ISOTROPIC STEERING OF LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Activation steering modifies intermediate model activations during inference, providing a promising solution to intervene in the behavior of large language models (LLMs) without updating model parameters. However, interventions from existing steering methods often suffer from strong anisotropy, which harms steering robustness and transferability. To address this issue, we propose an Iso**tr**op**i**c **S**teering (**TRIS**) method for LLMs that calibrates intervention strengths across different steering directions and different activations. In particular, TRIS preconditions the gradient of a steering objective with regularized covariance of activations and obtains normalized interventions in coordinates aligned with the activation distribution. From an optimization perspective, we show that TRIS is equivalent to Steepest Ascent under the Mahalanobis metric. Across four LLMs from the Mistral, Llama, and Qwen families, TRIS improves the primary task metric over ODESteer in 10 of 12 model–task comparisons. In addition, we find that covariance estimated on one task transfers effectively to other tasks within the same model, suggesting that TRIS captures reusable activation distribution and transfers well across tasks.

1 INTRODUCTION

Activation steering controls large language model (LLM) behavior by modifying intermediate activations at inference time (Turner et al., 2023; Zou et al., 2023; Rimsky et al., 2024). Such interventions leave the underlying model parameters unchanged (Subramani et al., 2022; Stolfo et al., 2025), enabling applications such as improving truthfulness and helpfulness and reducing toxic text generation during inference (Li et al., 2023; Kong et al., 2024; Singh et al., 2024; Rodriguez et al., 2025a; Wang et al., 2025a; Zhao et al., 2026). To pursue a desired behavior, a steering method must determine both the direction and strength of intervention in the activation space.

Existing methods derive fixed or activation-dependent intervention directions and control their magnitude through scaling coefficients, step sizes, integration time, calibrated thresholds, or activation-dependent rules (Li et al., 2023; Rimsky et al., 2024; Kong et al., 2024; Hedström et al., 2025; Feng et al., 2026; Zhao et al., 2026). Notably, activation distributions are highly anisotropic (Timkey & van Schijndel, 2021; Huang et al., 2025), with different scales of variation across directions (as shown in Figure 1). Many existing methods measure intervention strength under the Euclidean metric, e.g., MiMiC (Singh et al., 2024), MERA (Hedström et al., 2025), and ODESteer (Zhao et al., 2026). As illustrated in Figure 1, the same Euclidean distance can represent different relative intervention strengths across directions. Such interventions are anisotropic relative to the activation distribution and can lead to over-steering in some directions and under-steering in others. In addition, when steering directions change with the current activation, the same strength setting can produce different relative changes across activations. These variations complicate robust control, as insufficient interventions can fail to induce the desired behavior and excessive interventions can impair model performance (Hedström et al., 2025; Huang et al., 2025). These observations motivate using activation covariance to calibrate intervention strengths relative to natural activation variation and perform isotropic steering in coordinates aligned with the activation distribution.

In this study, we propose Iso**tr**op**i**c **S**teering (**TRIS**) for large language models. TRIS uses a differentiable objective to specify the desired behavior and estimates a regularized covariance from model activations. At each step, it preconditions the objective gradient with this covariance and normalizes the update under the Mahalanobis metric. We show that these updates follow the steepest ascent direction under this metric, equivalently achieving isotropic steering in coordinates aligned

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.

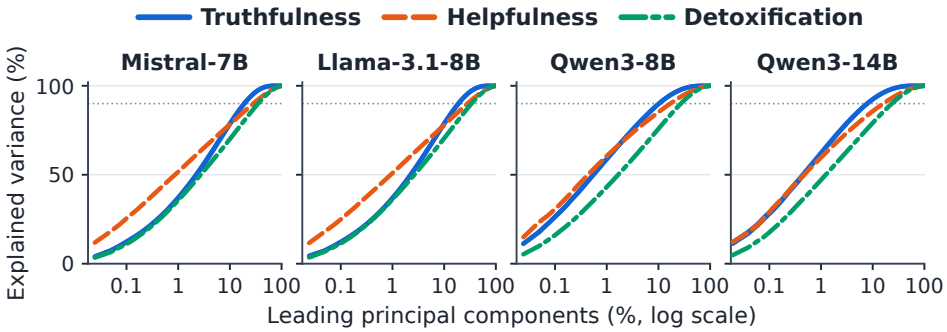


Figure 1: Cumulative activation variance explained across four models and three tasks. The leading 10% of principal components capture 70.1–92.5% of the variance.

with the activation distribution. We integrate the unit-speed field for a flow time T, which directly controls path length in covariance-standardized coordinates. Together, these steps address over- and under-steering across directions and keep intervention strength consistent across activations, enabling robust and effective control of LLM behavior.

We evaluate TRIS on four LLMs from the Mistral, Llama, and Qwen families across truthfulness, helpfulness, and detoxification. Using the same flow-time grid and integration resolution as ODESteer, TRIS achieves effective steering and robust strength selection. In addition, covariance estimated on one task supports effective steering on other tasks within the same model. This suggests that TRIS captures activation statistics that can be reused to calibrate intervention strengths for different target behaviors.

2 Related Work

Steering direction estimation. Fixed directions extracted from contrastive activation differences or principal components encode a target behavior in a vector shared across inputs (Turner et al., 2023; Rimsky et al., 2024; Zou et al., 2023). Linear probes identify behavior-relevant attention heads (Li et al., 2023), and supervised optimization learns intervention vectors from desired outputs (Subramani et al., 2022). AxBench (Wu et al., 2025) evaluates directions from sparse autoencoders alongside supervised low-rank interventions, comparing their effects on concept control and generation quality. Input-dependent methods adapt directions through element-wise activation modulation (Wang et al., 2025b) or generate query-specific corrections through conditional flow matching (Wang et al., 2025a). Gradient-based methods use differentiable attribute, value, or concept functions as steering objectives and evaluate their gradients at the current activation (Dathathri et al., 2020; Kong et al., 2024; Li et al., 2026). ODESteer (Zhao et al., 2026) formulates steering as an ordinary differential equation (ODE), following normalized gradients of a learned density-ratio barrier function as the activation evolves.

Intervention strength control. Intervention strength is controlled through scalar coefficients (Rimsky et al., 2024) or the integration time of a steering flow (Zhao et al., 2026). ITI (Li et al., 2023) scales each intervention by the activation standard deviation along its direction. Adaptive methods vary intervention strength across tokens and layers using classifier predictions (Ferrando et al., 2026), or across activation units using their contributions to the target behavior (Feng et al., 2026). Methods also differ in how they measure intervention strength. MERA (Hedström et al., 2025) minimizes the Euclidean distance between original and steered activations subject to a calibrated prediction-error threshold. Rotational methods control intervention strength through rotation angles while preserving activation norms (Pham & Nguyen, 2024; Vu & Nguyen, 2025; You et al., 2026). Information-geometric approaches seek target changes with limited off-target distortion, measuring output-distribution changes through the information geometry of softmax (Park et al., 2026) or a pullback Fisher metric (Wang et al., 2026).

Covariance-based activation steering. Covariance-based methods use second-order activation statistics to account for feature scales and correlations when constructing an intervention. LEACE (Belrose et al., 2023) uses whitening and projection to remove linearly decodable concept information with minimum distortion, and MidSteer (Gaintseva et al., 2026) extends this

affine framework to concept switching. MiMiC (Singh et al., 2024) derives minimally distorting affine maps by matching class-conditional means and covariances. Related transport methods fit coordinate-wise affine maps to align activation distributions, either separately at each layer (Rodriguez et al., 2025a) or jointly across layers (Rodriguez et al., 2025b). Spectral methods identify subspaces correlated with desired and undesired demonstrations using cross-covariances (Qiu et al., 2024), or construct soft projections from activation second moments (Postmus & Abreu, 2024). TRIS uses regularized covariance estimated from pooled activations to calibrate intervention strengths across directions and activations. It performs steepest ascent under the Mahalanobis metric, yielding isotropic steering in coordinates aligned with the activation distribution.

3 Proposed Method

3.1 An Optimization Viewpoint of Steering

Let $\boldsymbol{x}_0 \in \mathbb{R}^d$ denote a token's residual-stream activation at a selected layer of a frozen language model, where d is the hidden dimension. Following ODESteer (Zhao et al., 2026), we represent activation steering as an ordinary differential equation (ODE):

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(\boldsymbol{x}(t)), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \qquad t \in [0, T], \tag{1}$$

where $\boldsymbol{v} : \mathbb{R}^d \mapsto \mathbb{R}^d$ is a velocity field and t denotes integration time within the intervention. The velocity field determines the steering direction and speed, and the total integration in $[0, T]$ controls overall intervention strength. The steered activation $\boldsymbol{x}(T)$ replaces $\boldsymbol{x}_0$ for the remaining layers.

