STYLES ARE ISLANDS: THE DISCRETE GEOGRAPHY OF MUSIC GENERATION AND WHY STEERING IS ONE-WAY

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-music models now write usable music from a sentence. Steering functions give such models a second control. A direction is added to a hidden state, or an affine map is applied to it, and an attribute of the music is dialed up or down without touching the prompt. This is the control a user needs when the request is "more of this" rather than a sentence, or when the model takes no prompt at all. The family has worked well for removing attributes, and it carries an unstated assumption: the control that removes a style also adds it. We test this assumption on a text-to-music model with an automatic listener as the judge, and it fails. The same steering vector removes a style almost completely and adds it hardly at all. Removal recovers most of the distance between the two styles, addition about a tenth, and the two dose curves enclose a hysteresis loop. The natural repair is an affine map that matches the shape of the target class. It is more faithful to the class statistics and fails in the same way, as do re-timed pushes and pushes optimized directly against the listener. We resolve the conflict by asking what a style is in state space. A style is not a region but a thin set of trajectories that occupies no volume. Leaving such a set is certain under any push, and landing on it is impossible for every steering function, affine maps included. Transplanting a real state of the target style reaches the style, whereas the same state restricted to its most discriminating directions does not. The diagnosis yields the method. Transport the state to the target class with the optimal affine map, then project it onto the nearest real trajectory of that class and follow it. Neither step suffices alone. Together they close the loop on electronic texture at both model sizes and part of the way on jazz, with the music label preserved, where the steering vector reaches a tenth of the way.

1 INTRODUCTION

Music generation has become one of the most used and most contested products of generative modeling. Text-to-music models (Copet et al., 2023) write a usable backing track from a sentence. The same ease floods streaming services with tracks that no musician made, sometimes under a real musician's name. A musician wants to say "more like this, less like that" to a running performance, and a platform wants to know what a model can be made to do. The control the field offers for a frozen model is the steering function. A direction is added to a hidden state at every step, which is the steering vector, or an affine map is applied to the state (Rodríguez et al., 2025a; Singh et al., 2024; Gaintseva et al., 2026; Jin et al., 2026; Shi et al., 2026; Dalili et al., 2026). The whole family carries an unspoken promise, reversibility: the dial turns both ways.

Whether it does matters beyond music, because a hidden state must be steered exactly where a prompt cannot be changed. The target may have examples but no name, as in "the feel of these three recordings". It may be a response measured on the listener rather than a description. The model may take no prompt at all, as in audio continuation. Adding a style to a performance that lacks it, or removing it from one that has it, is the application. What a method actually does is move the hidden states of a performance from one class of performances to another, and we call the two directions the addition arm and the removal arm.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

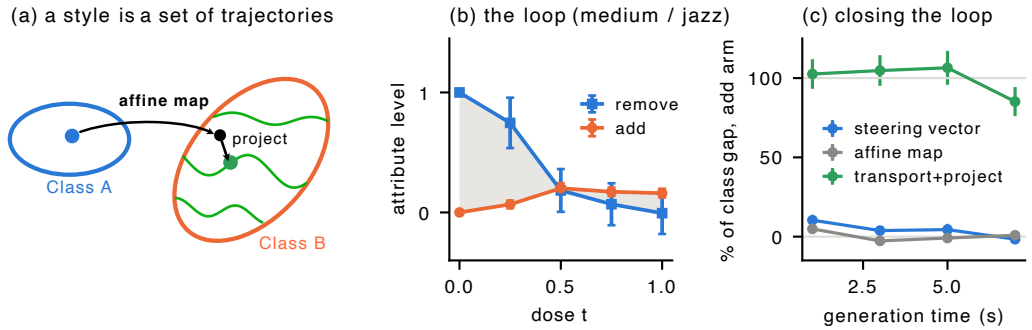

Figure 1: (a) Hidden states of a source class (A, blue) and a target style (B, orange). The style is not the orange region but the green curves inside it, the valid states real performances trace: an affine map lands a state of A beside every curve, the projection moves it onto the nearest one. (b) The loop under the steering vector for one attribute; the enclosed area is the irreversibility index H. (c) Closing the loop: the addition arm in two-second windows.

Our first experiment tests the promise directly (Figure 1b). Take the hidden states the model passes through while it plays jazz, and the states it passes through while it plays a baroque ensemble. Each set is a cloud of points, and the two clouds sit in different places. The line from the center of one cloud to the center of the other is the "jazz direction". Steering pushes every state along this line, toward the baroque cloud to remove jazz and toward the jazz cloud to add it. The removal push works almost completely. The addition push barely moves the music at all, for every attribute and model we tried at one layer. Music steering studies have noticed the same imbalance (Koo et al., 2025; Narashiman et al., 2025; Singh et al., 2026; Zhao et al., 2026) and treated it as a property of the attribute. A picture of two clouds cannot explain this. The two pushes are mirror images, and a mirror image of a cloud is still a cloud, so whatever makes one push succeed should make the other succeed too. Even the best map between the two clouds adds nothing and turns the music into noise. The resolution is that a style is not a cloud. Only a few threads inside the cloud are real performances, and a push lands between the threads, where the model plays nothing in particular. Put a real jazz state in place of a baroque one and the model plays jazz. Put in only the part of that state that points along the jazz direction and it does not.

