WHEN ONE DIRECTION IS NOT ENOUGH: GRASP FOR CONDITIONAL ACTIVATION STEERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Activation steering provides a lightweight way to control LLMs without updating model parameters. However, most existing methods represent a target behavior with a single global direction, implicitly assuming that the same intervention is suitable for every input associated with that behavior. Analysis shows that behavior gradients can be low-dimensional yet multi-directional: different inputs may require distinct, sometimes opposing, steering directions. Averaging these directions into a single global vector may therefore miss useful control signals. To preserve this input-dependent structure, we propose **G**radient-**R**outed **A**ctivation **S**ubspace **P**rototypes (**GRASP**), a conditional activation steering framework that replaces a single global direction with a small set of representative directions. GRASP first projects calibration gradients into a low-rank subspace and clusters their normalized directions to construct gradient prototypes. It then learns a lightweight router that uses the current input activation to select which prototype to apply at inference time. Across seven behavioral steering tasks and two LLM backbones, GRASP improves seven-task macro accuracy over the unmodified Qwen2.5-7B and Llama-3.1-8B models by 13.37 and 20.23 percentage points, respectively. Ablation studies show that two prototypes outperform a shared direction, while routing controls confirm the need for input-dependent selection.

1 INTRODUCTION

Activation steering has emerged as a lightweight approach for controlling the behavior of large language models (LLMs) without updating their parameters. Given a set of calibration examples, existing methods typically identify a behavior-relevant direction in the model's hidden representation space and add this direction to intermediate activations during inference, thereby biasing subsequent generation toward a desired behavioral target (Li et al., 2023; Rimsky et al., 2024; Cao et al., 2024; Rodriguez et al., 2025). This simple intervention provides an efficient alternative to parameter fine-tuning and enables behavioral control while keeping the underlying model frozen. However, most activation steering methods compress information from many calibration examples into a single global direction, implicitly assuming that inputs associated with the same behavioral target can be effectively controlled by the same intervention. Recent evidence that steering vectors can behave inconsistently across prompts and models raises an important question: *does one global direction adequately represent the intervention required by every input?* (Tan et al., 2024; Da Silva et al., 2025)

Significant efforts have been devoted to improving activation steering, including better direction estimation, conditional intervention, and more structured representation control. Specifically, directional steering methods such as CAA and BiPO derive a behavior-relevant vector from contrastive activations or preference supervision and apply it to hidden representations during inference (Rimsky et al., 2024; Cao et al., 2024). More recent approaches improve how such interventions are constructed or applied: LinEAS learns activation transformations through a distributional objective, while Angular Steering rotates representations within a feature plane (Rodriguez et al., 2025; Vu & Nguyen, 2025). Beyond a fixed intervention rule, conditional methods such as CAST and K-CAST use input-dependent signals to determine whether, or how, a predefined steering direction should be applied (Lee et al., 2025; Valentino et al., 2026). Nevertheless, these approaches largely retain a common control abstraction: the target behavior is represented by one principal direction or a predefined intervention axis. This abstraction can be restrictive when different inputs associated with the

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.

same behavioral objective require distinct steering directions. In such cases, compressing heterogeneous directional information into a single vector may weaken the intervention through directional cancellation, limiting the flexibility of existing steering methods.

We observe that behavior-specific calibration gradients can exhibit a compact yet multi-directional structure: although they are largely concentrated in a low-dimensional subspace, different inputs may correspond to distinct, and sometimes opposing, steering directions. This suggests that the behavioral objective itself may already contain multiple useful directions for input-dependent control. Motivated by this observation, we argue that improving activation steering may not always require estimating a stronger global vector or introducing more complex inference-time mechanisms. Instead, the key may lie in preserving the directional structure revealed during calibration and learning which direction is appropriate from the model's own internal activations. If these directions can be predicted from forward activations, gradients are needed only during calibration, while deployment can remain lightweight and keep the language model frozen. This raises a natural question: *Can multiple behavior-relevant directions be discovered from calibration gradients and reliably selected for individual inputs at inference time, without requiring labels or gradient computation?*

Prior studies have investigated activation representations from the perspectives of behavioral steering, interpretability, and conditional control (Rimsky et al., 2024; Arditi et al., 2024; Lee et al., 2025; Valentino et al., 2026). However, relatively little attention has been paid to the directional structure of objective gradients across individual calibration inputs. Existing steering methods typically aggregate calibration signals into a single global direction, while conditional approaches mainly decide whether or how a predefined intervention should be applied. Such formulations may overlook the possibility that examples sharing the same behavioral objective induce systematically different steering directions. Motivated by this observation, we conduct a fine-grained analysis of calibration gradients and find that they can be strongly concentrated in a low-dimensional subspace while still forming distinct, and sometimes opposing, directional modes. These results suggest that the main limitation is not necessarily the absence of a compact behavioral representation, but the loss of input-dependent directional structure when multiple modes are collapsed into a single steering vector.

By analyzing the directional structure of calibration gradients, we identify multiple behavior-relevant modes that would be obscured by collapsing all examples into a single global steering vector. Building on this observation, we propose Gradient-Routed Activation Subspace Prototypes (GRASP), a lightweight conditional steering framework that preserves these distinct directions and selects among them according to the current input. Specifically, GRASP first projects calibration gradients into a low-rank subspace, clusters their normalized directions to construct a small set of gradient prototypes, and then trains an activation-based router to associate input representations with the corresponding prototype. At inference time, the router selects a cached steering direction using only forward activations, which is then injected into the model residual stream to guide generation. The language model remains completely frozen, and deployment requires neither labels, gradient computation, nor parameter updates. This two-stage design, gradient-prototype discovery during calibration and activation-based routing during inference, enables input-dependent steering while retaining the efficiency and simplicity of activation intervention. Finally, we summarize our main contributions as follows:

- We reveal a previously underexplored structure in activation steering: behavior-specific calibration gradients can be strongly low-dimensional while remaining multi-directional, forming distinct and often opposing modes. It shows that collapsing heterogeneous gradients into a single global direction can suppress useful input-dependent steering signals.
- We propose Gradient-Routed Activation Subspace Prototypes (GRASP), a lightweight conditional steering framework that preserves these directional modes as gradient prototypes and learns to select among them from forward activations. GRASP keeps the backbone frozen and requires neither labels, gradients, nor parameter updates at inference time.
- We evaluate GRASP across seven behavioral steering tasks and two LLM backbones, obtaining macro-accuracy gains of 13.37 and 20.23 percentage points over the respective unmodified models. Prototype-count ablations and routing controls further show that effective steering depends on both preserving multiple directions and selecting them conditionally.