Let $h : \mathbb{R}^d \to \mathbb{R}$ be a continuously differentiable steering objective that assigns larger values to activations associated with the desired behavior. In the barrier-function formulation of ODESteer (Zhao et al., 2026), the condition $h(\boldsymbol{x}) \geq 0$ defines the preferred region of activation space. With objective gradient $\boldsymbol{g}(\boldsymbol{x}) = \nabla h(\boldsymbol{x})$, the chain rule gives

$$\frac{\mathrm{d}}{\mathrm{d}t} h(\boldsymbol{x}(t)) = \boldsymbol{g}(\boldsymbol{x}(t))^\top \boldsymbol{v}(\boldsymbol{x}(t)). \tag{2}$$

The steepest-ascent direction maximizes this rate under a fixed norm constraint. Under the Euclidean metric, the activation steering is formulated as the local optimization problem

$$\max_{\|\boldsymbol{v}\|_2 \leq 1} \boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{v}. \tag{3}$$

For $\boldsymbol{g}(\boldsymbol{x}) \neq \boldsymbol{0}$, Cauchy–Schwarz (Boyd & Vandenberghe, 2004) gives the unique solution

$$\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x}) = \frac{\boldsymbol{g}(\boldsymbol{x})}{\|\boldsymbol{g}(\boldsymbol{x})\|_2}. \tag{4}$$

This recovers the normalized gradient field of ODESteer (Zhao et al., 2026). The velocity field has unit Euclidean speed wherever the gradient is nonzero, and we set $\boldsymbol{v}_{\mathrm{E}} = \boldsymbol{0}$ when $\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}$.

However, the unit Euclidean speed causes anisotropic steering when accounting for activation variations across directions, which negatively impacts practical applications.

Proposition 1 (Anisotropy of Euclidean Steering)**.** *Let $\boldsymbol{X}$ denote the random activation at the selected layer, with mean $\boldsymbol{\mu} = \mathbb{E}[\boldsymbol{X}]$ and population covariance $\boldsymbol{C} = \mathrm{Cov}(\boldsymbol{X}) \succ \boldsymbol{0}$. Let $c_{\min}$ and $c_{\max}$ be the smallest and largest eigenvalues of $\boldsymbol{C}$, respectively, and define the energy norm in term of $\boldsymbol{C}^{-1}$ as $\|\boldsymbol{v}\|_{\boldsymbol{C}^{-1}} = \sqrt{\boldsymbol{v}^\top \boldsymbol{C}^{-1} \boldsymbol{v}}$. Given an arbitrary instance of $\boldsymbol{X}$, denoted as $\boldsymbol{x}$, the strength of $\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x})$ in the Mahalanobis metric space*[1] *is*

$$\|\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x})\|_{\boldsymbol{C}^{-1}} = \sqrt{\frac{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}^{-1} \boldsymbol{g}(\boldsymbol{x})}{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{g}(\boldsymbol{x})}}. \tag{5}$$

This speed lies in $[1/\sqrt{c_{\max}}, 1/\sqrt{c_{\min}}]$, and its maximum-to-minimum ratio over all directions is $\sqrt{c_{\max}/c_{\min}}$. Consequently, any trajectory following $\boldsymbol{v}_{\mathrm{E}}$ with nonzero gradients on $[0, T]$ satisfies

$$\frac{T}{\sqrt{c_{\max}}} \leq \int_0^T \|\dot{\boldsymbol{x}}(t)\|_{\boldsymbol{C}^{-1}} \, \mathrm{d}t \leq \frac{T}{\sqrt{c_{\min}}}. \tag{6}$$

[1] For any two points $\boldsymbol{x}_1, \boldsymbol{x}_2$ in the Mahalanobis metric space, the Mahalanobis distance between them is $d_{\mathrm{M}}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sqrt{(\boldsymbol{x}_1 - \boldsymbol{x}_2)^\top \boldsymbol{C}^{-1} (\boldsymbol{x}_1 - \boldsymbol{x}_2)}$.

Proof. Substitution of Equation (4) gives Equation (5), whose square is a Rayleigh quotient of $\boldsymbol{C}^{-1}$. Its eigenvalue bounds give the stated speed range, and integration over $[0, T]$ gives Equation (6). □

Proposition 1 means that the unit Euclidean speed does not represent equal intervention strengths relative to the activation variations. This mismatch has two practical consequences:

1. **Directional over- and under-steering:** The same Euclidean speed produces a larger relative change in a low-variance direction and a smaller relative change in a high-variance direction. A strength sufficient in a high-variance direction can therefore *over-steer* a low-variance direction, whereas a strength safe in the low-variance direction can *under-steer* the high-variance direction.
2. **Activation-sensitive strength:** For activation-dependent vector fields, $\boldsymbol{g}(\boldsymbol{x})$ changes across prompts, tokens, and states along a trajectory, so the speed in Equation (5) also changes due to the existence of $\boldsymbol{C}^{-1}$, as shown in Equation (6). A fixed T can therefore induce different path lengths across activation trajectories, making a shared strength setting sensitive to the activations and hyperparameters (e.g., step size) during steering.

As a result, the robustness of the anisotropic steering to activations and hyperparameter settings becomes questionable.

3.2 Isotropic Steering

Motivated by Proposition 1, we propose TRIS to address the above issues. In principle, TRIS replaces the Euclidean constraint in Equation (3) with an energy-norm constraint induced by activation variations. Accordingly, the optimization problem becomes

$$\max_{\|\boldsymbol{v}\|_{\boldsymbol{C}^{-1}} \le 1} \boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{v}. \tag{7}$$

This problem corresponds to the steepest ascent under the Mahalanobis metric, which also has a closed-form solution.

Proposition 2 (Steepest ascent under the Mahalanobis metric)**.** *For $\boldsymbol{C} \succ \boldsymbol{0}$ and $\boldsymbol{g}(\boldsymbol{x}) \neq \boldsymbol{0}$, the unique solution of Equation (7) is*

$$\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) = \frac{\boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}{\sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}}. \tag{8}$$

It satisfies $\|\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})\|_{\boldsymbol{C}^{-1}} = 1$ (e.g., the unit Mahalanobis speed) and attains the maximum value $\sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}$. Notably, this solution is proportional to that of the Mahalanobis steepest ascent problem, i.e., $\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) \propto \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x}) = \arg\max_{\boldsymbol{v}} 2\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{v} - \|\boldsymbol{v}\|^2_{\boldsymbol{C}^{-1}}$.

Proof. The Cauchy–Schwarz inequality gives

$$\boldsymbol{g}^\top \boldsymbol{v} = (\boldsymbol{C}^{1/2}\boldsymbol{g})^\top (\boldsymbol{C}^{-1/2}\boldsymbol{v}) \le \sqrt{\boldsymbol{g}^\top \boldsymbol{C}\boldsymbol{g}}\, \|\boldsymbol{v}\|_{\boldsymbol{C}^{-1}} \le \sqrt{\boldsymbol{g}^\top \boldsymbol{C}\boldsymbol{g}}. \tag{9}$$

Equality holds exactly when $\boldsymbol{C}^{-1/2}\boldsymbol{v} = \boldsymbol{C}^{1/2}\boldsymbol{g}/\sqrt{\boldsymbol{g}^\top \boldsymbol{C}\boldsymbol{g}}$, giving Equation (8). Substitution verifies the speed and the attained rate. □

Figure 2 illustrates the scheme of TRIS. Essentially, the $\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})$ in Equation (8) preconditions the objective gradient $\boldsymbol{g}(x)$ with $\boldsymbol{C}$ and normalizes it to unit Mahalanobis speed. Plugging this velocity field into Equation (1), with zero update at a zero gradient, we can rewrite Equation (2) as

$$\frac{\mathrm{d}}{\mathrm{d}t} h(\boldsymbol{x}(t)) = \sqrt{\boldsymbol{g}(\boldsymbol{x}(t))^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x}(t))} \ge 0. \tag{10}$$

Thus, the continuous flow increases the objective wherever its gradient is nonzero, which supports TRIS's rationale in optimization. Moreover, $\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})$ **leads to isotropic steering in the coordinate system calibrated with the activation distribution.**

Proposition 3 (Isotropic Steering in the Mahalanobis metric space)**.** *For a random activation $\boldsymbol{X}$ defined in Proposition 1, we can whiten its instance by $\boldsymbol{z} = \boldsymbol{C}^{-1/2}(\boldsymbol{x} - \boldsymbol{\mu})$ and define $\widetilde{h}(\boldsymbol{z}) = h(\boldsymbol{\mu} + \boldsymbol{C}^{1/2}\boldsymbol{z})$. Given nonzero gradient $\widetilde{\boldsymbol{g}}(\boldsymbol{z}) = \nabla_{\boldsymbol{z}} \widetilde{h}(\boldsymbol{z})$, the activation steering in the calibrated coordinate system satisfies*

$$\boldsymbol{C}^{-1/2}\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) = \boldsymbol{u}_{\mathrm{E}}(\boldsymbol{z}) = \arg\max_{\|\boldsymbol{u}\|_2 \le 1} \widetilde{g}(\boldsymbol{z})^\top \boldsymbol{u}. \tag{11}$$

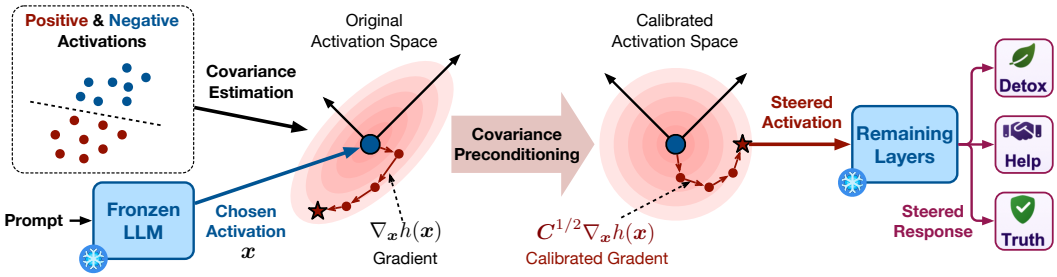


Figure 2: An illustration of our TRIS method.