The literature has approached this from three sides, and each leaves a gap. **Challenge 1: the asymmetry is neither measured nor explained.** Steering methods, from activation addition (Turner et al., 2023; Rimsky et al., 2024; Li et al., 2023) to erasure (Ravfogel et al., 2020; Belrose et al., 2023) and transport maps (Rodríguez et al., 2025a; Singh et al., 2024; Gaintseva et al., 2026), are validated on removal of bias, toxicity, refusal or a style. A stronger negative arm has been noticed (Ali et al., 2025; Tan et al., 2024; Miehling et al., 2025) and attributed to alignment training or baseline skew. None of these accounts predicts anything for a new attribute. **Challenge 2: no steering function adds a style.** Moving the shape of the class with its center (Singh et al., 2024; Rodríguez et al., 2025a), re-timing the push, or optimizing it against the judge all assume that the target is a region that a map can be aimed at. None of them reaches it. **Challenge 3: success is certified inside the model.** Music steering work (Koo et al., 2025; Narashiman et al., 2025; Singh et al., 2026; Prokopiou et al., 2026) reports additions with a monitor in the loop and does not test the removal arm. Criteria such as indistinguishability at the edited layer (Singh et al., 2024) or prompt-reachability (Mishra et al., 2026) certify states the model does not play. We answer the three in order.

1. **We characterize the asymmetry in music models and explain it (Challenge 1).** With a two-arm protocol under an automatic listener we show that removing a style is easy and adding it is hard, for every attribute and model size tested. We quantify the gap as a hysteresis loop and prove that neither the class geometry nor the affine family can produce it. We then show by transplanting real states that a style is a discrete set of trajectories rather than a continuous region, and that this alone produces the asymmetry (Sections 2 and 3.2).

2. **We propose a method that adds a style (Challenge 2).** Transport the state to the target class with the optimal affine map, then project it onto the nearest real trajectory of that class and follow it. The method needs no training, no gradient and no access to the judge. It traces the path from one style to another that a push cannot find (Section 2.2, Algorithm 1).
3. **We validate it against the affine family and a listener-optimized push, with judges outside the model (Challenge 3).** On held-out prompts over full clips the method reaches the target style on electronic texture and part of the way on jazz, with the music label preserved. The steering vector, the affine map and a listener-optimized push reach a tenth of the way, and every component of the method is ablated (Section 3.3).

2 WHY STEERING FUNCTIONS FAIL HERE, AND WHAT A STYLE IS

A reader who skips the formulas should keep the following. First, we show that the asymmetry it measures cannot come from the shapes of the two clouds. A translation moves one cloud onto the other as readily as it moves it back. Second, we propose that a style is a thin set of trajectories inside its cloud, which produces the asymmetry and turns the task into a projection onto real trajectories rather than a push.

2.1 STEERING FUNCTIONS AND THEIR ASYMMETRY

Let $h \in \mathbb{R}^p$ be the residual-stream state at layer L at one generation step. Under prompts that ask for the attribute the states have mean μ_1 and covariance Σ_1. Under prompts for the contrast class they have μ_0 and Σ_0, and $\delta = \mu_1 - \mu_0$. A steering function is a map applied at every step s,

$$h \;\mapsto\; h + \underbrace{w_s}_{\text{step weight}} \big(\underbrace{Mh + b}_{\text{affine target}} - h\big), \tag{1}$$

where M is a matrix, b a vector, and $w_s \geq 0$ a weight that may vary with the step. The steering vector is $M = I$, $b = t\delta$ and $w_s \equiv 1$, with t the dose, and every method of Table 1 is a choice of (M, b, w_s). The reading is an automatic evaluator f of the whole clip, normalized to the class gap. Let $\bar{f}_1$ and $\bar{f}_0$ be the mean readings of unedited clips from the attribute class and the contrast class. An edited clip started from class c scores $(\mathbb{E}f - \bar{f}_c)/(\bar{f}_1 - \bar{f}_0)$, signed so that moving toward the target is positive. A score of 100% is what rewriting the prompt achieves. The removal arm starts in class 1 and pushes $-t\delta$. The addition arm starts in class 0 and pushes $+t\delta$. The irreversibility index H is the area between the addition curve $a(t)$ and the removal curve $1 - r(t)$ over $t \in [0, 1]$, both as attribute level in units of the gap. $H = 0$ is a dial and $H = 1$ a control that never adds.

The natural explanation is geometric: a wide cloud translated into a narrow one overlaps it differently than the reverse. It is wrong twice over. The optimal way to move one Gaussian cloud onto another is affine. Between $N(\mu_0, \Sigma_0)$ and $N(\mu_1, \Sigma_1)$ the transport map (Knott & Smith, 1984) is

$$T(h) = \underbrace{\mu_1}_{\text{target center}} + \underbrace{A}_{\text{shape match}} (h - \mu_0), \qquad A = \Sigma_0^{-1/2}\big(\Sigma_0^{1/2}\Sigma_1\Sigma_0^{1/2}\big)^{1/2}\Sigma_0^{-1/2}, \tag{2}$$

where A is the unique symmetric positive matrix with $A\Sigma_0 A = \Sigma_1$, so T carries the contrast cloud exactly onto the attribute cloud. The transport cost splits into a location part $\|\delta\|^2$ and a shape part, the squared Bures distance $\mathcal{B}^2$ between the two covariances. A translation by any b therefore leaves a residual of at least $\mathcal{B}^2$, and a rank-one direction can only translate (Table 1).

If the downstream network read a state by asking which cloud it more plausibly belongs to, its reading would be the log-odds $r(h)$ below, and a translation would be symmetric:

Observation 1 (translation is symmetric in expected log-odds)**.** *Let $r(h) = \log N(h; \mu_1, \Sigma_1) - \log N(h; \mu_0, \Sigma_0)$ be the log-odds of the attribute fit against the contrast fit. For a contrast state $X_0 \sim N(\mu_0, \Sigma_0)$ pushed by $+\delta$ and an attribute state $X_1 \sim N(\mu_1, \Sigma_1)$ pushed by $-\delta$, $\mathbb{E}\, r(X_0 + \delta) - \mathbb{E}\, r(X_0) = \frac{1}{2}(\delta^\top\Sigma_0^{-1}\delta + \delta^\top\Sigma_1^{-1}\delta) = -\big(\mathbb{E}\, r(X_1 - \delta) - \mathbb{E}\, r(X_1)\big)$, the location part of the Jeffreys divergence, and the shape part is untouched by either arm.*

