibrant"	"cargo van"	"next to a waterfall"
"purple, glossy"	"limousine"	"in a grassy meadow"
"pink, pastel"	"sports car"	"through a rocky canyon"
"brown, rustic"	"microcar"	"in heavy rainstorm"
"gray, matte"	"standard motorcycle"	"on a wide highway"
"beige, plain"	"motor scooter"	"near an active volcano"
"teal, glossy"	"moped"	"under the northern lights"
"navy blue, shiny"	"dirt bike motorcycle"	"in a futuristic city"
"maroon, matte"	"touring motorcycle"	"inside a highway tunnel"
"ivory, smooth"	"cruiser motorcycle"	"on a suspension bridge"
"bronze, metallic"	"all-terrain vehicle (ATV)"	"beside a tall lighthouse"
"copper, shiny"	"utility task vehicle (UTV)"	"on a sandy dune"
"chrome, reflective"	"monster truck"	"in a busy marketplace"
"pearl white, shimmering"	"golf cart"	"under cherry blossom trees"
"matte black, dull"	"go-kart"	"in front of a medieval castle"
"glossy white, polished"	"city bus"	"at an airport runway"
"emerald green, shiny"	"double-decker bus"	"on a racetrack"
"ruby red, glossy"	"school bus"	"inside a scrapyard"
"sapphire blue, shiny"	"electric trolleybus"	"on a battlefield"
"amber yellow, glowing"	"street tram"	"in an abandoned ghost town"
"charcoal gray, matte"	"light rail train"	"beside a farm barn"
"steel silver, brushed"	"monorail train"	"through vineyards"
"deep purple, glossy"	"subway train"	"on cobblestone streets"
"forest green, matte"	"passenger train"	"in a suburban neighborhood street"
"sky blue, bright"	"freight train"	"beside a skyscraper"
"sunset orange, glowing"	"high-speed train"	"inside a factory yard"
"lemon yellow, bright"	"armored personnel carrier (APC)"	"on a cliffside road"
"rose pink, soft"	"military tank"	"through misty hills"
"sand beige, dusty"	"bulldozer"	"in a crater"
"stone gray, rough"	"excavator"	"inside a dark cave"
"lava red, fiery"	"forklift truck"	"in an abandoned warehouse"
"ice blue, frosty"	"cement mixer truck"	"under a starry night sky"
"neon green, glowing"	"fire engine truck"	"beside a spaceport"
"neon pink, glowing"	"ambulance vehicle"	"at sunset on the horizon"
"pastel blue, soft"	"police patrol car"	"on a frozen tundra"
"pastel yellow, soft"	"tow truck"	"in thick fog"
"midnight black, glossy"	"garbage truck"	"through rice fields"
"frost white, icy"	"snowplow truck"	"beside a wind farm"
"mirror chrome, shiny"	"logging truck"	"under a rainbow"
"brushed aluminum, dull"	"farm tractor"	"near a medieval stone gate"
"glossy teal, shiny"	"combine harvester"	"on an icy highway"
"metallic purple, shiny"	"horse-drawn carriage"	"in a neon-lit street"
"bronze, weathered"	"canoe boat"	"beside a carnival fairground"
"flat black, matte"	"kayak boat"	"in a junkyard"
"desert tan, dusty"	"rowboat"	"through a wheat field"
"jungle green, camo"	"pedal boat"	"in a tropical rainforest"
"navy gray, military"	"sailboat"	"on a wooden boardwalk"
"rust red, corroded"	"luxury yacht"	"at a construction site"
"storm gray, rough"	"catamaran boat"	"on a winding mountain pass"
"bright yellow, shiny"	"inflatable dinghy"	"beside a glacier"
"glossy red, polished"	"fishing boat"	"on a cratered moon surface"
"flat white, plain"	"harbor tugboat"	"inside a space station"
"sparkling silver, glittery"	"passenger ferry"	"through an asteroid field"
"dull gray, industrial"	"speedboat"	"on the surface of Mars"
"deep green, glossy"	"jet ski watercraft"	"inside a lunar base"
"ocean blue, wavy"	"hovercraft vehicle"	"inside an aircraft hangar"
"fire orange, glowing"	"houseboat"	"on a rocket launch pad"
"sun gold, shiny"	"pontoon boat"	"at a desert oasis"
"candy apple red, glossy"	"container cargo ship"	"on a tropical island beach"
"storm gray, matte"	"general cargo ship"	"in a canyon riverbed"
"ice silver, frosty"	"oil tanker ship"	"on an offshore oil rig"
"jet black, shiny"	"cruise ship"	"on a dry salt flat"
"steel blue, metallic"	"battleship"	"inside a military base"
"military green, matte"	"aircraft carrier ship"	"in an amusement park"
"glossy maroon, shiny"	"military submarine"	"on a snowy city street"
"matte navy blue"	"destroyer warship"	"beside a frozen waterfall"
"carbon fiber pattern"	"frigate warship"	"at a cultural festival plaza"
"transparent, glassy"	"hot air balloon"	"inside an industrial plant"
"camouflage green, patterned"	"sailplane glider"	"on a dirt trail"
"chrome gold, shiny"	"hang glider"	"in a foggy swamp"
"metallic blue, glossy"	"paraglider"	"beside a mountain lake"
"dark gray, dull"	"airship blimp"	"on a coastal cliff road"
"vibrant purple, glowing"	"helicopter"	"in front of ancient ruins"
"fluorescent yellow, glowing"	"gyrocopter"	"beside a pyramid"
"sparkling white, glittery"	"small propeller aircraft"	"inside an old temple"
"pearl blue, shimmering"	"seaplane"	"through rolling hills"
"shiny copper, metallic"	"amphibious aircraft"	"in a field of flowers"
"bronze, antique"	"commercial airliner jet"	"beside a farmstead"
"olive green, matte"	"private jet plane"	"at a roadside gas station"
"bright teal, glowing"	"supersonic passenger jet"	"inside a futuristic arena"
"plain beige, flat"	"fighter jet aircraft"	"inside a spaceship hangar"
"polished black, shiny"	"bomber aircraft"	"on a collapsing bridge"
"bright gold, reflective"	"stealth aircraft"	"on a volcanic lava plain"
"storm blue, dark"	"quadcopter drone"	"beside a crystal cave"
"camo brown, patterned"	"cargo plane"	"on a wooden pier"
"stealth gray, matte"	"snowmobile vehicle"	"inside a mining colony"
"metallic orange, glossy"	"mountain cable car"	"on a tall city rooftop"
"diamond white, shiny"	"space shuttle orbiter"	"through a canyon pass"
"rust brown, corroded"	"spaceplane vehicle"	"inside a virtual reality world"
"emerald green, glossy"	"rocketship"	"on an alien desert planet"
"jet silver, reflective"	"lunar exploration rover"	"inside a submarine base"
"space black, glossy"	"mars exploration rover"	"inside an underground bunker"