In other words, TRIS performs normalized gradient ascent in the calibrated coordinate system with $\|\boldsymbol{u}_{\mathrm{E}}(\boldsymbol{z})\|_2 = \|\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})\|_{\mathrm{M}} = 1$*, independently of the steering direction and current activation.*

Proof. Since $\boldsymbol{C}^{1/2}$ is symmetric, the chain rule gives $\widetilde{g}(\boldsymbol{z}) = \boldsymbol{C}^{1/2}\boldsymbol{g}(\boldsymbol{x})$. Substituting $\boldsymbol{v} = \boldsymbol{C}^{1/2}\boldsymbol{u}$ yields $\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{v} = \widetilde{g}(\boldsymbol{z})^\top \boldsymbol{u}$ and $\|\boldsymbol{v}\|^2_{\boldsymbol{C}^{-1}} = \boldsymbol{v}^\top \boldsymbol{C}^{-1}\boldsymbol{v} = \|\boldsymbol{u}\|_2^2$, establishing Equation (11). Moreover, $\|\widetilde{g}(\boldsymbol{z})\|_2^2 = \boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})$. Substituting Equation (8) into $\dot{\boldsymbol{z}} = \boldsymbol{C}^{-1/2}\dot{\boldsymbol{x}}$ then gives $\boldsymbol{C}^{-1/2}\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) = \boldsymbol{u}_{\mathrm{E}}(\boldsymbol{z})$ and unit speed. □

Proposition 3 ensures that TRIS can mitigate the robustness issues discussed in Section 3.1:

1. **Robustness across steering directions.** The unit Mahalanobis speed ensures that the intervention strengths across steering directions match well with the activation distribution. Along each steering direction, the intervention strength is calibrated with the corresponding activation variation, which reduces the risks of over- and under-steering effectively.
2. **Robustness across activations.** The unit Mahalanobis speed holds as the activation changes, so T directly controls the total intervention strength in the calibrated coordinate system. For nonzero gradients almost everywhere on $[0, T]$, we have $\int_0^T \|\dot{\boldsymbol{x}}(t)\|_{\boldsymbol{C}^{-1}}\,\mathrm{d}t = T$. Thus, the same T imposes the same standardized path-length budget even when prompts, tokens, and evolving activations induce different gradient directions, which improves the robustness to the uncertainty of activation. In addition, this unit-speed bound also enhances the robustness to T. For fixed h, $\boldsymbol{C}$, and initial activation $\boldsymbol{x}_0$, let $\boldsymbol{x}_T = \boldsymbol{x}_0 + \int_0^T \dot{\boldsymbol{x}}(t)\mathrm{d}t$ denote a point of the trajectory at time T. Integrating $\|\dot{\boldsymbol{x}}(t)\|_{\boldsymbol{C}^{-1}} \le 1$ between any two such points and applying the triangle inequality gives $\|\boldsymbol{x}_{T_2} - \boldsymbol{x}_{T_1}\|_{\boldsymbol{C}^{-1}} \le |T_2 - T_1|$. Hence, adjusting T changes the steered activation by at most the same amount under the covariance metric.

These properties support effective steering and robust strength selection. Figure 2 is the framework of TRIS, illustrating how the fitted covariance and objective determine steering updates. Obviously, when $\boldsymbol{C} = \boldsymbol{I}$, TRIS degrades to the ODESteer field in Equation (4) for the same objective.

3.3 Practical Implementation

Covariance estimation. Note that estimating a full covariance from limited samples in high dimensions is prone to substantial sampling error (Bickel & Levina, 2008). Through covariance preconditioning, this error perturbs update directions and affects steering effectiveness. Motivated by structural covariance regularization, we adopt a diagonal estimator that retains marginal scale calibration while excluding estimated cross-coordinate couplings. We shrink its variances toward a common scale (Ledoit & Wolf, 2004) and further apply an upper cap, following the principle of condition-number regularization (Won et al., 2013).

Specifically, let $\mathcal{X}_+$ and $\mathcal{X}_-$ contain N_+ positive and N_- negative raw residual activations at the selected layer, respectively. For the pooled set $\mathcal{X} = \mathcal{X}_+ \cup \mathcal{X}_- = \{\boldsymbol{x}_i\}_{i=1}^n$, where $n = N_+ + N_-$ and $\boldsymbol{x}_i = [x_{ij}] \in \mathbb{R}^d$, let $\overline{\boldsymbol{x}} = [\overline{x}_j] \in \mathbb{R}^d$ denote the empirical mean of the activations. We estimate the activation variance along each dimension by

$$s_j = \frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2, \qquad j = 1, \ldots, d. \tag{12}$$

Let $m = \max\{\text{median}_j\, s_j, \epsilon_s\}$, where $\epsilon_s > 0$ is a numerical floor. We shrink each variance toward m and cap unusually large estimates to obtain the regularized diagonal covariance estimate:

$$\widehat{s}_j = \min\{(1-\lambda)s_j + \lambda m, \kappa m\}\,, \qquad \widehat{\boldsymbol{C}} = \text{diag}(\widehat{s}_1, \ldots, \widehat{s}_d), \tag{13}$$

where $\lambda \in (0, 1]$ controls shrinkage and $\kappa \geq 1$ controls the upper cap. Because $\lambda m \leq \widehat{s}_j \leq \kappa m$, the estimate is positive definite and satisfies $\text{cond}(\widehat{\boldsymbol{C}}) \leq \kappa/\lambda$. In this study, we fix $\lambda = 0.6$ and $\kappa = 5$ for every model and task. Importantly, $\widehat{\boldsymbol{C}}$ retains the squared units of the raw activations and is held fixed during generation. Retaining the covariance in raw activation units makes $\boldsymbol{z} = \widehat{\boldsymbol{C}}^{-1/2}(\boldsymbol{x} - \overline{\boldsymbol{x}})$ a standardized coordinate system.

Objective and integration. The integration time T directly specifies the path-length budget. We therefore integrate the flow directly to T, which is implemented by K Euler steps. In particular, we adopt the log density-ratio barrier of ODESteer (Zhao et al., 2026). A learned logistic classifier gives the objective of steering

$$h(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\phi}(\tilde{\boldsymbol{x}}) + b, \tag{14}$$

where $\tilde{\boldsymbol{x}} = \boldsymbol{x}/(\|\boldsymbol{x}\|_2 + \epsilon_\phi)$ with $\epsilon_\phi > 0$, $\boldsymbol{\phi} : \mathbb{R}^d \to \mathbb{R}^D$ is a polynomial TensorSketch feature map (Pham & Pagh, 2013), and $\boldsymbol{w} \in \mathbb{R}^D$ and $b \in \mathbb{R}$ are parameters fitted on the labeled activations $\mathcal{X}$.

Algorithm 1 TRIS

Require: $\mathcal{X}_+, \mathcal{X}_-, \boldsymbol{x}_0, T, K, \epsilon_v, \lambda, \kappa$
1: Obtain $\widehat{\boldsymbol{C}}$ from $\mathcal{X}_+ \cup \mathcal{X}_-$ using (12)–(13).
2: Learn h to fit $\mathcal{X}_+ \cup \mathcal{X}_-$.
3: **for** $k = 0, \ldots, K-1$ **do**
4: $\quad \boldsymbol{g}_k \leftarrow \nabla h(\boldsymbol{x}_k)$
5: $\quad \boldsymbol{v}_k \leftarrow \widehat{\boldsymbol{C}}\boldsymbol{g}_k$
6: $\quad \boldsymbol{v}_k \leftarrow \boldsymbol{v}_k / \max\{\sqrt{\boldsymbol{g}_k^\top \boldsymbol{v}_k}, \epsilon_v\}$
7: $\quad \boldsymbol{x}_{k+1} \leftarrow \boldsymbol{x}_k + (T/K)\boldsymbol{v}_k$
8: **end for**
9: **return** $\boldsymbol{x}_K$

Given h, $\widehat{\boldsymbol{C}}$, and an arbitrary activation $\boldsymbol{x}_0$ obtained during inference, we integrate the flow with K Euler steps of size $\alpha = T/K$, i.e., $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha \boldsymbol{v}_{\text{M}}(\boldsymbol{x}_k)$, for $k = 0, \ldots, K-1$. At each step, we recompute the objective gradient and normalize it after covariance preconditioning. Here, a fixed floor $\epsilon_v > 0$ stabilizes the velocity field denominator. The time T controls intervention strength, while K controls integration resolution. The final state $\boldsymbol{x}_K$ replaces the original activation for the remaining model layers. We evaluate the updates in FP32 and restore the model's original precision afterward. Algorithm 1 shows the algorithmic scheme of TRIS. Appendix A.6 gives a sufficient condition for objective ascent with finite steps, and Appendix A.7 details numerical stabilization.