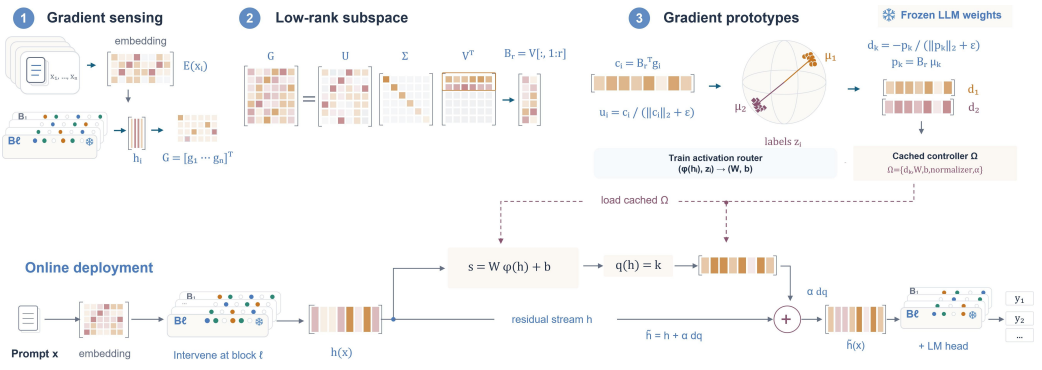

Figure 1: **Framework of GRASP.** During offline calibration, the language model remains frozen while per-example gradients at intervention block ℓ are projected into a rank-r subspace and clustered into prototype directions. Gradient-derived assignments z_i supervise an activation router, which maps the corresponding activations to prototype identities; the router and normalized directions are then cached. At online deployment, the router selects direction $d_{q(h)}$ from the current activation and adds $\alpha d_{q(h)}$ to the residual stream at block ℓ. Thus, the backbone parameters remain frozen throughout, and inference requires neither labels nor gradients.

2 Related Work

2.1 Activation Steering and Conditional Control

Activation steering. Activation steering controls language-model behavior by intervening directly on intermediate representations while keeping the underlying model parameters fixed. Early and representative approaches typically construct behavior-relevant directions from model activations or preference supervision. CAA derives a steering vector from contrastive activation differences, while BiPO learns a behavior direction from preference pairs (Rimsky et al., 2024; Cao et al., 2024). Subsequent work has explored richer forms of representation intervention. LinEAS learns activation transformations through a distributional objective, whereas Angular Steering modifies representations by rotating them within a feature plane (Rodriguez et al., 2025; Vu & Nguyen, 2025). Other studies identify refusal-related features, reshape safety representations, or distribute steering signals across features and decoding positions (Arditi et al., 2024; Yousefpour et al., 2025; Roller et al., 2021b; Klerings et al., 2025). Despite their different estimation and intervention mechanisms, these approaches primarily seek a behavior-level control object that can be reused across inputs. Such a formulation is effective when examples associated with the same behavioral objective admit a common intervention, but can become restrictive when their preferred steering directions differ.

Conditional control. More recent methods introduce input dependence into activation steering. CAST uses an activation-derived condition to determine whether a predefined behavior vector should be applied, while K-CAST retrieves similar conditions to determine the sign or parameters of a shared contrastive intervention (Lee et al., 2025; Valentino et al., 2026). AlphaSteer constrains steering to protect benign-input representations, and PID Steering adjusts intervention strength through closed-loop feedback (Sheng et al., 2026; Nguyen et al., 2026). These studies demonstrate that the appropriate intervention can depend on the current input. However, conditionality is generally introduced around an already defined steering axis or intervention rule. In contrast, GRASP asks a complementary question: whether the behavioral objective itself contains multiple useful directions that should first be discovered and preserved, and then selected conditionally for individual inputs.

2.2 Internal Representations and Objective-Driven Intervention

Internal representations. A broad line of work has shown that intermediate representations encode linguistic, semantic, factual, and alignment-related information. Probing methods characterize information contained in contextual representations (Tenney et al., 2019; Conia & Navigli, 2022; Immer et al., 2022), while concept erasure and model-editing studies manipulate internal features to

establish their relationship with model behavior (Arditi et al., 2024; Meng et al., 2022). Mechanistic analyses further identify representation-level structures associated with alignment and behavioral tendencies (Lee et al., 2024). These studies motivate the use of internal states as control signals, but identifying information in a representation does not by itself determine which intervention direction should be applied to a particular input.

Objective-driven intervention. Task objectives and gradients provide a complementary source of information for controlling model behavior. Gradient-based sampling, expert decoding, and activation-calibration methods use task-level signals to guide generation or representation modification (Kumar et al., 2022; Liu et al., 2021; 2024), while distribution-wise interventions learn broader transformations over representation regions (Deng et al., 2025). Sparse expert architectures similarly route inputs toward different computational components according to their internal representations (Roller et al., 2021a; Lewis et al., 2021; Zhou et al., 2022). GRASP combines these perspectives in a different way: gradients are used only during calibration to expose the directional structure of a behavioral objective, whereas deployment uses a lightweight activation-based router to select among a small set of cached steering directions. Thus, no gradient computation, parameter update, or expert network is required at inference time.

3 GRADIENT-ROUTED ACTIVATION SUBSPACE PROTOTYPES

This section presents GRASP, which preserves multiple behavior-relevant directions instead of compressing them into a single global vector. GRASP therefore operates in two stages. During *offline calibration*, it extracts a low-rank gradient subspace, discovers a small set of gradient prototypes, and learns an activation-based router. During *online deployment*, the cached router selects one prototype from the current input activation and injects it into the frozen model.

3.1 DISCOVERING BEHAVIOR-SPECIFIC GRADIENT PROTOTYPES

We first identify the directional structure induced by a behavioral objective. Given calibration examples, we compute how the objective changes with respect to the model representation at a selected intervention block, compress these gradients into a low-rank subspace, and then separate distinct directional modes within that subspace.

Behavioral gradients. Let $h_i \in \mathbb{R}^d$ denote the residual representation of the final prompt token for calibration example x_i at the selected intervention block. The language model is kept frozen throughout calibration. Let a_i^+ denote the candidate token associated with the desired behavior and a_i^- its alternative. We define the behavioral loss and its representation gradient as

$$\mathcal{L}_i(h_i) = logit(a_i^- \mid h_i, \theta) - logit(a_i^+ \mid h_i, \theta), \qquad g_i = \nabla_{h_i} \mathcal{L}_i(h_i), \tag{1}$$

where $logit(\cdot \mid h_i, \theta)$ is the corresponding next-token logit produced by the remaining frozen layers. Minimizing $\mathcal{L}_i$ increases the target logit margin, so $-g_i$ locally indicates a behavior-improving direction.

Constructing a low-rank gradient subspace. We stack the calibration gradients into $G = [g_1^\top; \ldots; g_n^\top] \in \mathbb{R}^{n \times d}$ and perform an uncentered thin singular value decomposition:

$$G = U \Sigma V^\top, \qquad B_r = V_{[:,1:r]}, \qquad E_r = \frac{\sum_{j=1}^{r} \sigma_j^2}{\sum_j \sigma_j^2}, \tag{2}$$

where $B_r \in \mathbb{R}^{d \times r}$ contains the leading r right-singular vectors and E_r measures the retained gradient energy. A high E_r indicates that the gradients are geometrically compact, but it does not imply that they all point in the same direction.