A reader that thresholds r instead of averaging it is asymmetric, with a clean sign rule. In the coordinates that diagonalize both covariances, each direction along which the attribute class is narrower

method	direction (A, B)	operator M	changes shape?
DiffMean / ActAdd / CAA (Turner et al., 2023; Rimsky et al., 2024)	$(\delta\delta^\top, I)$	I	no
Probe / ITI (Li et al., 2023)	$(\delta\delta^\top, \Sigma)$	I	no
sLDA (Ledoit & Wolf, 2004)	$(\delta\delta^\top, \Sigma_\rho)$	I	no
Proj / INLP / DirAbl (Ravfogel et al., 2020; Arditi et al., 2024)	any d	$I - dd^\top$	collapses along d
LEACE (Belrose et al., 2023)	$(\Sigma_{XZ}, \Sigma_{XX})$	$I - W^+PW$	collapses, whitened
LEACE-Switch (Gaintseva et al., 2026)	same	$I - 2W^+PW$	no (a reflection)
Lin-AcT (Rodríguez et al., 2025a)	per coordinate	diagonal	diagonal only
PAS (Chen et al., 2026)	(Σ_Δ, I), paired diffs	$\beta^2 PP^\top + \beta(I - PP^\top)$	rescales, no ratio
Conceptor (Postmus & Abreu, 2024)	(R_{tgt}, I)	$R(R + \alpha^{-1}I)^{-1}$	yes
MiMiC (Singh et al., 2024)	full	Eq. (2)	yes, all of it
gated transport (ours, App. A)	(Σ_1, Σ_0), gated	Eq. (S1)	certified part
transport + projection (ours)	full	Eq. (2) then Eq. (6)	onto real trajectories

Table 1: Steering methods on two axes: how the direction or subspace is chosen and what the operator does to the class covariance. The Rayleigh pair (A, B) is the pair of matrices whose top generalized eigenvector, the maximizer of $d^\top Ad/d^\top Bd$, is the method's direction. A rank-one A yields a vector, and a vector can only translate; the last row leaves the affine family.

penalizes addition, and each along which it is wider penalizes removal. The narrower class is the hard one to reach. On the measured covariances the attribute class is the wider one, so this reader predicts the opposite of what we observe. So one moves the shape as well, with the full map (2), per coordinate (Rodríguez et al., 2025a), or on certified directions (Appendix A). None of these adds the attribute either, and the full map destroys the music (Section 3.2). The affine family is exhausted, and the asymmetry is not in the clouds.

Suppose instead that the class is not the cloud but a thin set inside it. We use *thin*, *discrete* and, in the title, *islands* for the same object, a set of Lebesgue measure zero; a finite union of curves is the picture. Let $\mathcal{V}_1 \subset \mathbb{R}^p$ be the states the model actually visits when it plays the attribute, a union of trajectories, one per performance, of Lebesgue measure zero, and likewise $\mathcal{V}_0$. A reader that responds to membership in $\mathcal{V}_1$ then behaves as follows.

Lemma 2 (leaving is certain, reaching is impossible)**.** *Let $\mathcal{V} \subset \mathbb{R}^p$ be a set of Lebesgue measure zero (a union of finitely many curves, say) and let S be any steering function of the family (1). Apply S to a state h whose distribution has a density (as the Gaussian fits do), or let S be a random push $h \mapsto h + \xi$ with ξ having a density. Then $\Pr[S(h) \in \mathcal{V}] = 0$ for a state started off $\mathcal{V}$. For a state $h \in \mathcal{V}$ and any S with $S(h) \neq h$, also $\Pr[S(h) \in \mathcal{V}] = 0$, whereas $\Pr[h \in \mathcal{V}] = 1$ before the push. A reader that fires on an ε-neighborhood of $\mathcal{V}$ sees removal at full and addition at $O(\varepsilon^{p-d})$, with d the dimension of $\mathcal{V}$.*

Proof. An affine image or a continuous convolution of an absolutely continuous law is absolutely continuous, and an absolutely continuous law gives measure zero to a null set; the neighborhood statement is the volume of a tube of radius ε around a d-dimensional set in $\mathbb{R}^p$. □

This is the asymmetry in one line. The removal arm asks a push to leave a null set, which any push does. The addition arm asks it to land on one, which no push of the family (1) does, however M, b and w_s are chosen.

Lemma 2 orders the methods of Table 1 into a ladder that Section 3.3 climbs. *The first stage is translation.* The steering vector and its schedules move the center of the cloud. If the class were a region around μ_1 this would reach it, and Observation 1 says it should do so as readily as it leaves. *The second stage is affine transport.* The full map (2), its per-coordinate and gated restrictions, and a push optimized against the listener move the shape as well. They match the class statistics as closely as any map of the family can, so if the class were the cloud they would reach it. *The third stage is projection.* If the class is a thin set, the state must be carried to the set itself. No map of the family does this, and a nearest-neighbor step does. The three stages share prompts, listener and arms, so each failure is evidence for the next premise.

2.2 A STYLE IS A DISCRETE SET, AND THE PROBLEM THAT FOLLOWS

If the attribute class were a region, a small part of a real attribute state should carry most of the attribute. That part could be a line, a plane or its 32 most discriminating directions. If it is a thin set,

only the whole state should. The whole state of a real electronic performance transplanted at every step reaches the class at 83% of its level with the music intact. The same state on a line, a plane or 32 directions gives 3 to 5%, and mixing the whole state in switches on near one half. The set is thin, reaching it is a threshold event on the whole vector, and Lemma 2 applies.