Computational Complexity. Computing $\widehat{\boldsymbol{C}}$ takes $\mathcal{O}(nd)$ time, and its diagonal representation requires $\mathcal{O}(d)$ storage. For a batch of B activations, diagonal multiplication and normalization add $\mathcal{O}(Bd)$ work per step. Beyond evaluating the gradient, the additional cost for K-step steering is $\mathcal{O}(KBd)$, which is comparable to that of existing methods like ODESteer (Zhao et al., 2026).

4 Experiments

We evaluate steering performance across three tasks, then examine robustness to steering strength, covariance transfer, and estimator efficiency.

4.1 Experimental Setup

Datasets and models. We evaluate TRIS on four base LLMs: Mistral-7B, Llama-3.1-8B, Qwen3-8B, and Qwen3-14B (Jiang et al., 2023; Grattafiori et al., 2024; Yang et al., 2025). We use all 817 TruthfulQA questions for Truthfulness, 500 UltraFeedback prompts for Helpfulness, and 500 toxic prompts from RealToxicityPrompts for Detoxification (Lin et al., 2022; Cui et al., 2024; Gehman et al., 2020). We use two-fold cross-fitting on TruthfulQA. Helpfulness and Detoxification use 10,000 UltraFeedback preference pairs and 10,000 Jigsaw training examples for fitting, respectively.

Baselines. We compare TRIS with NoSteer, ITI, CAA, Linear-AcT, MiMiC, and ODESteer (Li et al., 2023; Rimsky et al., 2024; Rodriguez et al., 2025a; Singh et al., 2024; Zhao et al., 2026). NoSteer leaves the model unchanged. All methods use the same fitting data, decoding settings, and evaluators. Following ODESteer, we select one intervention layer per model using CAA on TruthfulQA and share it across methods and tasks.

Table 1: **Main results.** Best primary scores are bold; second best are underlined.

Model	Method	Truthfulness (TruthfulQA)			Helpfulness (UltraFeedback)			Detoxification (RealToxicityPrompts)		
		T×I (%) ↑	Truth (%) ↑	Info (%) ↑	Win (%) ↑	RM_{mean} ↑	RM_{P90} ↑	Toxic ↓	PPL ↓	Dist-2 ↑
Mistral-7B	NoSteer	37.5	41.5	95.7	50.0	-11.29	-1.13	0.156	18.68	0.989
	ITI	44.8	49.2	95.3	48.5	-11.32	-1.02	0.092	19.15	0.993
	CAA	46.1	51.3	94.7	52.7	-10.62	-0.24	0.122	19.34	0.990
	Linear-AcT	45.9	51.3	94.4	50.8	-10.94	-0.68	0.126	19.50	0.991
	MiMiC	45.3	54.1	90.9	47.2	-11.79	-0.57	0.124	28.92	0.988
	ODESteer	63.3	72.9	90.3	55.3	-9.93	1.01	0.026	23.84	0.994
	TRIS	**68.9**	76.9	91.9	**59.2**	-9.42	1.04	**0.014**	28.35	0.991
Llama-3.1-8B	NoSteer	42.0	43.8	97.6	50.0	-10.23	0.17	0.160	18.68	0.990
	ITI	48.7	51.3	97.1	46.2	-10.67	-1.15	0.100	19.55	0.990
	CAA	49.4	51.7	97.6	54.3	-9.83	0.14	0.128	18.86	0.992
	Linear-AcT	48.7	51.5	96.9	53.9	-9.72	0.53	0.120	19.04	0.990
	MiMiC	49.0	57.2	91.2	53.8	-9.60	0.27	0.102	19.18	0.992
	ODESteer	60.8	65.1	95.3	58.5	-8.40	1.00	0.020	26.91	0.993
	TRIS	**67.3**	74.9	92.4	**65.0**	-7.23	2.76	**0.014**	28.80	0.992
Qwen3-8B	NoSteer	61.7	82.9	76.5	50.0	-6.23	4.35	0.098	25.20	0.997
	ITI	61.9	83.8	76.0	49.1	-6.05	4.57	0.098	26.59	0.998
	CAA	63.9	84.8	77.0	53.0	-5.92	4.13	0.070	25.01	0.998
	Linear-AcT	62.1	83.8	76.4	51.6	-6.07	4.09	0.074	25.22	0.997
	MiMiC	53.5	84.6	66.3	53.3	-5.56	4.91	0.062	25.31	0.997
	ODESteer	**65.9**	89.8	74.7	55.8	-5.70	4.68	**0.022**	27.63	0.998
	TRIS	65.0	89.0	75.3	**56.9**	-4.87	6.00	0.026	28.55	0.999
Qwen3-14B	NoSteer	69.2	89.5	79.2	50.0	-4.87	5.27	0.162	20.79	0.992
	ITI	71.6	89.2	81.8	51.7	-4.56	6.44	0.138	20.85	0.992
	CAA	**72.9**	89.4	82.9	52.2	-4.72	6.51	0.098	21.39	0.993
	Linear-AcT	70.9	89.1	81.4	51.6	-4.72	6.00	0.094	21.12	0.993
	MiMiC	72.7	90.6	81.8	51.3	-4.84	6.29	0.124	20.88	0.993
	ODESteer	71.1	90.2	80.2	54.3	-4.14	6.55	0.058	21.42	0.993
	TRIS	71.8	90.5	80.5	**57.7**	-3.54	7.44	**0.014**	28.36	0.996

Evaluation. Truth×Info measures the fraction of responses judged both truthful and informative; we also report Truth and Info separately. Helpfulness uses the win rate against NoSteer and the mean and ninetieth-percentile scores from the Skywork reward model. Detoxification uses Toxic Rate from RoBERTa, GPT-2 XL perplexity (PPL), and Distinct-2 to measure toxicity, fluency, and lexical diversity.

Implementation details. ODESteer and TRIS share the same fitted objective and use $K = 20$ Euler steps. Following ODESteer, we compare steering strengths and report the best primary score for each model and task. Both methods evaluate integer $T \in [1, 50]$, with Detoxification PPL ≤ 30 required for selection. TRIS fixes the shrinkage weight $\lambda = 0.6$ and upper cap $\kappa = 5$ across all models and tasks. The fitted objective and covariance remain fixed throughout the T sweep. Each method generates one continuation per prompt. Appendix B details data splits and settings.

4.2 Main Comparison

Table 1 shows that TRIS achieves the best primary score in 9 of 12 model–task comparisons and outperforms ODESteer in 10. On TruthfulQA, it achieves the highest Truth×Info on Mistral-7B (68.9%) and Llama-3.1-8B (67.3%), improving over ODESteer by 5.6 and 6.5 percentage points (pp). On Mistral-7B, both Truth and Info improve over ODESteer. Its scores on both Qwen models remain competitive with the strongest baselines.

For Helpfulness, TRIS achieves the highest win rate, mean reward, and ninetieth-percentile reward on all four models. Win-rate gains over ODESteer range from 1.1 to 6.5pp. On Llama-3.1-8B, the mean reward rises from -8.40 to -7.23, and the ninetieth-percentile reward from 1.00 to 2.76.

For Detoxification, TRIS reaches a Toxic Rate of 0.014 on Mistral-7B, Llama-3.1-8B, and Qwen3-14B, reducing toxicity by 46.2%, 30.0%, and 75.9% relative to ODESteer. On Qwen3-8B, its Toxic Rate of 0.026 is comparable to ODESteer's 0.022, while its Distinct-2 is the highest among the methods. Across all four models, PPL stays below 30, with Distinct-2 of 0.991–0.999.

Table 2: **Performance with shared T.** Task scores are percentages; $\sum \Delta$ is the three-task loss in pp. Bold marks the better value.