To expose directional variation inside this subspace, we project each gradient and normalize its direction:

$$c_i = B_r^\top g_i, \qquad w_i = \|c_i\|_2, \qquad u_i = \frac{c_i}{\max(w_i, \epsilon)}, \tag{3}$$

where $c_i \in \mathbb{R}^r$ enotes the low-dimensional gradient coordinate and u_i is its unit direction. Clustering is performed on u_i rather than c_i making the prototypes invariant to gradient magnitude and focused on directional structure.

Discovering gradient modes. We partition the normalized directions using K-means:

$$\min_{\{m_k\},\{z_i\}} \sum_{i=1}^{n} \|u_i - m_{z_i}\|_2^2, \qquad z_i \in \{1,\dots,K\}, \tag{4}$$

where z_i denotes the mode assigned to example i. Each fitted center is normalized and mapped back from the low-rank coordinates to the original activation space:

$$\mu_k = \frac{m_k}{\max(\|m_k\|_2, \epsilon)}, \qquad d_k = -\frac{B_r \mu_k}{\max(\|B_r \mu_k\|_2, \epsilon)}. \tag{5}$$

The resulting $d_k \in \mathbb{R}^d$ is the k-th *gradient prototype*. The negative sign follows the descent direction of Eq. 1. When $K = 1$, GRASP reduces to a single shared steering direction; for $K > 1$, it preserves several behavior-relevant directions that would otherwise be averaged together.

Why can one shared direction be insufficient? The fitted partition also provides a simple first-order view of directional cancellation. Let $\mathcal{C}_k = \{i : z_i = k\}$, $s_k = \sum_{i \in \mathcal{C}_k} c_i$, and $s = \sum_k s_k$. For a fixed intervention strength α, the best first-order improvement attainable by one shared direction and by one direction per cluster is respectively

$$G_1^\star(\alpha) = \frac{\alpha}{n} \|s\|_2, \qquad G_K^\star(\alpha) = \frac{\alpha}{n} \sum_{k=1}^{K} \|s_k\|_2. \tag{6}$$

Their difference defines the *conditionality gap*:

$$\Delta_{\text{cond}} = \frac{1}{n} \left(\sum_{k=1}^{K} \|s_k\|_2 - \|s\|_2 \right) \geq 0, \qquad G_K^\star - G_1^\star = \alpha \Delta_{\text{cond}}. \tag{7}$$

The inequality follows directly from the triangle inequality. If different cluster resultants point in opposing directions, $\|s\|_2$ can be substantially smaller than $\sum_k \|s_k\|_2$, even though all gradients lie in the same low-dimensional subspace. Thus, *low-dimensional does not necessarily mean single-directional*: averaging can suppress useful input-specific control signals.

Importantly, Eq. 7 is a diagnostic of the fitted gradient geometry rather than a guarantee of downstream accuracy. GRASP must still determine which prototype is appropriate for a new input without observing its gradient.

3.2 Activation-Routed Steering

Once the gradient prototypes are constructed, the remaining problem is to predict the appropriate prototype from information available at inference time. GRASP therefore learns a lightweight router from calibration activations to their gradient-mode assignments.

Learning an activation router. Let $\phi(h_i)$ denote the standardized last-token activation at the output of the intervention block. Using the cluster assignments z_i obtained in Eq. 4, we train a regularized linear classifier:

$$\begin{aligned} \pi(h) &= \text{softmax}(W\phi(h) + b), \\ (W^\star, b^\star) &= \arg\min_{W,b} \left[-\sum_{i=1}^{n} \log \pi_{z_i}(h_i) + \frac{\lambda}{2} \|W\|_F^2 \right]. \end{aligned} \tag{8}$$

This converts the gradient-derived modes available during calibration into an activation-based decision rule that can be evaluated with a standard forward pass.

Routing and residual intervention. At deployment, GRASP predicts the prototype index and applies the corresponding cached direction:

$$q(h) = \arg\max_k \pi_k(h), \qquad \widetilde{h} = h + \alpha d_{q(h)}, \tag{9}$$

where α controls intervention strength. The remaining transformer layers then continue the forward computation normally. All prototypes, router parameters, normalization statistics, and the selected α are cached after calibration. Consequently, test-time inference requires only the current activation and introduces no gradient computation or parameter update.

Local interpretation of prototype selection. The usefulness of routing can also be understood from a local smoothness argument. Suppose $\mathcal{L}$ has an H-Lipschitz gradient along the intervention segment. For a nonzero projected gradient with $w = \left\|B_r^\top g\right\|_2$ and $u = B_r^\top g/w$, applying prototype d_k yields

$$\mathcal{L}(h + \alpha d_k) - \mathcal{L}(h) \leq -\alpha w\, u^\top \mu_k + \frac{H\alpha^2}{2}. \tag{10}$$

The first-order term depends on the alignment $u^\top \mu_k$ between the current gradient direction and the selected prototype. Hence, retaining a low-rank subspace alone is insufficient: successful conditional steering also requires selecting a prototype aligned with the current input. GRASP approximates this otherwise gradient-dependent choice using the activation router in Eq. 8.

In summary, GRASP separates gradient information from its deployment mechanism. Gradients are used only once during calibration to discover behavior-relevant directional modes, whereas inference relies solely on cached prototypes and forward activations. This design preserves input-dependent directional structure while retaining the frozen-backbone, gradient-free deployment advantages of activation steering.

4 Experimental Evaluation

4.1 Experimental Setup

Models and baselines. We conduct experiments in two backbones, Qwen2.5-7B and Llama-3.1-8B. Unless otherwise specified, GRASP uses a gradient subspace of rank $r = 8$, $K = 2$ gradient prototypes, and applies the intervention to the last-token residual-stream representation at the output of transformer block 20 for Qwen and block 23 for Llama (zero-based indexing). We compare against representative inference-time activation steering methods: CAA (Rimsky et al., 2024), BiPO (Cao et al., 2024), LinEAS (Rodriguez et al., 2025), CAST (Lee et al., 2025), K-CAST (Valentino et al., 2026), Spherical Steering (You et al., 2026), K-Steering (Oozeer et al., 2025) and Angular Steering (Vu & Nguyen, 2025).

Evaluation protocol. We evaluate GRASP on seven behavioral steering tasks from the CAA benchmark: AI Coordination, Corrigibility, Hallucination, Myopic Reward, Survival Instinct, Sycophancy, and Refusal. Main comparisons are reported over five calibration seeds $\{42, 123, 456, 789, 2026\}$. Unless otherwise stated, the ablation, mechanism, and sensitivity analyses use the three-seed diagnostic protocol $\{42, 123, 456\}$. We report binary-choice accuracy, with fit data used to construct the gradient subspace, prototypes, and router, tuning data used only for intervention-strength selection, and test data reserved for evaluation. All experiments were performed using a single NVIDIA A100 GPU.