The control problem is then no longer to choose (M, b, w_s). It is to find, for each state, the nearest member of $\mathcal{V}_1$ that the model will continue as the attribute:

$$\hat{h}_s = \underbrace{\arg\min_{v \in \mathcal{V}_1} \big\| v - T(h_s) \big\|_{\Sigma_1^{-1}}}_{\text{projection onto the real states}}, \qquad \underbrace{T(h_s) = \mu_1 + A(h_s - \mu_0)}_{\text{transport, Eq. (2)}}, \tag{3}$$

where h_s is the source state at step s, $\hat{h}_s$ the state written back into the model, and $\|z\|^2_{\Sigma_1^{-1}} = z^\top \Sigma_1^{-1} z$. Here $\mathcal{V}_1$ is represented by a bank of the layer-L states of n unedited clips of the target class, $\{v_{j,u}\}$ indexed by clip j and step u, whose prompts are disjoint from the test set. The natural distance is Mahalanobis under Σ_1, so that directions along which the class varies freely count for little. In practice Σ_1 is estimated at about one sample per dimension, its inverse weights the noisiest directions most, and the plain Euclidean distance finds better neighbors (Section 3.3). So $\| \cdot \|_{\Sigma_1^{-1}}$ below is read as $\| \cdot \|$ unless stated. The transport T carries the state into the right neighborhood, and the projection lands it on the set. Three projection operators realize the argmin. In all three, $x = T(h_s)$ is the transported state at step s, $v_{j,u}$ a bank state, w a window in steps, and k_0 a number of committing steps.

$$\Pi_{\text{NN}}(x)_s = v_{j^\star,u^\star}, \qquad (j^\star, u^\star) = \arg\min_{|u-s| \le w} \|v_{j,u} - x\|_{\Sigma_1^{-1}} \tag{4}$$

$$\Pi_k(x)_s = \sum_{(j,u) \in N_k(x)} \omega_{j,u}\, v_{j,u}, \qquad \omega_{j,u} \propto \exp\big(- \|v_{j,u} - x\|_{\Sigma_1^{-1}} / \bar{d} \big) \tag{5}$$

$$\Pi_{\text{traj}}(x)_s = v_{j^\star,s}, \qquad j^\star = \arg\min_j \sum_{s' \le k_0} \min_{|u-s'| \le w} \|v_{j,u} - x_{s'}\|_{\Sigma_1^{-1}} \tag{6}$$

The *nearest-neighbor* projection (4) takes the argmin afresh at every step within a window w of the current step, so consecutive states may come from different clips. The *soft* projection (5) averages the k nearest bank states $N_k(x)$ with weights ω decaying in distance, $\bar{d}$ being their mean distance. It lands between members of the set. The *trajectory* projection (6) chooses one clip $j^\star$ by the distance summed over the first k_0 steps ($x_{s'}$ being the transported state at step s') and thereafter follows that clip at the same step. Algorithm 1 states the recommended trajectory variant. The bank is a sample of the class, the map is the class statistics, and the whole procedure is one matrix product per step against 20,000 bank states. It applies unchanged to the removal arm with the classes, the map and the bank swapped.

Algorithm 1 Transport, then project onto the nearest real trajectory

Require: bank $\{v_{j,u}\}$ of layer-L states of n unedited target-class clips ($j \le n$, $u \le T$); class statistics μ_0, μ_1 and the map A of Eq. (2); window w; commitment length k_0

$d_j \leftarrow 0$ for all j
for generation step $s = 1, \dots, T$ **do**
  run the forward pass to layer L and read the state h_s
  $h'_s \leftarrow \mu_1 + A(h_s - \mu_0)$ ▷ transport
  if $s \le k_0$ **then**
    $d_j \leftarrow d_j + \min_{|u-s| \le w} \|v_{j,u} - h'_s\|$ for every clip j ▷ distance from clip j's nearest state near step s, summed over steps
    $j^\star \leftarrow \arg\min_j d_j$
  end if
  $\hat{h}_s \leftarrow v_{j^\star,s}$ ▷ follow the chosen trajectory at the same step
  overwrite the layer-L state with $\hat{h}_s$, finish the forward pass, sample the next token
end for

3 EXPERIMENTS

Setup. MusicGen is a language model over EnCodec tokens (Défossez et al., 2023): four token streams at 50 Hz, and a decoder that predicts the next tokens from a text prompt, ten seconds being about 500 steps. We use the 300M `small` model (24 layers, width 1024) and the 1.5B `medium` model (48 layers, width 1536) at pinned revisions, intervening at layer 18 and layer 36, the same relative depth. Three attributes support audio conclusions: `jazz` (contrast: a baroque chamber ensemble), `electronic` (contrast: an unamplified acoustic folk song) and `drums` (contrast: no percussion). Piano, major mode and fast tempo move on neither arm under any method, and we have not tested whether this reflects their evaluators or their states, so we exclude them. A cell is one (model, attribute) pair. Class statistics use 50 prompts per side, effects 40 held-out prompts per side, and each edited clip is paired with the unedited clip from the same prompt and seed. The primary listener is a PANNs AudioSet tagger (Kong et al., 2020) for the genres and the Demucs (Rouard et al., 2023) drum-stem energy for drums. CLAP (Wu et al., 2023) is excluded because its jazz score does not separate the two jazz prompt classes (AUC 0.37 and 0.44, below chance). Because the tagger responds to level, every clip in Sections 3.2 and 3.3 is peak-normalized before scoring. On the loop of Section 3.1 normalization moves removal by at most 4 points and addition by at most 11, so the loop is not a loudness artifact. A second listener, a zero-shot audio–text model (MS-CLAP (Elizalde et al., 2024)), chooses among the attribute, the contrast and three "neither" descriptions (ambient, noise and machinery, organ or brass), and says where the music lands. Effects are percent of the class gap with paired standard errors. Appendices A to C, Tables S1 to S4 and Figure S1 are in the supplementary material.