Model	Method	T	T×I↑	Win↑	Toxic↓	$\sum \Delta \downarrow$
Shared within each model						
Mistral-7B	ODESteer	4	63.28	50.10	2.60	5.20
	TRIS	37	**67.93**	**57.20**	**2.00**	**3.58**
Llama-3.1-8B	ODESteer	5	60.83	58.50	3.80	**1.80**
	TRIS	37	**67.32**	**63.80**	**2.20**	2.00
Qwen3-8B	ODESteer	28	63.65	55.80	5.20	5.20
	TRIS	37	**64.75**	**56.80**	**3.60**	**1.34**
Qwen3-14B	ODESteer	29	69.52	54.30	7.80	3.59
	TRIS	44	**71.85**	**57.30**	**2.60**	**1.60**
Shared across all models						
Mistral-7B	ODESteer	4	63.28	50.10	2.60	5.20
	TRIS	37	**67.93**	**57.20**	**2.00**	**3.58**
Llama-3.1-8B	ODESteer	4	59.24	55.80	5.20	7.49
	TRIS	37	**67.32**	**63.80**	**2.20**	**2.00**
Qwen3-8B	ODESteer	4	60.34	50.30	9.00	17.81
	TRIS	37	**64.75**	**56.80**	**3.60**	**1.34**
Qwen3-14B	ODESteer	4	**69.89**	49.00	14.60	15.32
	TRIS	37	69.16	**57.70**	**2.60**	**3.89**

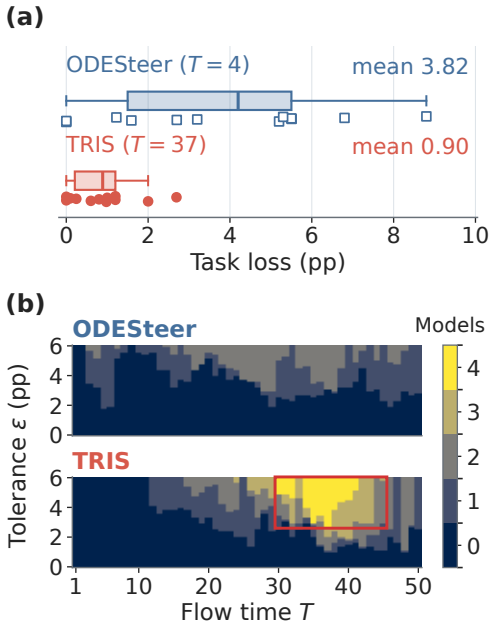


Figure 3: **Robustness to shared T.** (a) 12 model–task losses at global T. (b) Models with all three losses $\leq \epsilon$; red marks high coverage.

4.3 Further Analysis

Robustness to Steering Time. We test whether one T can maintain performance across tasks and models. For ODESteer and our TRIS, let $\Delta_{m,q}(T)$ be the loss in percentage points (pp) from its best score for model m and task q. This is the optimum minus the score for Truthfulness and Helpfulness, and the reverse for Toxic Rate. We select one T per model or one across all models by

$$T_m = \arg\min_T \sum_q \Delta_{m,q}(T), \qquad T_{\text{all}} = \arg\min_T \sum_{m,q} \Delta_{m,q}(T). \tag{15}$$

Both choices use $T \in [1, 50]$ and require Detoxification PPL ≤ 30.

Table 2 shows that TRIS improves all 12 task metrics over ODESteer when each model shares one T across tasks. It selects $T = 37$ for three models and $T = 44$ for Qwen3-14B. A global $T = 37$ improves 11 of 12 metrics over ODESteer at its selected $T = 4$. In Figure 3(a), the mean loss decreases from 3.82pp to 0.90pp (76.4%), and the largest loss from 8.80pp to 2.69pp. Figure 3(b) shows a broad region of high model coverage at $T = 30$–45 for TRIS. This indicates that TRIS is less sensitive to the choice of T when balancing performance across tasks and models.

Cross-Task Covariance Transfer. Using the per-model T in Table 2, we replace only the covariance source and retain the target-task objective and generation settings. Each model has six directed transfers.

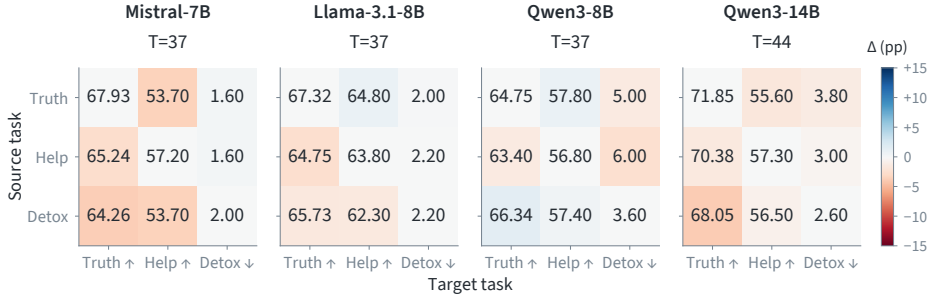


Figure 4: **Cross-task covariance transfer.** Rows: covariance source; columns: target task. Labels: scores (%); colors: gains over the native diagonal (pp), with a shared linear scale.

Figure 5 shows cosine similarities of 0.950–0.990 between covariance diagonals across the 12 task pairs, indicating similar marginal-scale profiles. In Figure 4, 17 of 24 transfers stay within 2pp of the native result; the average signed loss is 1.18pp. On Qwen3-8B, covariance from Detoxification raises Truth×Info from 64.75% to 66.34% and the win rate from 56.80% to 57.40%. On Llama-3.1-8B, covariance from Truthfulness raises the win rate from 63.80% to 64.80% and lowers Toxic Rate from 2.20% to 2.00%. All eight Detoxification transfers keep PPL below 26.4. These results support sharing both covariance and steering time across tasks within a model.

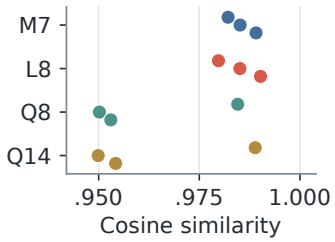


Figure 5: **Covariance similarity.** Three task pairs per model.

Covariance Estimator Analysis. Table 3 compares six covariance estimators on Llama-3.1-8B. Here, D removes off-diagonal entries, S applies shrinkage with $\lambda = 0.6$, and U imposes an upper cap with $\kappa = 5$. Checkmarks indicate enabled components. With all three disabled, row 1 uses the full sample covariance. Row 2 applies shrinkage and capping to its eigenvalues while retaining the full covariance structure. Rows 3–5 retain only the marginal variances, with no regularization, shrinkage alone, or capping alone, respectively. Row 6 is TRIS, which combines diagonalization with both regularizers. Best scores are bold and second best are underlined.

Table 3: **Covariance estimates.** Llama-3.1-8B scores.

	D	S	U	T×I↑	Win↑	Toxic↓
①	×	×	×	57.16	62.30	8.20
②	×	✓	✓	62.79	62.90	2.40
③	✓	×	×	<u>65.85</u>	58.70	4.20
④	✓	✓	×	65.61	59.80	**2.00**
⑤	✓	×	✓	63.04	**64.50**	2.60
⑥	✓	✓	✓	**67.32**	<u>63.80</u>	<u>2.20</u>

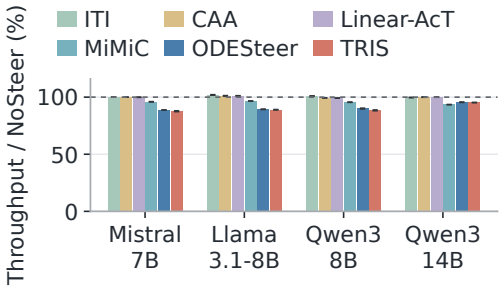


Figure 6: **Relative generation throughput.** NoSteer = 100%; mean ± SD over three timing repeats.

The unregularized diagonal estimate achieves 65.85% Truth×Info, compared with 57.16% for the full sample covariance. Combining diagonalization with both regularizers gives TRIS the highest Truth×Info (67.32%). It also ranks second on Helpfulness and Detoxification, providing a strong balance across the three tasks. Compared with the full covariance estimator with shrinkage and capping, TRIS improves Truth×Info by 4.53pp and win rate by 0.90pp, while reducing Toxic Rate from 2.40% to 2.20%. All six estimators keep Detoxification PPL below 30. Diagonalization retains marginal scale calibration while excluding estimated cross-coordinate couplings. Moreover, diagonal covariance reduces storage and each matrix–vector product from $\mathcal{O}(d^2)$ to $\mathcal{O}(d)$.

To further validate the efficiency of our method, we measure generation throughput on 100 TruthfulQA questions per model using A100 80GB GPUs, FP32, and batch size 10. Timing covers prefill, decoding, sampling, and steering until natural EOS termination. Figure 6 shows that TRIS retains 87.7–95.3% of NoSteer's throughput and 98.3–99.7% of ODESteer's, with each flow method using its selected Truthfulness T and $K = 20$. TRIS thus preserves the efficiency of the underlying gradient flow.