4.2 Main Results

Table 1 shows that GRASP achieves the highest seven-task macro accuracy on both backbones, reaching 71.94% on Qwen2.5-7B and 84.23% on Llama-3.1-8B. These results correspond to gains of 13.37 and 20.23 percentage points over the respective Base models, and improvements of 5.89 and 13.49 points over the strongest non-GRASP baseline by seven-task macro accuracy. GRASP obtains the best mean accuracy in 10 of the 14 backbone–behavior settings and ranks second in the remaining four. Individual baselines remain stronger on several behaviors: Angular Steering leads on Qwen AI Coordination, BiPO on Qwen Corrigibility, and LinEAS on Qwen Hallucination and Llama Survival Instinct. Overall, the released-split results show that GRASP maintains strong performance across the tested behaviors and backbones.

To better understand the source of the gains in Table 1, we next examine the structure of the calibration gradients. As shown in Figure 2(a), a rank-8 subspace already retains 81.9–92.1% of the gradient energy across all seven behaviors. Importantly, this compact representation does not collapse into a single direction: Table 2 shows consistently negative cosine similarities between the two fitted prototypes (-0.887 to -0.395), while the smaller cluster still contains 38.0–47.9% of the fit examples. Thus, the fitted $K = 2$ partitions yield two consistently negatively aligned and well-populated directional modes. This structure is also reflected in steering performance. Figure 2(b) shows that rank 8 yields strong gains across all seven behaviors and the highest equal-weight mean

Table 1: Main Results. Binary-choice accuracy (%, mean ± sample SD over five random seeds). Bold and underline mark the best and second best mean within each backbone and behavior, respectively.

Method	AI Coordination	Corrigibility	Hallucination	Myopic Reward	Survival Instinct	Sycophancy	Refusal
Qwen2.5-7B	18.00	74.00	60.00	60.00	46.00	74.00	78.00
+ CAA (Rimsky et al., 2024)	18.80±1.10	72.80±1.79	59.20±1.10	61.60±1.67	47.60±0.89	74.00±0.00	79.20±1.10
+ BiPO (Cao et al., 2024)	36.80±8.32	**81.20±5.59**	62.80±3.03	73.20±7.69	51.20±5.22	74.40±0.89	82.80±1.10
+ LinEAS (Rodriguez et al., 2025)	22.80±3.35	74.40±3.85	**73.60±2.97**	69.60±5.18	49.20±2.28	74.00±1.41	86.40±4.34
+ Angular (Vu & Nguyen, 2025)	**55.60±9.42**	70.40±4.56	51.60±9.84	73.20±12.46	52.00±5.48	72.40±2.61	82.80±1.10
+ CAST (Lee et al., 2025)	18.80±1.10	78.40±1.67	61.60±0.89	63.60±2.61	46.40±0.89	74.00±0.00	80.00±0.00
+ K-CAST (Valentino et al., 2026)	19.60±1.67	75.60±0.89	60.80±1.10	62.80±3.03	46.80±1.10	74.00±0.00	78.00±0.00
+ Spherical Steering (You et al., 2026)	18.00±0.00	77.60±1.67	59.60±0.89	60.00±0.00	47.20±1.10	75.60±0.89	77.60±0.89
+ K-Steering (Oozeer et al., 2025)	18.00±0.00	74.00±0.00	60.00±0.00	60.00±0.00	46.00±0.00	74.00±0.00	78.00±0.00
+ **GRASP (Ours)**	43.20±5.02	80.80±2.28	69.20±1.10	**88.80±1.79**	**54.40±8.17**	**76.80±2.28**	**90.40±1.67**
Llama-3.1-8B	38.00	80.00	68.00	56.00	68.00	74.00	64.00
+ CAA (Rimsky et al., 2024)	57.20±2.28	85.60±1.67	81.60±0.89	57.60±2.61	67.60±2.19	70.80±2.68	74.80±1.10
+ BiPO (Cao et al., 2024)	44.80±5.40	72.80±11.01	71.20±3.03	50.80±2.68	66.00±0.00	78.00±6.32	57.60±8.76
+ LinEAS (Rodriguez et al., 2025)	66.00±8.25	85.60±12.03	81.20±13.54	53.20±7.82	**73.20±2.28**	70.00±3.16	60.00±5.83
+ Angular (Vu & Nguyen, 2025)	57.20±7.16	88.40±2.19	70.00±2.83	68.80±9.65	66.80±2.28	71.60±0.89	69.20±10.64
+ CAST (Lee et al., 2025)	44.40±6.23	79.20±3.35	68.80±1.10	54.80±2.68	66.00±3.74	69.60±4.34	64.40±0.89
+ K-CAST (Valentino et al., 2026)	41.20±5.22	80.40±0.89	68.00±0.00	55.20±1.10	67.60±0.89	72.80±1.79	64.00±0.00
+ Spherical Steering (You et al., 2026)	42.00±0.00	80.80±1.10	74.80±1.79	54.80±1.79	65.60±0.89	70.40±0.89	64.40±0.89
+ K-Steering (Oozeer et al., 2025)	40.00±3.16	80.80±1.79	68.00±0.00	55.60±0.89	67.20±1.79	74.00±0.00	64.00±0.00
+ **GRASP (Ours)**	**79.60±5.55**	**90.40±2.61**	**92.80±3.35**	**88.80±7.29**	68.40±8.41	**84.00±3.74**	**85.60±2.61**

Table 2: Geometry of the two fitted gradient prototypes. Qwen2.5-7B fit-split diagnostics at block 20, shown as mean ± sample SD over three calibration seeds. Negative cosine indicates opposing directions, while a smaller-cluster share near 50% indicates two comparably sized modes.

Geometry	Corrigibility	Hallucination	Sycophancy	Refusal	AI Coordination	Myopic Reward	Survival Instinct
Prototype cosine	−0.608±0.015	−0.885±0.036	−0.395±0.062	−0.589±0.136	−0.582±0.053	−0.771±0.007	−0.887±0.054
Smaller-cluster (%)	43.75±5.41	45.31±4.13	45.31±1.56	47.92±0.90	44.79±7.71	46.35±1.80	38.02±2.39

improvement (+14.7 pp), while higher ranks provide little additional benefit. Together, these results suggest that the behavior-relevant gradient geometry is compact yet inherently multi-directional, motivating conditional prototype routing rather than a single global steering direction.

4.3 Ablation and Mechanism Analysis

Having established that the gradient subspace is compact yet multi-directional, we next examine how many prototypes are needed to exploit this structure. Table 3 shows that replacing a single shared direction ($K = 1$) with two prototypes ($K = 2$) improves all seven behaviors, increasing the equal-weight mean gain from +5.05 to +14.67 percentage points. Increasing the prototype count further to $K = 4$ provides no additional overall benefit: the mean gain slightly decreases to +14.10 points. At the behavior level, $K = 4$ matches $K = 2$ on Hallucination, Sycophancy, and Myopic Reward, improves Survival Instinct by 2.67 points, but performs worse on the remaining three behaviors. For the tested settings, $K = 2$ captures most of the available gain, while increasing the prototype count to $K = 4$ provides no further improvement in the equal-weight mean.