3.1 THE LOOP

Figure 2 is the central empirical result, with exact values in Table S4 (Appendix C). The shaded area H is the loop's size: if the two arms were mirror images the curves would coincide and H would be 0, and the larger H, the more the control behaves as a one-way valve. In six cells at two sizes, without exception, removal reaches 59 to 101% of the gap. Addition reaches 6 to 16% on the genres and falls below zero on both drums cells. The drum reading is a level-sensitive stem energy that can read destroyed audio as removal, so the two drums values of H are provisional (Appendix B). Against a style-free contrast removal is 73 to 112% and addition -18 to 29%. Under a constant push removal grows with generation time and addition decays to zero (Figure S1).

3.2 WHY THE LOOP DOES NOT CLOSE: A STYLE IS A THIN SET

The layer's geometry predicts the wrong sign: the attribute class is the *wider* one along its most discriminating directions and has the larger entropy, so the thresholded reader of Section 2.1 predicts that removal fails and addition partly succeeds.

What the loop does with a push is visible once the same clips are read with two labels (Figure 3a; Table S2, Appendix B). The push takes the music *out* of the class its prompt asked for on both arms, 76 to 133% in seven of eight arms, and *into* the other class in one arm of eight. The other seven reach it at -20 to 16%. Removal "succeeds" because leaving is what we score on that arm, and addition "fails" because reaching is. Under a symmetric score, the attribute label minus the contrast label in units of that score's own class gap, the addition arm reaches 44 to 48% and the removal arm 36 to 86% at $t = 1$ (Table S2), so neither arm completes a transfer: the asymmetry is between leaving and reaching, which a user experiences as one-way. The music lands on no class at all: a push is an exit, not a transfer. The shape-transport maps and the five schedules of Appendix A (Table S1) never bring addition above 30%.

What does reach the class is a real state (Figure 3b). Transplanting, at every step, the layer-18 state that a real electronic clip had at the same step into a folk generation yields electronic music at 83% of the class level. The music probability stays intact and the second listener agrees. Only part of the same state, its line, its plane or its 32 directions, gives 3 to 5%. Mixing the whole state in switches on between a quarter (5%) and a half (70%), so reaching the class needs most of the coordinates at once. This is the reason behind both halves of the paper. A style is a configuration of the whole state rather than a direction in it, so breaking the configuration takes a push along a few coordinates and

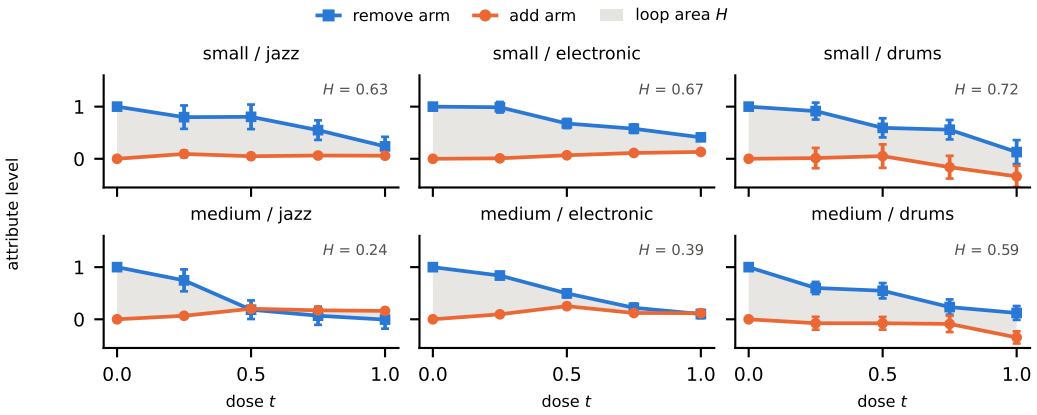

Figure 2: All six cells: attribute level (0 = contrast class, 1 = attribute class) against dose t. Blue: removal arm (start at attribute, push $-t\delta$). Orange: addition arm (start at contrast, push $+t\delta$). Shaded: the irreversibility index H, 0 for a control that turns both ways and 1 for one that removes at once and never adds. Whiskers: one paired standard error. In every cell the blue curve falls and the orange curve never rises.