5 Conclusion

We presented TRIS, which uses regularized activation covariance to calibrate intervention strengths across directions and activations. Its updates follow steepest ascent under the Mahalanobis metric, yielding isotropic steering in coordinates aligned with the activation distribution. Experiments on four LLMs across truthfulness, helpfulness, and detoxification show effective steering and robust strength selection. Covariance estimates also transfer across tasks within the same model, indicating that TRIS captures reusable activation statistics. These results show that calibrating interventions to the activation distribution provides a simple basis for robust and effective control of LLM behavior.

AI USE STATEMENT

In this work, we used LLMs to refine the language of the paper, assist with proof checking, and support parts of the code implementation.

ETHICS STATEMENT

This work studies activation steering of large language models to improve truthfulness and helpfulness and reduce toxic generation. The experiments use existing models and benchmark datasets, and the Detoxification data and generated responses may contain offensive language. Steering objectives and automatic evaluators can inherit biases from their data, so improvements in benchmark scores do not establish factuality or safety in deployment. The same steering mechanism could also be used to encourage harmful behavior or alter model outputs without users' awareness. We document the target behaviors, fitting data, intervention settings, and evaluators to make the scope of the reported results explicit.

REPRODUCIBILITY STATEMENT

Section 3 specifies the steering objective, covariance estimator, and integration procedure, with pseudocode in Algorithm 1 and complete derivations in Appendix A. Section 4 describes the experimental comparisons and evaluation criteria. Appendix B documents the models, intervention layers, fitting and evaluation splits, and parameter selection.

REFERENCES

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. In *Advances in Neural Information Processing Systems*, volume 36, pp. 66044–66063. Curran Associates, Inc., 2023.

Peter J. Bickel and Elizaveta Levina. Regularized estimation of large covariance matrices. *The Annals of Statistics*, 36(1):199–227, 2008.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9722–9744. PMLR, 2024.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*, 2020.

Zijian Feng, Tianjiao Li, Zixiao Zhu, Hanzhang Zhou, Junlang Qian, Li Zhang, Jia Jim Deryl Chua, Lee Onn Mak, Gee Wah Ng, and Kezhi Mao. Fine-grained activation steering: Steering less, achieving more. In *International Conference on Learning Representations*, volume 2026, pp. 39421–39443, 2026.

Alex Ferrando, Xavier Suau, Jordi Gonzàlez, and Pau Rodriguez. Dynamically scaled activation steering. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856.

Tatiana Gaintseva, Andrew Stepanov, Ziquan Liu, Martin Benning, Gregory Slabaugh, Jiankang Deng, and Ismail Elezi. MidSteer: Optimal affine framework for steering generative models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 3356–3369. Association for Computational Linguistics, 2020.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Anna Hedström, Salim I. Amoukou, Tom Bewley, Saumitra Mishra, and Manuela Veloso. To steer or not to steer? mechanistic error reduction with abstention for language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 22924–22945. PMLR, 2025.

Yao Huang, Huanran Chen, Shouwei Ruan, Yichi Zhang, Xingxing Wei, and Yinpeng Dong. Mitigating overthinking in large reasoning models via manifold steering. In *Advances in Neural Information Processing Systems*, volume 38, pp. 102543–102568. Curran Associates, Inc., 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Lingkai Kong, Haorui Wang, Wenhao Mu, Yuanqi Du, Yuchen Zhuang, Yifei Zhou, Yue Song, Rongzhi Zhang, Kai Wang, and Chao Zhang. Aligning large language models with representation editing: A control perspective. In *Advances in Neural Information Processing Systems*, volume 37, pp. 37356–37384. Curran Associates, Inc., 2024.

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

Jiaqian Li, Yanshu Li, and Kuan-Hao Huang. Steering vector fields for context-aware inference-time control in large language models. *arXiv preprint arXiv:2602.01654*, 2026.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 41451–41530, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252. Association for Computational Linguistics, 2022.

Kiho Park, Todd Nief, Yo Joong Choe, and Victor Veitch. The information geometry of softmax: Probing and steering. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Ninh Pham and Rasmus Pagh. Fast and scalable polynomial kernels via explicit feature maps. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 239–247. Association for Computing Machinery, 2013.

Van-Cuong Pham and Thien Huu Nguyen. Householder pseudo-rotation: A novel approach to activation editing in LLMs with direction-magnitude perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 13737–13751. Association for Computational Linguistics, 2024.

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

Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo M. Ponti, and Shay B. Cohen. Spectral editing of activations for large language model alignment. In *Advances in Neural Information Processing Systems*, volume 37, pp. 56958–56987. Curran Associates, Inc., 2024.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522. Association for Computational Linguistics, 2024.

Pau Rodriguez, Arno Blaas, Michal Klein, Luca Zappella, Nicholas Apostoloff, Marco Cuturi, and Xavier Suau. Controlling language and diffusion models by transporting activations. In *International Conference on Learning Representations*, volume 2025, pp. 89812–89855, 2025a.

Pau Rodriguez, 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 *Advances in Neural Information Processing Systems*, volume 38, pp. 82356–82398. Curran Associates, Inc., 2025b.

Shashwat Singh, Shauli Ravfogel, Jonathan Herzig, Roee Aharoni, Ryan Cotterell, and Ponnurangam Kumaraguru. Representation surgery: Theory and practice of affine steering. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 45663–45680. PMLR, 2024.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. Improving instruction-following in language models through activation steering. In *International Conference on Learning Representations*, volume 2025, pp. 55790–55823, 2025.

Nishant Subramani, Nivedita Suresh, and Matthew E. Peters. Extracting latent steering vectors from pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 566–581. Association for Computational Linguistics, 2022.

William Timkey and Marten van Schijndel. All bark and no bite: Rogue dimensions in transformer language models obscure representational quality. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 4527–4546. Association for Computational Linguistics, 2021.

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

Minh Hieu Vu and Tan M. Nguyen. Angular steering: Behavior control via rotation in activation space. In *Advances in Neural Information Processing Systems*, volume 38, pp. 121653–121690. Curran Associates, Inc., 2025.

Hanyu Wang, Bochuan Cao, Yuanpu Cao, and Jinghui Chen. TruthFlow: Truthful LLM generation via representation flow correction. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 62423–62444. PMLR, 2025a.

Sihan Wang, Jiayi Zhao, Qingyan Cao, Hongbo Yao, and Lin Shu. FishBack: Pullback Fisher geometry for optimal activation steering in transformers. *arXiv preprint arXiv:2605.17231*, 2026.

Weixuan Wang, Jingyuan Yang, and Wei Peng. Semantics-adaptive activation intervention for LLMs via dynamic steering vectors. In *International Conference on Learning Representations*, volume 2025, pp. 79334–79351, 2025b.

Joong-Ho Won, Johan Lim, Seung-Jean Kim, and Bala Rajaratnam. Condition-number-regularized covariance estimation. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 75(3):427–450, 2013.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D Manning, and Christopher Potts. AxBench: Steering LLMs? Even simple baselines outperform sparse autoencoders. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 67035–67080. PMLR, 2025.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Zejia You, Chunyuan Deng, and Hanjie Chen. Spherical steering: Geometry-aware activation rotation for language models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Hongjue Zhao, Haosen Sun, Jiangtao Kong, Xiaochang Li, Qineng Wang, Liwei Jiang, Qi Zhu, Tarek Abdelzaher, Yejin Choi, Manling Li, and Huajie Shao. ODESteer: A unified ODE-based steering framework for LLM alignment. In *International Conference on Learning Representations*, volume 2026, pp. 69961–69988, 2026.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Complete Derivations

This appendix expands the optimization analysis and implementation details in Section 3. We use $\boldsymbol{x}(t)$ for the activation trajectory, h for the continuously differentiable steering objective, and $\boldsymbol{g}(\boldsymbol{x}) = \nabla h(\boldsymbol{x})$ for its gradient. The population mean and covariance are $\boldsymbol{\mu}$ and $\boldsymbol{C} \succ \mathbf{0}$, respectively, and the Mahalanobis norm is $\|\boldsymbol{v}\|_{\boldsymbol{C}^{-1}} = \sqrt{\boldsymbol{v}^\top \boldsymbol{C}^{-1}\boldsymbol{v}}$. For the practical implementation, we replace $\boldsymbol{C}$ and $\boldsymbol{\mu}$ with the regularized covariance estimate $\widehat{\boldsymbol{C}}$ and the empirical mean $\overline{\boldsymbol{x}}$ defined in Section 3.3. Both are held fixed during an intervention.