Complementary routing controls on Corrigibility, Hallucination, Sycophancy, and Refusal isolate the role of prototype selection (Appendix Table B3). Random and shuffled routing achieve -0.67 and 0.00 pp macro gain over Base, respectively, compared with $+8.17$ pp for GRASP. The adapted K-CAST routing control reaches only $+0.50$ pp, while the non-deployable gradient oracle reaches $+10.83$ pp. These results indicate that preserving multiple directions is not sufficient by itself; the input-dependent correspondence between activations and prototypes is a key part of the gain.

We next test whether forward activations are sufficient to select among the gradient-derived prototypes at inference time. As shown in Figure 3(a), the learned linear router achieves 87.40% equal-weight balanced accuracy across the seven behaviors, compared with 79.68%, 80.31%, and 80.42% for activation-centroid, cosine 1-NN, and cosine 5-NN routing, respectively. Although simpler routers perform better on a few individual behaviors, the learned router provides the strongest overall route prediction. Across 420 tuning decisions, it agrees with the gradient-nearest prototype

Table 3: **Prototype-count ablation on Qwen2.5-7B.** Mean accuracy gain over Base (pp) across three calibration seeds, with rank fixed to 8 and intervention block fixed to 20.

Controller	Corrigibility	Hallucination	Sycophancy	Refusal	AI Coordination	Myopic Reward	Survival Instinct
$K = 1$ (shared)	+10.00	+0.00	+0.67	+0.00	+9.33	+12.00	+3.33
$K = 2$ (GRASP)	**+12.67**	**+8.00**	**+4.00**	**+8.00**	**+28.00**	**+28.67**	+13.33
$K = 4$	+10.00	**+8.00**	**+4.00**	+4.67	+27.33	**+28.67**	**+16.00**

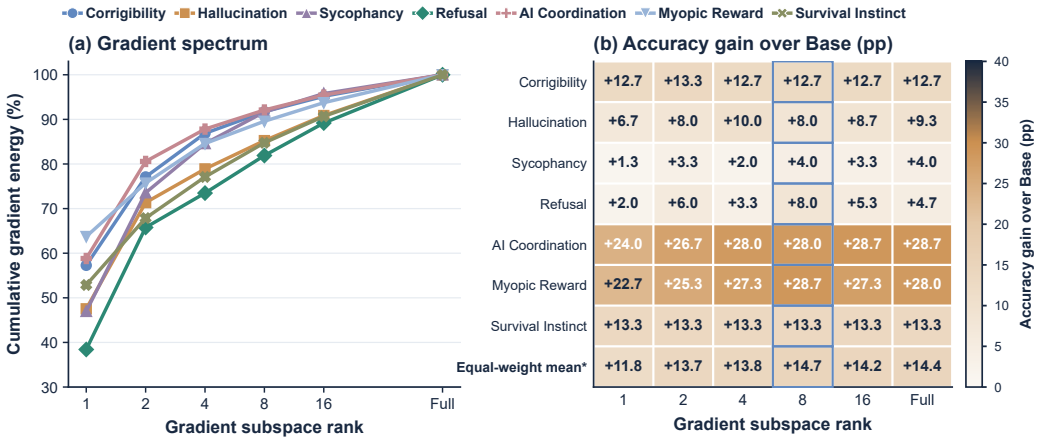

Figure 2: **Low-rank gradients preserve useful steering structure.** Qwen2.5-7B results at block 20. (a) Cumulative gradient energy across subspace ranks, averaged over three calibration seeds. (b) Accuracy gain over Base with $K = 2$ across ranks; rank 8 is the default setting. The last row reports the equal-weight mean across seven behaviors.

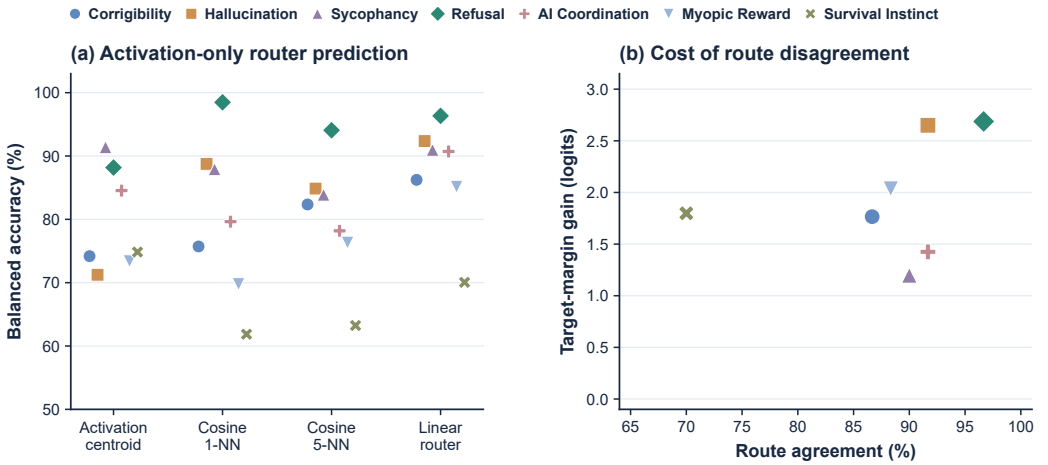

Figure 3: **Forward activations recover gradient-derived routes.** Qwen2.5-7B tuning-set diagnostics over three calibration seeds. (a) Balanced accuracy of activation-based routing methods across seven behaviors. (b) Agreement with the gradient-nearest prototype versus the target-margin gain obtained by correcting observed route disagreements. The gradient-nearest route is used only as a diagnostic oracle.

in 87.86% of cases, with the largest drop observed for Survival Instinct (70.0%). Figure 3(b) further shows that these routing errors have a measurable intervention cost: across 51 observed disagreements, replacing the learned route with the gradient-nearest prototype increases the target margin by 1.84 logits on average. Appendix Fig. B1 additionally shows a mean binary-choice NLL reduction of 0.745 nats after the same correction. Together, these results show that forward activations

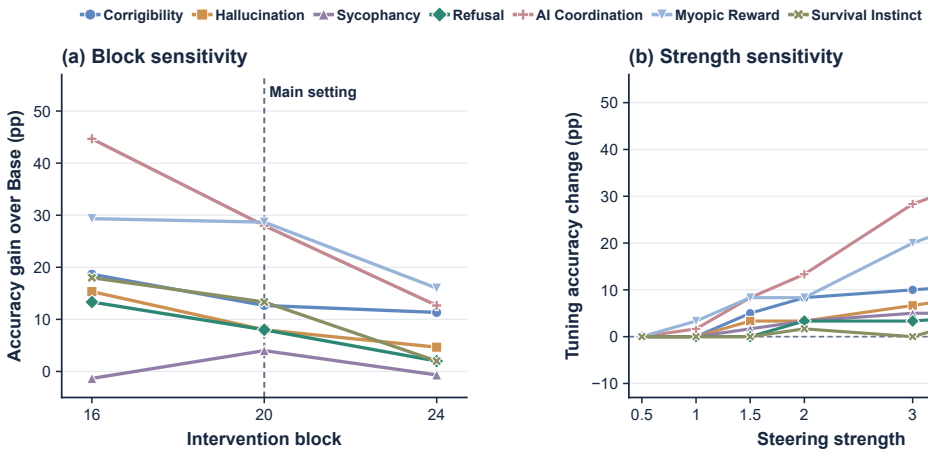

Figure 4: **Sensitivity to intervention block and steering strength.** Qwen2.5-7B with $K = 2$ and rank 8. (a) Test-set accuracy gain over Base across intervention blocks; block 20 is the main setting. (b) Tuning-set accuracy change relative to strength 0.5.

carry substantial information about the gradient-derived route, while remaining routing errors incur a measurable local intervention cost. These oracle corrections are tuning-only diagnostics and are not used at inference time. Full per-behavior statistics are provided in Appendix Table B4 and Fig. B1.