Table 3 lists the possible attributes, vehicle types, and environment strings used for the prompt grammar used in the experiments reported in Section 4 (with details in Section H) and Section J.

P Experiments compute resources

This appendix summarises the compute used by each experiment in the paper. The dominant cost in every experiment is generation and evaluation of the underlying generative model; O^3 itself adds negligible overhead, since the surrogate chart ϕ is evaluated in closed form and the optimisers (BO, CMA-ES, random search) operate on $\mathcal{U} \subseteq \mathbb{R}^{K-1}$ at low dimensionality. We report the type of GPU used, the wall-clock time of a single optimisation run, the number of function evaluations per run, the number of runs, and the resulting total wall-clock. The reported figures do not include preliminary or failed experiments; an estimate of that overhead is given at the end of this appendix.

Table 4: **Compute resources used by the experiments reported in this paper.** ‘Evals/run’ counts oracle (objective) evaluations, each of which corresponds to one full generation through the relevant model pipeline.

Experiment	Model / pipeline	Hardware	Evals/run	# runs	Per-run wall-clock	Total wall-clock
<i>Figure 3 – Latent cosine similarity vs. objective similarity (motivating analysis)</i>						
Pairs analysis, images	Flux (29) + ImageReward + PickScore	RTX 5090	1000 generations per (seed, K); 5 seeds $\times$ 4 values of $K \in \{3, 10, 30, 90\}$	1	~ 75 h	~ 75 h
Pairs analysis, proteins	RFDiffusion + RMSE + TM-score	RTX 4090	450	1	~ 40 h	~ 40 h
<i>Section 4.2 / Appendix G – ImageReward benchmark on SD 1.5 (30 reps per prompt $\times$ 3 prompts = 90 total runs per row)</i>						
Standard model sampling	SD 1.5	RTX 5090	100	90	~ 17 m	~ 25 h
Best-1-of-2, 100 times (1/2 eff.)	SD 1.5	RTX 5090	200	90	~ 33 m	~ 50 h
Best-100-of-200 (1/2 eff.)	SD 1.5	RTX 5090	200	90	~ 33 m	~ 50 h
Grid in $\mathcal{U}^{1:3.5}$ (1/2 eff.)	SD 1.5	RTX 5090	200	90	~ 33 m	~ 50 h
Best-1-of-6, 100 times (1/6 eff.)	SD 1.5	RTX 5090	600	90	~ 1.7 h	~ 150 h
Best-100-of-600 (1/6 eff.)	SD 1.5	RTX 5090	600	90	~ 1.7 h	~ 150 h
Grid in $\mathcal{U}^{1:3.5}$ (1/6 eff.)	SD 1.5	RTX 5090	600	90	~ 1.7 h	~ 150 h
Aggregate wall-clock for the SD 1.5 image experiments above: ~ 625 RTX 5090 h.						
<i>Section 4.3 / Appendix H – Image Feature Composition Benchmark on SD 2.1</i>						
BO / CMA-ES / RS in $\mathcal{U}^K$ ($K=3$ seeds)	SD 2.1	GH200	100	30	~ 0.1 h	~ 3 h
BO / CMA-ES / RS in $\mathcal{U}^K$ ($K=9$ seeds)	SD 2.1	GH200	100	30	~ 0.1 h	~ 3 h
TaRBO / CMA-ES / RS in $\mathcal{U}^{K^2}$ ($K=90$ seeds)	SD 2.1	GH200	1000	30	~ 1 h	~ 30 h
Random / CMA-ES in $\mathcal{Z}$ baseline	SD 2.1	GH200	100-1000	20	~ 1.1 h	~ 22 h