A.1 Anisotropy of Euclidean Steering

Proof of Proposition 1. For an activation $\boldsymbol{x}$ with $\boldsymbol{g}(\boldsymbol{x}) \neq \mathbf{0}$, the normalized Euclidean field in Equation (4) has squared Mahalanobis speed

$$\|\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x})\|^2_{\boldsymbol{C}^{-1}} = \frac{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}^{-1}\boldsymbol{g}(\boldsymbol{x})}{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{g}(\boldsymbol{x})}. \tag{16}$$

This is a Rayleigh quotient of $\boldsymbol{C}^{-1}$. Let $c_{\min}$ and $c_{\max}$ be the smallest and largest eigenvalues of $\boldsymbol{C}$, as in the proposition. The eigenvalues of $\boldsymbol{C}^{-1}$ lie in $[1/c_{\max}, 1/c_{\min}]$, so

$$\frac{1}{\sqrt{c_{\max}}} \leq \|\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x})\|_{\boldsymbol{C}^{-1}} \leq \frac{1}{\sqrt{c_{\min}}}. \tag{17}$$

The lower and upper bounds are attained when the gradient lies in the eigenspaces associated with $c_{\max}$ and $c_{\min}$, respectively. Hence, the maximum-to-minimum speed ratio over all directions is $\sqrt{c_{\max}/c_{\min}}$. For a trajectory with nonzero gradients on $[0, T]$, integrating these bounds gives

$$\frac{T}{\sqrt{c_{\max}}} \leq \int_0^T \|\dot{\boldsymbol{x}}(t)\|_{\boldsymbol{C}^{-1}}\, \mathrm{d}t \leq \frac{T}{\sqrt{c_{\min}}}, \tag{18}$$

which proves Equation (6). □

Thus, unit Euclidean speed can produce different intervention strengths relative to activation variation. As the gradient direction changes across activations, the same integration time T can also give different path lengths in the calibrated coordinate system.

A.2 Steepest Ascent under the Mahalanobis Metric

Proof of Proposition 2. Fix an activation $\boldsymbol{x}$ with $\boldsymbol{g}(\boldsymbol{x}) \neq \mathbf{0}$ and a covariance $\boldsymbol{C} \succ \mathbf{0}$. For any feasible velocity $\boldsymbol{v}$ in Equation (7), let $\boldsymbol{u} = \boldsymbol{C}^{-1/2}\boldsymbol{v}$. Then $\|\boldsymbol{u}\|_2 = \|\boldsymbol{v}\|_{\boldsymbol{C}^{-1}} \leq 1$, and Cauchy–Schwarz gives

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{v} &= (\boldsymbol{C}^{1/2}\boldsymbol{g}(\boldsymbol{x}))^\top \boldsymbol{u} \\ &\leq \sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}\, \|\boldsymbol{u}\|_2 \leq \sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}. \end{aligned} \tag{19}$$

Because $\boldsymbol{C}^{1/2}\boldsymbol{g}(\boldsymbol{x}) \neq \mathbf{0}$, equality holds exactly when $\boldsymbol{u} = \boldsymbol{C}^{1/2}\boldsymbol{g}(\boldsymbol{x})/\sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}$. Mapping this velocity back to the activation coordinates gives

$$\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) = \frac{\boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}{\sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}}. \tag{20}$$

This is the unique solution in Equation (8). Substitution gives $\|\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})\|_{\boldsymbol{C}^{-1}} = 1$ and the maximum objective rate $\sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}$. □

When $\boldsymbol{g}(\boldsymbol{x}) = \mathbf{0}$, every feasible velocity gives zero first-order change in the objective. As in Section 3.2, we set $\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) = \mathbf{0}$ at such an activation.

The corresponding unconstrained optimization problem in Proposition 2 follows by completing the square:

$$2\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{v} - \|\boldsymbol{v}\|^2_{\boldsymbol{C}^{-1}} = \boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x}) - \|\boldsymbol{v} - \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})\|^2_{\boldsymbol{C}^{-1}}.$$

Its unique maximizer is $\boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})$. Normalizing this velocity to unit Mahalanobis speed gives Equation (8) when the gradient is nonzero.

A.3 Continuous-Time Objective Ascent

Consider a trajectory following Equation (1) with the TRIS velocity field in Equation (8). At any activation with a nonzero gradient, substitution into Equation (2) gives

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}h(\boldsymbol{x}(t)) &= \boldsymbol{g}(\boldsymbol{x}(t))^\top \boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}(t)) \\ &= \sqrt{\boldsymbol{g}(\boldsymbol{x}(t))^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x}(t))} \geq 0.\end{aligned} \tag{21}$$

At a zero gradient, the velocity and objective rate are both zero, so the same equality holds. Thus, the objective is nondecreasing along the continuous trajectory on its interval of existence and increases wherever its gradient is nonzero. Appendix A.6 gives a sufficient condition for objective ascent with finite Euler steps.

A.4 Isotropic Steering in Calibrated Coordinates

Proof of Proposition 3. Let $\boldsymbol{z} = \boldsymbol{C}^{-1/2}(\boldsymbol{x} - \boldsymbol{\mu})$ and $\widetilde{h}(\boldsymbol{z}) = h(\boldsymbol{\mu} + \boldsymbol{C}^{1/2}\boldsymbol{z})$, as in the proposition. The Euclidean distance between two activations in these coordinates equals their Mahalanobis distance in the original coordinates. Since $\boldsymbol{C}^{1/2}$ is symmetric, the chain rule gives $\widetilde{\boldsymbol{g}}(\boldsymbol{z}) = \nabla_{\boldsymbol{z}}\widetilde{h}(\boldsymbol{z}) = \boldsymbol{C}^{1/2}\boldsymbol{g}(\boldsymbol{x})$. Under $\boldsymbol{v} = \boldsymbol{C}^{1/2}\boldsymbol{u}$, the constraint becomes $\|\boldsymbol{u}\|_2 \leq 1$ and the objective rate becomes $\widetilde{\boldsymbol{g}}(\boldsymbol{z})^\top \boldsymbol{u}$. For a nonzero gradient, the normalized Euclidean ascent field is

$$\boldsymbol{u}_{\mathrm{E}}(\boldsymbol{z}) = \frac{\widetilde{\boldsymbol{g}}(\boldsymbol{z})}{\|\widetilde{\boldsymbol{g}}(\boldsymbol{z})\|_2} = \frac{\boldsymbol{C}^{1/2}\boldsymbol{g}(\boldsymbol{x})}{\sqrt{\boldsymbol{g}(\boldsymbol{x})^\top \boldsymbol{C}\boldsymbol{g}(\boldsymbol{x})}} = \boldsymbol{C}^{-1/2}\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}). \tag{22}$$

Hence, $\dot{\boldsymbol{z}} = \boldsymbol{u}_{\mathrm{E}}(\boldsymbol{z})$ and $\|\boldsymbol{u}_{\mathrm{E}}(\boldsymbol{z})\|_2 = \|\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})\|_{\boldsymbol{C}^{-1}} = 1$ wherever the gradient is nonzero. This establishes isotropic steering in the calibrated coordinate system, independently of the steering direction and current activation. □

In the practical implementation, the same derivation uses $\boldsymbol{z} = \widehat{\boldsymbol{C}}^{-1/2}(\boldsymbol{x} - \overline{\boldsymbol{x}})$, with each coordinate calibrated by the regularized variance in Equation (13).

When $\boldsymbol{C} = \boldsymbol{I}$, the velocity field reduces to

$$\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}) = \frac{\boldsymbol{g}(\boldsymbol{x})}{\|\boldsymbol{g}(\boldsymbol{x})\|_2}, \tag{23}$$

which is $\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x})$ in Equation (4) and recovers ODESteer for the same steering objective. Both fields are set to zero at a zero gradient. If all empirical variances are equal and no smaller than ϵ_s, then $s_j = m$ for every j and Equation (13) gives $\widehat{\boldsymbol{C}} = m\boldsymbol{I}$. The resulting velocity is $\sqrt{m}\,\boldsymbol{v}_{\mathrm{E}}(\boldsymbol{x})$, so one unit of path length in the calibrated coordinates corresponds to $\sqrt{m}$ units in the raw activation space.

A.5 Intervention Strength and Activation Rescaling

For an absolutely continuous TRIS trajectory with fixed $\boldsymbol{C}$, the unit Mahalanobis speed gives

$$\int_0^T \|\dot{\boldsymbol{x}}(t)\|_{\boldsymbol{C}^{-1}}\,\mathrm{d}t \leq T, \tag{24}$$

with equality when the gradient is nonzero for almost every $t \in [0, T]$. The triangle inequality also gives $\|\boldsymbol{x}(T) - \boldsymbol{x}_0\|_{\boldsymbol{C}^{-1}} \leq T$. Thus, the integration time T specifies a path-length budget in the calibrated coordinate system and bounds the displacement from the initial activation. For two times T_1 and T_2 along the same trajectory, the same argument gives $\|\boldsymbol{x}_{T_2} - \boldsymbol{x}_{T_1}\|_{\boldsymbol{C}^{-1}} \leq |T_2 - T_1|$, where $\boldsymbol{x}_T = \boldsymbol{x}(T)$ as in Section 3.2.