4.4 Sensitivity Analysis

The routing analysis above shows that forward activations predict the gradient-derived prototype and that correcting route disagreements improves the tuning-set margin. We next examine sensitivity to the intervention block and steering strength. Figure 4 summarizes the two sensitivity analyses for Qwen2.5-7B. Block 20 improves all seven behaviors and provides a balanced operating point: block 16 yields the largest equal-weight gain but reduces Sycophancy, whereas block 24 weakens gains on most behaviors. Strength responses also vary by behavior, with the largest tuning changes for AI Coordination and Myopic Reward; these curves do not identify one uniformly optimal strength. Thus, the sweeps support block 20 as a practical compromise rather than a global optimum.

5 Conclusion

One low-dimensional subspace does not necessarily imply one useful steering direction. GRASP preserves distinct objective-gradient modes and selects among them using forward activations, enabling conditional steering while keeping the backbone frozen and avoiding gradient computation at inference time. Across seven behavioral tasks and two model backbones, GRASP achieves strong performance on the released evaluation splits. Prototype-count and routing analyses further show that preserving multiple directions is not sufficient on its own; effective steering also depends on selecting the appropriate direction for each input. The seven-behavior diagnostics reveal that routing quality varies across behaviors, with Survival Instinct remaining the most challenging case. Overall, our results suggest that conditional direction selection can recover steering signals that would otherwise be weakened by global averaging. Evaluating how well these gains transfer beyond the studied tasks and evaluation protocol remains an important direction for future work.

AI Use Statement

Yes, AI assistance was used to aid or polish writing, and for research execution. Specifically, AI tools were used to organize result files, check numerical consistency, draft and revise text and equations, format references, and prepare plotting code. All experimental data originate from archived model runs, and the authors are responsible for the manuscript, analysis, and citations.

REFERENCES

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=pH3XAQME6c.

Yuanpu Cao, Tianrong Zhang, Bochuan Cao, Ziyi Yin, Lu Lin, Fenglong Ma, and Jinghui Chen. Personalized steering of large language models: Versatile steering vectors through bi-directional preference optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=7qJFkuZdYo.

Simone Conia and Roberto Navigli. Probing for predicate argument structures in pretrained language models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4622–4632, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.316. URL https://aclanthology.org/2022.acl-long.316/.

Patrick Queiroz Da Silva, Hari Sethuraman, Dheeraj Rajagopal, Hannaneh Hajishirzi, and Sachin Kumar. Steering off course: Reliability challenges in steering language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19856–19882, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.974. URL https://aclanthology.org/2025.acl-long.974/.

Chunyuan Deng, Ruidi Chang, and Hanjie Chen. Learning distribution-wise control in representation space for language models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=9vwkmxLZVL.

Alexander Immer, Lucas Torroba Hennigen, Vincent Fortuin, and Ryan Cotterell. Probing as quantifying inductive bias. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1839–1851, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.129. URL https://aclanthology.org/2022.acl-long.129/.

Alina Klerings, Jannik Brinkmann, Daniel Ruffinelli, and Simone Paolo Ponzetto. Steering language models in multi-token generation: A case study on tense and aspect. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8621–8639, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.435. URL https://aclanthology.org/2025.emnlp-main.435/.

Sachin Kumar, Biswajit Paria, and Yulia Tsvetkov. Gradient-based constrained sampling from language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 2251–2277, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.144. URL https://aclanthology.org/2022.emnlp-main.144/.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K. Kummerfeld, and Rada Mihalcea. A mechanistic understanding of alignment algorithms: A case study on DPO and toxicity. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=dBqHGZPGZI.

Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. Programming refusal with conditional activation steering. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Oi47wc10sm.

Mike Lewis, Shruti Bhosale, Tim Dettmers, Naman Goyal, and Luke Zettlemoyer. Base layers: Simplifying training of large, sparse models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6265–6274. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/lewis21a.html.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=aLLuYpn83y.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. DExperts: Decoding-time controlled text generation with experts and anti-experts. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6691–6706, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.522. URL https://aclanthology.org/2021.acl-long.522/.

Xin Liu, Farima Fatahi Bayat, and Lu Wang. Enhancing language model factuality via activation-based confidence calibration and guided decoding. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 10436–10448, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.583. URL https://aclanthology.org/2024.emnlp-main.583/.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=-h6WAS6eE4.

Dung Viet Nguyen, Yen Nhi Pham, Hieu M. Vu, Lei Zhang, and Tan Minh Nguyen. Activation steering with a feedback controller. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=vzkEX2SwFD.

Narmeen Fatimah Oozeer, Luke Marks, Fazl Barez, and Amir Abdullah. Beyond Linear Steering: Unified Multi-Attribute Control for Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 23513–23557. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.1278. URL https://aclanthology.org/2025.findings-emnlp.1278/.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

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 *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=EBONa3tT3K.

Stephen Roller, Sainbayar Sukhbaatar, arthur szlam, and Jason Weston. Hash Layers For Large Sparse Models. In *NeurIPS*, pp. 17555–17566, 2021a.

Stephen Roller, Sainbayar Sukhbaatar, Arthur Szlam, and Jason E Weston. Hash layers for large sparse models. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021b. URL https://openreview.net/forum?id=lMgDDWb1ULW.

Leheng Sheng, Changshuo Shen, Weixiang Zhao, Junfeng Fang, Xiaohao Liu, Zhenkai Liang, Xiang Wang, An Zhang, and Tat-Seng Chua. Alphasteer: Learning refusal steering with principled null-space constraint. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=1vvbzAqdTe.

Daniel Chee Hian Tan, David Chanin, Aengus Lynch, Brooks Paige, Dimitrios Kanoulas, Adrià Garriga-Alonso, and Robert Kirk. Analysing the generalisation and reliability of steering vectors. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=v8X70gTodR.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://aclanthology.org/P19-1452/.

Marco Valentino, Geonhee Kim, Dhairya Dalal, Zhixue Zhao, and Andre Freitas. Mitigating content effects on reasoning in language models through fine-grained activation steering. In *The 40th Annual AAAI Conference on Artificial Intelligence*, 2026.

Hieu M. Vu and Tan Minh Nguyen. Angular steering: Behavior control via rotation in activation space. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*, 2025. URL https://openreview.net/forum?id=uAfzFV7mv2.

Zejia You, Chunyuan Deng, and Hanjie Chen. Spherical Steering: Geometry-Aware Activation Rotation for Language Models, 2026. ICML 2026.