Aggregate wall-clock for the SD 2.1 image experiments above: ~ 60 GH200 h.						
<i>Section 4 – BOLTZ-2 guidance under matched oracle-call budgets</i>						
O^3 + BO	BOLTZ-2 (PF-ODE)	H100	6 budgets up to $N=1000$	5 per budget	~ 0.1 h	~ 3 h
			$N=20$	5	2.7 m	13 m
			$N=50$	5	2.8 m	14 m
			$N=100$	5	3.1 m	16 m
			$N=200$	5	2.9 m	15 m
			$N=500$	5	3.4 m	17 m
			$N=1000$	5	18 m	1 h 31 m
Best-of- N	BOLTZ-2	H100	6 budgets up to $N=1000$	5 per budget	~ 0.1 h	~ 3 h
			$N=20$	5	2.7 m	13 m
			$N=50$	5	2.8 m	14 m
			$N=100$	5	3.1 m	16 m
			$N=200$	5	2.9 m	15 m
			$N=500$	5	3.4 m	17 m
			$N=1000$	5	18 m	1 h 31 m
FK-steering [45]	BOLTZ-2	H100	6 budgets up to $N=1000$	10 per budget	~ 0.1 h	~ 6 h
			$N=20$	10	3.3 m	33 m
			$N=50$	10	3.8 m	38 m
			$N=100$	10	4.1 m	41 m
			$N=200$	10	5 m	50 m
			$N=500$	10	6.9 m	1 h 10 m
			$N=1000$	10	11 m	1 h 50 m
DPO [41, 50] online	BOLTZ-2 (fine-tuned)	H100	6 budgets up to $N=1000$			
			$N=20$	4	4 m	16 m
			$N=50$	4	6 m	24 m
			$N=100$	5	11 m	55 m
			$N=200$	3	17 m	51 m
			$N=500$	5	44 m	3 h 40 m
			$N=1000$	4	1 h 31 m	6 h 04 m
DPO [41, 50] offline	BOLTZ-2 (fine-tuned)	H100	6 budgets up to $N=1000$			
			$N=20$	4	4 m	16 m
			$N=50$	5	7 m	35 m
			$N=100$	4	18 m	1 h 12 m
			$N=200$	5	56 m	4 h 40 m
			$N=500$	4	5 h 40 m	22 h 40 m
			$N=1000$	3	21 h 57 m	65 h 51 m
<i>Section 4.5 / Appendix L – RFDiffusion pipeline at $N=600$ residues</i>						
Random sampling in $\mathcal{Z}$ (baseline)	RFDiffusion + PROTEINMPNN + ALPHAFOLD2	RTX 4090	200	10	~ 33 h	~ 330 h
CMA-ES in $\mathcal{U}^{K-1}$, random seed latents ($K=24$)	RFDiffusion + PROTEINMPNN + ALPHAFOLD2	RTX 4090	200	10	~ 33 h	~ 330 h
CMA-ES in $\mathcal{U}^{K-1}$, filtered seed latents ($K=24$, +100 init.)	RFDiffusion + PROTEINMPNN + ALPHAFOLD2	RTX 4090	100 + 200	10	~ 50 h	~ 500 h
Aggregate wall-clock for the RFDiffusion experiments above: ~ 1000 RTX 4090 h.						
<i>Appendix N – Cross-modality grids (qualitative)</i>						
20 $\times$ 20 slice of TD4 grid	StableAudio2.0 [14]	V100	400	1	~ 1.1 h	~ 1.1 h
7 $\times$ 7 video grid in $\mathcal{U}^6$	HunyuanVideo [26]	V100	49	1	~ 10 h	~ 10 h
2D protein grid in $\mathcal{U}^4$ ($K=3$)	RFDiffusion pipeline	RTX 4090	400	1	~ 66 h	~ 132 h
<i>Appendix P – Stationarity assessment of ϕ_w</i>						
Chart-similarity sampling	(no generative model; closed-form ϕ_w)	CPU	10^6 pairs	12 (2 charts $\times$ 3 dims $\times$ 2 schemes)	~ 20 s	~ 4 m
<i>Reference baselines from Denker et al. [10] (reproduced from their paper)</i>						
DPOK [15]	SD 1.5 (fine-tuned)	A100	—	—	28 h	28 A100 h
Adjoint Matching [11]	SD 1.5 (fine-tuned)	A100	—	—	4 h	4 A100 h
Importance Fine-tuning [10]	SD 1.5 (fine-tuned)	RTX 4090	—	—	7 h	7 RTX 4090 h