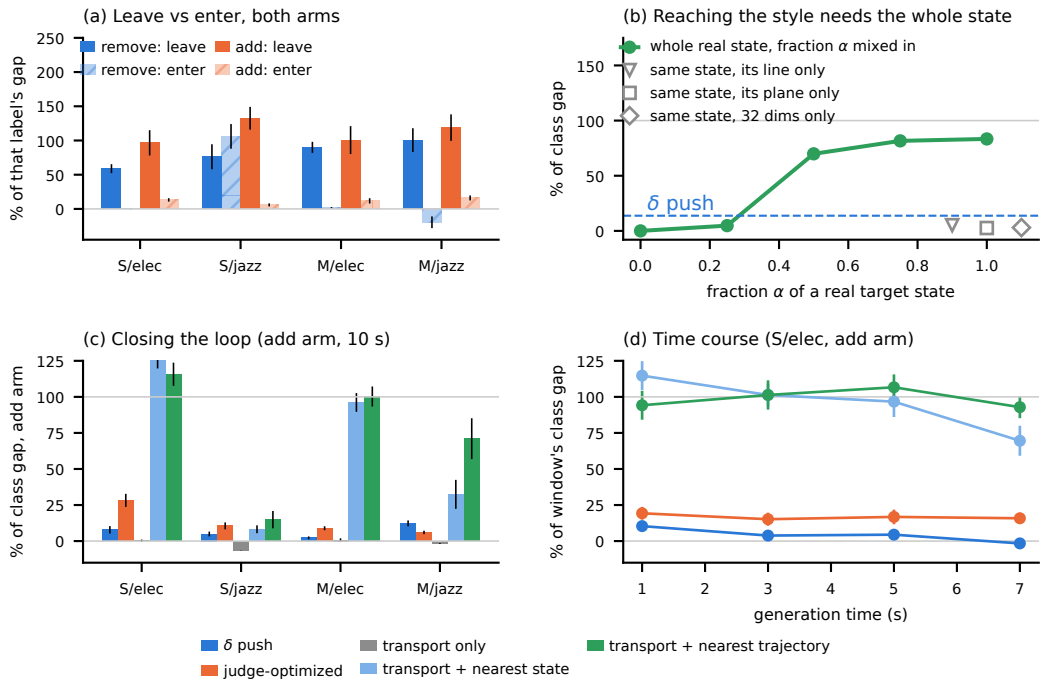

Figure 3: (a) The $t = 1$ push read with two labels: leaving the prompted class (solid) against reaching the other (hatched). (b) A real electronic state mixed into a folk generation at fraction α (green) reaches the style above one half; the same state restricted to a line, a plane or 32 directions (hollow) does not. (c) Addition arm, 10 s, five methods. (d) The same in 2 s windows on small/electronic. Whiskers: one paired standard error.

building one takes all of them, which is the asymmetry. And since only a whole real state is read as the style, the way to add one is to replace the state by a real one, which is the method of Section 2.2.

3.3 CLOSING THE LOOP

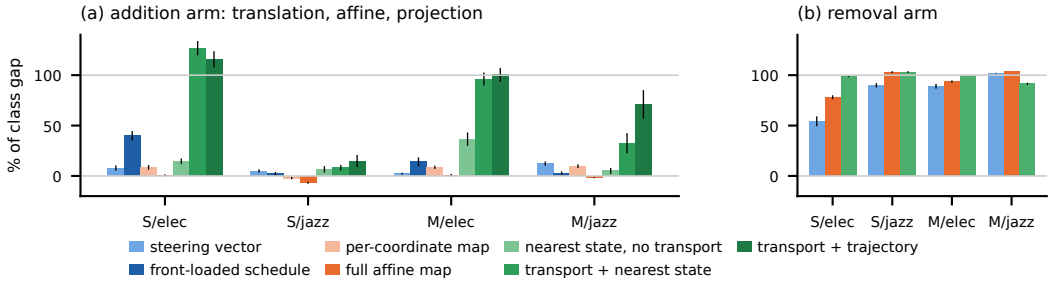

Figure 4: Three stages of the same test, 40 held-out prompts, 10 s, percent of the class gap. Blues: translation (steering vector, best schedule). Oranges: affine transport (per-coordinate, full map). Greens: projection without transport, transport with nearest-state projection, transport with trajectory projection. (a) Addition arm: only the greens reach the class. (b) Removal arm: every family succeeds. Table 2 lists every row.

family	method	S/elec	S/jazz	M/elec	M/jazz
Affine family	steering vector $+\delta$ (translation)	8 (0.47)	5 (0.71)	2 (0.32)	12 (0.69)
	per-coordinate map (Lin-AcT)	9 (0.41)	-2 (0.71)	9 (0.39)	10 (0.72)
	gated shape transport, k=32	3 (0.44)	-2 (0.71)	4 (0.19)	2 (0.68)
	$+\delta$, front-loaded schedule	40 (0.62)	2 (0.74)	14 (0.38)	3 (0.70)
Listener in the loop	gradient-optimized push (straight-through)	28 (0.55)	11 (0.75)	9 (0.46)	6 (0.69)
Projection family (ours)	full Bures–Wasserstein map (transport only)	0 (0.26)	-7 (0.23)	1 (0.13)	-2 (0.33)
	nearest state, no transport	14 (0.33)	7 (0.53)	37 (0.67)	5 (0.49)
	transport + soft 8-NN, Eq. (5)	19 (0.28)	3 (0.53)	18 (0.56)	0 (0.45)
	transport + nearest state, Euclidean, Eq. (4)	127 (0.78)	8 (0.76)	96 (0.80)	32 (0.74)
	transport + trajectory, Euclidean, Eq. (6)	116 (0.81)	15 (0.71)	100 (0.81)	71 (0.75)

Table 2: The addition arm on 40 held-out contrast prompts over 10 s, percent of the class gap (music probability in parentheses). One generation run per row family, audio peak-normalized before scoring, Euclidean distance for the projection rows; one paired standard error is 1 to 6 points (per-clip scores are released). Real attribute prompts score 100 with music 0.75 to 0.83; unedited contrast prompts 0 with 0.78 to 0.81.

Figure 4 and Table 2 run the same 40 held-out contrast prompts through every method of this paper. *Translation alone is not enough.* The steering vector gives 2 to 12% on the addition arm, and re-timing it helps once, at the price of an RMS six times baseline. *Affine is more faithful to the statistics and not enough.* The per-coordinate, gated and full maps close more of the transport gap than any translation can, and they remove faster. The full map reaches 78 to 103% on the removal arm against the vector's 54 to 102%. On the addition arm they give -7 to 10%, the full map with the music probability at 0.13 to 0.33. *Transport, then project, is enough.* Transport with trajectory projection brings the addition arm to 116% on small/electronic and 100% on medium/electronic, and to 15 and 71% on the two jazz cells. The music probability is 0.71 to 0.81 against 0.75 to 0.83 for real attribute prompts. On the removal arm, with the classes, the map and the bank swapped, the projection reaches 91 to 103%. Against a style-free contrast the trajectory projection gives 114, 3, 100 and 24% where every affine method is at or below zero. On drums it gives 109 and 9% (Appendix B). The supplementary demo page plays this ladder for fourteen held-out prompts across the four cells: the same performance under the steering vector, the affine map, transport with nearest-state projection and transport with trajectory projection, next to a real target-style clip. The affine column is audibly degraded, and the style arrives only in the two projection columns. Where a prompt exists, changing it at step 10 reaches 67 to 119%, more than the method on jazz and less on electronic. The method's use is where no prompt exists.