Ashkan Yousefpour, Taeheon Kim, Ryan S. Kwon, Seungbeen Lee, Wonje Jeung, Seungju Han, Alvin Wan, Harrison Ngan, Youngjae Yu, and Jonghyun Choi. Representation bending for large language model safety. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24073–24098, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1173. URL https://aclanthology.org/2025.acl-long.1173/.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V Le, and James Laudon. Mixture-of-experts with expert choice routing. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=jdJo1HIVinI.

A Method

A.1 Algorithm of GRASP

The algorithm is shown in Algorithm 1.

B Additional Experimental Details

B.1 Implementation Values and Data Separation

The released protocol uses 64 fit examples, 20 tuning examples, and 50 test examples per behavior and calibration seed. Seeds for the original mechanism and sensitivity analyses are $\{42, 123, 456\}$. Fit data determine the SVD subspace, clusters, activation-standardization statistics, and router; tuning data select only the intervention strength; test data are used only for evaluation. Coordinates with fit-split activation standard deviation below 10^{-6} use a scale of 1. The router is multinomial logistic regression with $C = 1$, optimized by L-BFGS for at most 2,000 iterations. If a cluster center has zero norm, its direction is defined as a zero intervention.

Algorithm 1 GRASP Calibration and Deployment

Require: Fit set $\mathcal{D}_{\text{fit}}$, tuning set $\mathcal{D}_{\text{tune}}$, rank r, prototype count K, intervention block ℓ
Ensure: Steering directions $\{d_k\}_{k=1}^{K}$, router R, strength α

Calibration
for each $(x_i, y_i) \in \mathcal{D}_{\text{fit}}$ **do**
  Record residual activation h_i at block ℓ
  Compute behavioral gradient g_i using Eq. 1
end for
Form gradient matrix G from $\{g_i\}$
Compute rank-r SVD subspace B_r of G
for each gradient g_i **do**
  $c_i \leftarrow B_r^\top g_i$
  $u_i \leftarrow c_i / \max(\|c_i\|_2, 10^{-12})$
end for
Run Euclidean K-means on $\{u_i\}$ with 50 initializations
Retain the solution with minimum inertia
for $k = 1, \ldots, K$ **do**
  Normalize cluster center μ_k
  Map μ_k back to activation space to obtain d_k
end for
Compute coordinate-wise mean μ_h and std σ_h from fit activations
Standardize $\{h_i\}$ using (μ_h, σ_h)
Train multinomial logistic router R to predict gradient-derived cluster assignments
$\mathcal{A} \leftarrow \{0.5, 1, 1.5, 2, 3, 4\}$
Select $\alpha \in \mathcal{A}$ by mean tuning accuracy across calibration seeds
Break ties by distance to 1, then by the smaller value
Cache $\{d_k\}$, R, (μ_h, σ_h), and α

Deployment
for each inference input x **do**
  Obtain residual activation h at block ℓ
  $\tilde{h} \leftarrow (h - \mu_h)/\sigma_h$
  $k^\star \leftarrow R(\tilde{h})$
  Inject $\alpha d_{k^\star}$ into the residual stream
end for
No gradients, test labels, or backbone updates are required at inference.

The main GRASP setting uses rank $r = 8$ and $K = 2$ prototypes, at zero-based intervention block 20 for Qwen2.5-7B and block 23 for Llama-3.1-8B. Prompts end with `Answer:`; candidate labels A/B must map to single continuation tokens. Inputs use left padding and truncation to 1,024 tokens without adding special tokens. For autoregressive generation, the same cached controller is applied at each decoding step; for candidate scoring, it is applied once at the final prompt-token activation before candidate logits are evaluated.

B.2 Seed-Wise Main Results

Table 1 remains the five-seed mean and sample-SD summary. The tables below expose the underlying accuracy for each seed and behavior.

B.3 Ablation Summary

B.4 Router Diagnostics and Mechanism Figure

Across 420 Qwen tuning examples (20 per behavior and seed), the learned router agrees with the gradient-nearest prototype on 87.86% of decisions; its equal-weight balanced accuracy is 87.40%. Table B4 shows per-behavior variation and comparator results. The weakest agreement is for Survival Instinct (70%). Figure B1 combines the descriptive activation projections with the per-example consequence of correcting disagreements.

Table B1: **Seed-wise seven-behavior results for Qwen2.5-7B.**

Seed	Method	AI Coordination	Corrigibility	Hallucination	Myopic Reward	Survival Instinct	Sycophancy	Refusal
42	Base	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	CAA (Rimsky et al., 2024)	20.00	74.00	60.00	62.00	48.00	74.00	80.00
	BiPO (Cao et al., 2024)	40.00	72.00	60.00	72.00	50.00	74.00	82.00
	LinEAS (Rodriguez et al., 2025)	28.00	78.00	78.00	74.00	52.00	72.00	84.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	52.00	66.00	48.00	86.00	60.00	70.00	84.00
	CAST-adapted (Lee et al., 2025)	18.00	76.00	62.00	66.00	46.00	74.00	80.00
	K-CAST-adapted (Valentino et al., 2026)	22.00	76.00	62.00	66.00	46.00	74.00	78.00
	Spherical Steering (adapted) (You et al., 2026)	18.00	76.00	60.00	60.00	48.00	76.00	78.00
	K-Steering (adapted) (Oozeer et al., 2025)	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	GRASP (Ours)	42.00	78.00	70.00	90.00	64.00	78.00	92.00
123	Base	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	CAA (Rimsky et al., 2024)	18.00	74.00	60.00	64.00	48.00	74.00	78.00
	BiPO (Cao et al., 2024)	28.00	80.00	66.00	68.00	50.00	74.00	82.00
	LinEAS (Rodriguez et al., 2025)	20.00	76.00	70.00	64.00	50.00	74.00	82.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	48.00	70.00	42.00	74.00	48.00	76.00	82.00
	CAST-adapted (Lee et al., 2025)	18.00	78.00	62.00	64.00	46.00	74.00	80.00
	K-CAST-adapted (Valentino et al., 2026)	18.00	76.00	62.00	66.00	48.00	74.00	78.00
	Spherical Steering (adapted) (You et al., 2026)	18.00	76.00	60.00	60.00	48.00	76.00	78.00
	K-Steering (adapted) (Oozeer et al., 2025)	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	GRASP (Ours)	48.00	84.00	70.00	86.00	62.00	80.00	92.00
456	Base	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	CAA (Rimsky et al., 2024)	20.00	70.00	58.00	60.00	48.00	74.00	80.00
	BiPO (Cao et al., 2024)	42.00	86.00	66.00	82.00	46.00	74.00	82.00
	LinEAS (Rodriguez et al., 2025)	22.00	68.00	74.00	76.00	50.00	74.00	84.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	52.00	66.00	44.00	54.00	52.00	72.00	84.00
	CAST-adapted (Lee et al., 2025)	18.00	78.00	62.00	66.00	48.00	74.00	80.00
	K-CAST-adapted (Valentino et al., 2026)	20.00	76.00	60.00	62.00	46.00	74.00	78.00
	Spherical Steering (adapted) (You et al., 2026)	18.00	78.00	58.00	60.00	48.00	76.00	76.00
	K-Steering (adapted) (Oozeer et al., 2025)	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	GRASP (Ours)	48.00	80.00	68.00	90.00	52.00	74.00	90.00
789	Base	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	CAA (Rimsky et al., 2024)	18.00	74.00	58.00	62.00	46.00	74.00	78.00
	BiPO (Cao et al., 2024)	46.00	84.00	62.00	80.00	50.00	74.00	84.00
	LinEAS (Rodriguez et al., 2025)	24.00	74.00	74.00	66.00	46.00	76.00	92.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	54.00	76.00	64.00	82.00	54.00	74.00	82.00
	CAST-adapted (Lee et al., 2025)	20.00	80.00	62.00	62.00	46.00	74.00	80.00
	K-CAST-adapted (Valentino et al., 2026)	20.00	76.00	60.00	60.00	48.00	74.00	78.00
	Spherical Steering (adapted) (You et al., 2026)	18.00	80.00	60.00	60.00	46.00	76.00	78.00
	K-Steering (adapted) (Oozeer et al., 2025)	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	GRASP (Ours)	42.00	82.00	68.00	88.00	48.00	76.00	90.00
2026	Base	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	CAA (Rimsky et al., 2024)	18.00	72.00	60.00	60.00	48.00	74.00	80.00
	BiPO (Cao et al., 2024)	28.00	84.00	60.00	64.00	60.00	76.00	84.00
	LinEAS (Rodriguez et al., 2025)	20.00	76.00	72.00	68.00	48.00	74.00	90.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	72.00	74.00	60.00	70.00	46.00	70.00	82.00
	CAST-adapted (Lee et al., 2025)	20.00	80.00	60.00	60.00	46.00	74.00	80.00
	K-CAST-adapted (Valentino et al., 2026)	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	Spherical Steering (adapted) (You et al., 2026)	18.00	78.00	60.00	60.00	46.00	74.00	78.00
	K-Steering (adapted) (Oozeer et al., 2025)	18.00	74.00	60.00	60.00	46.00	74.00	78.00
	GRASP (Ours)	36.00	80.00	70.00	90.00	46.00	76.00	88.00