Total project compute, including preliminary experiments. The numbers in Table 4 cover only the experiments reported in the paper. The full research project additionally required compute for preliminary explorations – including alternative weight charts ϕ_w , ablations of seed-selection strategies, alternative optimisers, hyperparameter sweeps, and exploratory generations on image, audio, video, and protein modalities – that did not make it into the final paper. We estimate this additional overhead at approximately 50% of the wall-clock figures reported in Table 4, distributed across the same hardware mix.

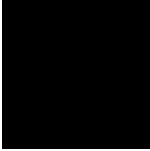
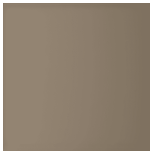
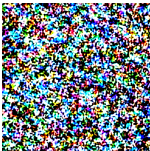
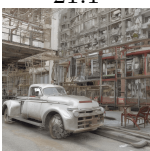
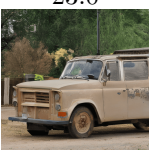
	$\mathcal{U} = [0, 1]^2$ 21.1	$\mathcal{U} = [0, 1]^9$ 21.1	$\mathcal{U} = [0, 1]^{89}$ 21.4
$z \sim p(z)$			
CMA-ES in $\mathcal{Z}$			
REMBO random seed latents (BO)			
REMBO filtered seed latents (BO)			
LOL random seed latents (BO)			
LOL filtered seed latents (BO)			
O^3 random seed latents (BO)			
O^3 filtered seed latents (BO)			

Figure 19: **Best generation found by each search method on the target prompt “A sand beige, dusty cargo van at a construction site”, across surrogate-space dimensionalities (columns) and search methods (rows); the number above each image is its PickScore (higher is better).**