Transport alone lands beside the class, in noise, with the music probability at 0.13 to 0.33. Projection without transport is the nearest house in the wrong neighborhood, 5 to 37%. The per-step projection lands on the set. Under the Mahalanobis distance it fades within two seconds, because states spliced from the wrong clips form no real trajectory, and under the Euclidean distance it holds (Figure 3d). A soft projection onto the eight nearest states, an average of real states, is below the hard projection in every cell. Between the houses is empty, the same threshold Figure 3b shows from the other side.

Started at step 100 or 250 instead of step 1, the same projection reaches only 18 and 14%. Once the model has emitted two seconds of its own audio, the layers above the edited one attend to their own cached past and outvote a real electronic state at layer 18 (Appendix B). The same memory holds a style once it is in: following the trajectory for 60 steps and then releasing gives 100% over ten seconds.

4 Related work

Activation addition (Turner et al., 2023; Rimsky et al., 2024; Li et al., 2023; Zou et al., 2023) adds a fixed vector, the mean of differences being least-squares optimal (Im & Li, 2025). INLP (Ravfogel et al., 2020) and LEACE (Belrose et al., 2023) erase by projection. Transport began with AcT (Rodríguez et al., 2025a) and LinEAS (Rodríguez et al., 2025b). Singh et al. (2024) derive the covariance-matching map we also use and judge success by indistinguishability at the edited layer, whereas our verdict is a listener outside the model. Gaintseva et al. (2026) extend LEACE to switching by a covariance-preserving reflection and a directed cross-covariance map. Conceptors (Postmus & Abreu, 2024) and PAS (Chen et al., 2026) rescale subspaces, and flow (Jin et al., 2026; Shi et al., 2026) and Schrödinger-bridge (Dalili et al., 2026) steering learn nonlinear transports. None tests the map in both directions with direction, operator and step fixed. The projection is a nearest-neighbor step in state space in the spirit of kNN language models (Khandelwal et al., 2020), and the transplant is activation patching (Meng et al., 2022). In music, SMITIN (Koo et al., 2025) steers MusicGen with per-head probes and a per-step monitor and adds drums. Singh et al. (2026), Zhao et al. (2026) and Narashiman et al. (2025) steer with sparse features, recursive feature machines and probes. Staniszewski et al. (2026) steer an audio diffusion model, Prokopiou et al. (2026) close the loop with a PID controller, and MuseControlLite (Tsai et al., 2025) fine-tunes a conditioner.

The reliability literature comes closest without asking the question. Tan et al. (2024) find effects that vary and sometimes reverse. Ali et al. (2025) report negative steering of refusal stronger than positive and blame alignment training. Miehling et al. (2025) find prompt steerability asymmetric and blame baseline skew, and Mishra et al. (2026) show steered states are almost never prompt-reachable. None held direction, operator and step fixed across arms, and all are on aligned models. Hysteresis, the loop enclosed by the forward and backward response of one control, is to our knowledge new to steering.

5 Limitations and conclusion

One model family, one layer per model, one modality, no human listener, and no positive control (an attribute a push can add). Verdicts rest on two automatic listeners that agree on electronic texture and only partly on jazz, where the loop closes half-way at the smaller size. The projection follows a real trajectory of the target class, so after the first ten steps the edited layer carries a state the model produced for another prompt. Little of the source survives: on small/electronic, tempo agreement with the unedited clip is 32% for the projection, 25% for the vector and 18% between unrelated prompts, and we have not yet measured how close the output is to the chosen bank clip. No method here preserves the source while reaching the target. The method needs target-class clips and must start at step one. Within these limits, a push is an exit, a style is a thin set, and transport followed by the nearest real trajectory closes the loop.

REPRODUCIBILITY STATEMENT

The two MusicGen checkpoints are used at pinned Hugging Face revisions (commit hashes `4c8334b0` for `small` and `d3bd7b00` for `medium`), so the weights cannot silently change under the experiments. Generation is deterministic given prompt and seed, and the same prompt and seed produced byte-identical audio across two nodes. Every effect size is computed from stored per-clip scores, which are released. Code, prompt banks, direction bundles, state banks and per-clip scores will be released, and an anonymized copy is provided as supplementary material.

THE USE OF LARGE LANGUAGE MODELS

In this work, we used generative AI tools to implement methods and analysis code, to clean and reformat experimental data, to help formulate the mathematical statements and draft their proofs, to refine hypotheses and help interpret results, and to provide feedback on the research methodology and on the technical content of the paper. We have not used generative AI tools to generate synthetic data sets or to assist with translation, and qualitative or thematic data analysis is not applicable to this work. Additionally, we used generative AI tools for literature search and for polishing the writing and its organization. We have reviewed all AI-assisted work: code was verified and tested by the authors, reported numbers were checked by the authors against the stored per-clip scores, and all text was edited by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work involves no human subjects, no personal data and no listening study; every measurement is made by automatic evaluators on audio generated by a publicly released model under its research license, from text prompts that describe musical styles and instrumentation. Steering methods are dual use: the same intervention that removes an attribute can in principle add one, and the paper's central finding is that the second use is far weaker than assumed, which bears on claims made for such methods in safety and creative settings alike. No capability is introduced that a prompt does not already provide. The evaluators are imperfect proxies for what a listener hears, and the paper says so where it matters.

REFERENCES

Sheikh Abdur Raheem Ali, Justin Xu, Ivory Yang, Jasmine Xinze Li, Ayse Arslan, and Clark Benham. Scaling laws for activation steering with Llama 2 models and refusal mechanisms. *arXiv:2507.11771*, 2025.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *NeurIPS*, 2024. arXiv:2406.11717.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. In *NeurIPS*, 2023. arXiv:2306.03819.

Zheng Chen, Zhaoxin Feng, Yip Tin Po, Jianfei Ma, Emmanuele Chersoni, and Bo Li. PCA-guided activation scaling for monotonic bidirectional control over LLM sycophancy. In *COLM*, 2026. arXiv:2608.16650.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. In *NeurIPS*, pp. 47704–47720, 2023.