We next consider a change of activation units in the practical implementation. Let $\boldsymbol{x}' = a\boldsymbol{x}$ and $\mathcal{X}' = a\mathcal{X}$ for $a > 0$, and assume that the variance floor ϵ_s is inactive both before and after rescaling. The empirical mean becomes $\overline{\boldsymbol{x}}' = a\overline{\boldsymbol{x}}$, while $s'_j = a^2 s_j$ and $m' = a^2 m$. For the same λ and κ, Equation (13) therefore gives

$$\widehat{\boldsymbol{C}}' = a^2\widehat{\boldsymbol{C}}, \qquad (\widehat{\boldsymbol{C}}')^{-1} = a^{-2}\widehat{\boldsymbol{C}}^{-1}. \tag{25}$$

Consequently, the standardized coordinates are unchanged:

$$(\widehat{\boldsymbol{C}}')^{-1/2}(\boldsymbol{x}' - \overline{\boldsymbol{x}}') = \widehat{\boldsymbol{C}}^{-1/2}(\boldsymbol{x} - \overline{\boldsymbol{x}}). \tag{26}$$

Express the same steering objective in the rescaled coordinates as $h'(\boldsymbol{x}') = h(\boldsymbol{x}'/a)$. Its gradient is $\boldsymbol{g}'(\boldsymbol{x}') = a^{-1}\boldsymbol{g}(\boldsymbol{x})$, so the velocity field computed with $\widehat{\boldsymbol{C}}'$ satisfies

$$\boldsymbol{v}'_{\mathrm{M}}(\boldsymbol{x}') = \frac{\widehat{\boldsymbol{C}}'\boldsymbol{g}'(\boldsymbol{x}')}{\sqrt{\boldsymbol{g}'(\boldsymbol{x}')^{\top}\widehat{\boldsymbol{C}}'\boldsymbol{g}'(\boldsymbol{x}')}} = a\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}). \tag{27}$$

The equality also holds at a zero gradient under the zero-velocity convention. Hence, if $\boldsymbol{x}(t)$ solves the original flow, then $\boldsymbol{x}'(t) = a\boldsymbol{x}(t)$ solves the rescaled flow with the same integration time T.

For K Euler steps, this relation holds at each update:

$$\frac{T}{K}\boldsymbol{v}'_{\mathrm{M}}(\boldsymbol{x}') = a\frac{T}{K}\boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x}). \tag{28}$$

Starting from $\boldsymbol{x}'_0 = a\boldsymbol{x}_0$, induction gives $\boldsymbol{x}'_k = a\boldsymbol{x}_k$ for every $k = 0, \ldots, K$. Thus, under the stated assumptions, the same T and K produce corresponding interventions after a change of activation units.

A.6 Objective Ascent with Finite Euler Steps

Consider an Euler step of size $\alpha = T/K > 0$ at an activation $\boldsymbol{x}$ with $\boldsymbol{g} = \boldsymbol{g}(\boldsymbol{x}) \neq \boldsymbol{0}$. Let $\boldsymbol{v} = \boldsymbol{v}_{\mathrm{M}}(\boldsymbol{x})$ be the normalized velocity in Equation (8), using $\widehat{\boldsymbol{C}}$ from Equation (13) and an inactive denominator floor. Suppose that h has an L-Lipschitz Euclidean gradient, with $L > 0$, on the segment from $\boldsymbol{x}$ to $\boldsymbol{x} + \alpha\boldsymbol{v}$. The smoothness inequality gives

$$h(\boldsymbol{x} + \alpha\boldsymbol{v}) \geq h(\boldsymbol{x}) + \alpha\boldsymbol{g}^{\top}\boldsymbol{v} - \frac{L\alpha^2}{2}\|\boldsymbol{v}\|_2^2. \tag{29}$$

The estimator satisfies $\lambda m \leq \widehat{s}_j \leq \kappa m$ for every coordinate, so $\widehat{\boldsymbol{C}}$ is positive definite and $\mathrm{cond}(\widehat{\boldsymbol{C}}) \leq \kappa/\lambda$. In particular, the upper cap gives $\widehat{\boldsymbol{C}} \preceq \kappa m\boldsymbol{I}$ and therefore $\widehat{\boldsymbol{C}}^{-1} \succeq (\kappa m)^{-1}\boldsymbol{I}$. Together with the unit Mahalanobis speed, this gives $1 = \boldsymbol{v}^{\top}\widehat{\boldsymbol{C}}^{-1}\boldsymbol{v} \geq (\kappa m)^{-1}\|\boldsymbol{v}\|_2^2$. Substituting $\|\boldsymbol{v}\|_2^2 \leq \kappa m$ and $\boldsymbol{g}^{\top}\boldsymbol{v} = \sqrt{\boldsymbol{g}^{\top}\widehat{\boldsymbol{C}}\boldsymbol{g}}$ into Equation (29) yields

$$h(\boldsymbol{x} + \alpha\boldsymbol{v}) - h(\boldsymbol{x}) \geq \alpha\sqrt{\boldsymbol{g}^{\top}\widehat{\boldsymbol{C}}\boldsymbol{g}} - \frac{L\kappa m\alpha^2}{2}. \tag{30}$$

The objective therefore increases whenever $0 < T/K < 2\sqrt{\boldsymbol{g}^{\top}\widehat{\boldsymbol{C}}\boldsymbol{g}}/(L\kappa m)$. This sufficient condition relates the integration resolution K to the integration time T, the local gradient, and the curvature of the objective. We record the objective change at each Euler step to assess the realized ascent.

A.7 Numerical Stabilization

Algorithm 1 reuses the fitted objective h and covariance estimate $\widehat{\boldsymbol{C}}$ during generation. At Euler step k, let $\boldsymbol{g}_k = \nabla h(\boldsymbol{x}_k)$. The denominator floor $\epsilon_v > 0$ gives the stabilized velocity

$$\boldsymbol{v}_k = \frac{\widehat{\boldsymbol{C}}\boldsymbol{g}_k}{\max\{\sqrt{\boldsymbol{g}_k^{\top}\widehat{\boldsymbol{C}}\boldsymbol{g}_k}, \epsilon_v\}},$$

which is zero when $\boldsymbol{g}_k = \boldsymbol{0}$. For any gradient, its Mahalanobis speed satisfies

$$\|\boldsymbol{v}_k\|_{\widehat{\boldsymbol{C}}^{-1}} = \frac{\sqrt{\boldsymbol{g}_k^{\top}\widehat{\boldsymbol{C}}\boldsymbol{g}_k}}{\max\{\sqrt{\boldsymbol{g}_k^{\top}\widehat{\boldsymbol{C}}\boldsymbol{g}_k}, \epsilon_v\}} \leq 1. \tag{31}$$

With $\alpha = T/K$ and $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha \boldsymbol{v}_k$, the accumulated step length therefore obeys

$$\sum_{k=0}^{K-1} \|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|_{\widehat{C}^{-1}} = \alpha \sum_{k=0}^{K-1} \|\boldsymbol{v}_k\|_{\widehat{C}^{-1}} \leq T.$$

Thus, the stabilized updates preserve the path-length budget. When the denominator floor is inactive, they coincide with the normalized updates analyzed above and satisfy the rescaling relation under the assumptions of Appendix A.5. As in Section 3.3, we evaluate activations, gradients, and velocities in FP32 and restore the model's original precision after the final step. We also check intermediate quantities for finite values.

B Experimental Details

B.1 Models, Activations, and Intervention Layers

Table 4: **Model architectures and intervention layers.** Layer indices are zero-based, with one-based indices in parentheses.

Model	Layers	Hidden dimension	Intervention layer
Mistral-7B	32	4096	15 (16)
Llama-3.1-8B	32	4096	13 (14)
Qwen3-8B	36	4096	18 (19)
Qwen3-14B	40	5120	24 (25)

Following ODESteer (Zhao et al., 2026), we run CAA with strength $T = 1$ at every layer on TruthfulQA and select the layer with the highest Truth×Info score. We break ties by choosing the shallowest layer and share it across all methods and tasks.

We fit steering components from residual-stream activations at the last input token and intervene at the same layer on each newly generated token.

B.2 Datasets and Splits

Truthfulness. All methods share two folds of 408 and 409 TruthfulQA questions (Lin et al., 2022). We fit on one fold and evaluate on the other, then reverse their roles to cover all 817 questions. For Llama-3.1-8B, the two fitting folds contain 1,311/1,672 and 1,289/1,646 positive/negative activations, respectively.

Helpfulness. All methods use the same 10,000 UltraFeedback preference pairs for fitting and 500 prompts in the same order for evaluation (Cui et al., 2024).

Detoxification. We fit all steering components on 5,000 toxic and 5,000 benign examples sampled from the Jigsaw training split. We evaluate on a fixed set of 500 toxic prompts from RealToxicityPrompts (Gehman et al., 2020).

Fitting and parameter selection. We use the baseline settings from ODESteer (Zhao et al., 2026). ODESteer and TRIS use $K = 20$ Euler steps and evaluate every integer $T \in [1, 50]$ for each model and task. We report the settings with the highest Truth×Info and win rate, and the lowest Toxic Rate with PPL ≤ 30.