Seyed Arshan Dalili, Ajay Narayanan Sridhar, Vijaykrishnan Narayanan, and Mehrdad Mahdavi. Conditional optimal bridge for Riemannian activation steering. *arXiv:2607.10517*, 2026.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *Transactions on Machine Learning Research*, 2023.

Benjamin Elizalde, Soham Deshmukh, and Huaming Wang. Natural language supervision for general-purpose audio representations. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

Tatiana Gaintseva, Andrew Stepanov, Ziquan Liu, Martin Benning, Gregory Slabaugh, Jiankang Deng, and Ismail Elezi. MidSteer: Optimal affine framework for steering generative models. *arXiv:2605.05220*, 2026.

Shawn Im and Yixuan Li. A unified understanding and evaluation of steering methods. *arXiv:2502.02716*, 2025.

Zehao Jin, Ruixuan Deng, Junran Wang, Xinjie Shen, and Chao Zhang. Beyond steering vector: Flow-based activation steering for inference-time intervention. *arXiv:2605.05892*, 2026.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations (ICLR)*, 2020.

Martin Knott and Cyril S. Smith. On the optimal mapping of distributions. *Journal of Optimization Theory and Applications*, 43(1):39–49, 1984.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D. Plumbley. PANNs: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020.

Junghyun Koo, Gordon Wichern, François G. Germain, Sameer Khurana, and Jonathan Le Roux. SMITIN: Self-monitored inference-time intervention for generative music transformers. *IEEE Open Journal of Signal Processing*, 6:266–275, 2025. doi: 10.1109/OJSP.2025.3534686.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *NeurIPS*, 2023.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *NeurIPS*, 2022.

Erik Miehling, Michael Desmond, Karthikeyan Natesan Ramamurthy, Elizabeth M. Daly, Pierre Dognin, Jesus Rios, Djallel Bouneffouf, and Miao Liu. Evaluating the prompt steerability of large language models. In *NAACL*, 2025.

Aayush Mishra, Daniel Khashabi, and Anqi Liu. Steered LLM activations are non-surjective. *arXiv:2604.09839*, 2026. ICLR 2026 Workshops (Sci4DL, Re-Align).

Swathi Narashiman, Pranay Mathur, Dipanshu Panda, Jayden Koshy Joe, Harshith M R, Anish Veerakumar, Aniruddh Krishna, and Keerthiharan A. Genre controlled music generation via activation steering. *arXiv:2506.10225*, 2025.

Joris Postmus and Steven Abreu. Steering large language models using conceptors: Improving addition-based activation engineering. *arXiv:2410.16314*, 2024.

Ioannis Prokopiou, Pantelis Vikatos, Maximos Kaliakatsos-Papakostas, Theodoros Giannakopoulos, and Themos Stafylakis. Closing the loop: PID feedback control for interpretable activation steering in symbolic music generation. In *ICML 2026 Workshop on Machine Learning for Audio (Learning to Listen)*, 2026. arXiv:2606.18790.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *ACL*, 2020.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. In *ACL*, 2024. arXiv:2312.06681.

Pau Rodríguez, Arno Blaas, Michal Klein, Luca Zappella, Nicholas Apostoloff, Marco Cuturi, and Xavier Suau. Controlling language and diffusion models by transporting activations. In *ICLR*, 2025a. arXiv:2410.23054.

Pau Rodríguez, Michal Klein, Eleonora Gualdoni, Valentino Maiorca, Arno Blaas, Luca Zappella, Marco Cuturi, and Xavier Suau. LinEAS: End-to-end learning of activation steering with a distributional loss. In *NeurIPS*, 2025b. arXiv:2503.10679.

Simon Rouard, Francisco Massa, and Alexandre Défossez. Hybrid transformers for music source separation. In *ICASSP*, 2023. arXiv:2211.08553.

Yingdong Shi, Ruiming Zhang, Changming Li, Zhiyu Yang, Kaixing Zhang, Jingyi Yu, and Kan Ren. UniSteer: Text-guided flow matching in activation space for versatile LLM steering. *arXiv:2605.30076*, 2026.

Nikhil Singh, Manuel Cherep, and Pattie Maes. Discovering and steering interpretable concepts in large generative music models. In *ICLR*, 2026.

Shashwat Singh, Shauli Ravfogel, Jonathan Herzig, Roee Aharoni, Ryan Cotterell, and Ponnurangam Kumaraguru. Representation surgery: Theory and practice of affine steering. In *ICML*, 2024. arXiv:2402.09631.

Łukasz Staniszewski, Katarzyna Zaleska, Mateusz Modrzejewski, and Kamil Deja. TADA! tuning audio diffusion models through activation steering. *arXiv:2602.11910*, 2026.

Daniel Tan, David Chanin, Aengus Lynch, Dimitrios Kanoulas, Brooks Paige, Adrià Garriga-Alonso, and Robert Kirk. Analysing the generalisation and reliability of steering vectors. In *NeurIPS*, 2024. arXiv:2407.12404.

Fang-Duo Tsai, Shih-Lun Wu, Weijaw Lee, Sheng-Ping Yang, Bo-Rui Chen, Hao-Chung Cheng, and Yi-Hsuan Yang. MuseControlLite: Multifunctional music generation with lightweight conditioners. In *ICML*, 2025.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv:2308.10248*, 2023.

Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP*, 2023.

Daniel Zhao, Daniel Beaglehole, Taylor Berg-Kirkpatrick, Julian McAuley, and Zachary Novack. Steering autoregressive music generation with recursive feature machines. In *ICLR*, 2026. arXiv:2510.19127.

Andy Zou et al. Representation engineering: A top-down approach to AI transparency. *arXiv:2310.01405*, 2023.