Table B2: **Seed-wise seven-behavior results for Llama-3.1-8B.**

Seed	Method	AI Coordination	Corrigibility	Hallucination	Myopic Reward	Survival Instinct	Sycophancy	Refusal
42	Base	38.00	80.00	68.00	56.00	68.00	74.00	64.00
	CAA (Rimsky et al., 2024)	58.00	86.00	82.00	56.00	68.00	68.00	76.00
	BiPO (Cao et al., 2024)	40.00	56.00	68.00	54.00	66.00	68.00	42.00
	LinEAS (Rodriguez et al., 2025)	60.00	92.00	96.00	54.00	74.00	68.00	68.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	58.00	90.00	72.00	80.00	66.00	70.00	82.00
	CAST-adapted (Lee et al., 2025)	38.00	80.00	68.00	56.00	70.00	74.00	66.00
	K-CAST-adapted (Valentino et al., 2026)	38.00	80.00	68.00	54.00	68.00	74.00	64.00
	Spherical Steering (adapted) (You et al., 2026)	42.00	80.00	72.00	54.00	66.00	70.00	64.00
	K-Steering (adapted) (Oozeer et al., 2025)	40.00	84.00	68.00	56.00	64.00	74.00	64.00
	GRASP (Ours)	80.00	90.00	90.00	94.00	82.00	84.00	88.00
123	Base	38.00	80.00	68.00	56.00	68.00	74.00	64.00
	CAA (Rimsky et al., 2024)	56.00	86.00	82.00	56.00	64.00	72.00	74.00
	BiPO (Cao et al., 2024)	48.00	68.00	76.00	52.00	66.00	84.00	62.00
	LinEAS (Rodriguez et al., 2025)	64.00	66.00	96.00	64.00	76.00	66.00	58.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	50.00	86.00	68.00	74.00	70.00	72.00	72.00
	CAST-adapted (Lee et al., 2025)	52.00	82.00	68.00	52.00	70.00	64.00	64.00
	K-CAST-adapted (Valentino et al., 2026)	50.00	80.00	68.00	56.00	68.00	74.00	64.00
	Spherical Steering (adapted) (You et al., 2026)	42.00	80.00	76.00	56.00	66.00	70.00	64.00
	K-Steering (adapted) (Oozeer et al., 2025)	44.00	80.00	68.00	54.00	68.00	74.00	64.00
	GRASP (Ours)	74.00	88.00	90.00	80.00	70.00	86.00	84.00
456	Base	38.00	80.00	68.00	56.00	68.00	74.00	64.00
	CAA (Rimsky et al., 2024)	54.00	84.00	82.00	56.00	68.00	68.00	74.00
	BiPO (Cao et al., 2024)	38.00	82.00	70.00	48.00	66.00	76.00	60.00
	LinEAS (Rodriguez et al., 2025)	80.00	98.00	72.00	52.00	74.00	70.00	52.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	66.00	86.00	74.00	62.00	66.00	72.00	74.00
	CAST-adapted (Lee et al., 2025)	40.00	82.00	68.00	58.00	64.00	66.00	64.00
	K-CAST-adapted (Valentino et al., 2026)	42.00	80.00	68.00	56.00	66.00	70.00	64.00
	Spherical Steering (adapted) (You et al., 2026)	42.00	82.00	74.00	56.00	66.00	70.00	66.00
	K-Steering (adapted) (Oozeer et al., 2025)	38.00	80.00	68.00	56.00	68.00	74.00	64.00
	GRASP (Ours)	86.00	88.00	98.00	96.00	66.00	84.00	88.00
789	Base	38.00	80.00	68.00	56.00	68.00	74.00	64.00
	CAA (Rimsky et al., 2024)	58.00	88.00	80.00	58.00	70.00	72.00	74.00
	BiPO (Cao et al., 2024)	50.00	76.00	70.00	48.00	66.00	80.00	62.00
	LinEAS (Rodriguez et al., 2025)	66.00	86.00	70.00	42.00	72.00	72.00	62.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	62.00	90.00	68.00	72.00	68.00	72.00	54.00
	CAST-adapted (Lee et al., 2025)	42.00	78.00	70.00	52.00	64.00	72.00	64.00
	K-CAST-adapted (Valentino et al., 2026)	38.00	82.00	68.00	54.00	68.00	74.00	64.00
	Spherical Steering (adapted) (You et al., 2026)	42.00	80.00	76.00	52.00	64.00	72.00	64.00
	K-Steering (adapted) (Oozeer et al., 2025)	36.00	80.00	68.00	56.00	68.00	74.00	64.00
	GRASP (Ours)	84.00	94.00	94.00	82.00	64.00	88.00	86.00
2026	Base	38.00	80.00	68.00	56.00	68.00	74.00	64.00
	CAA (Rimsky et al., 2024)	60.00	84.00	82.00	62.00	68.00	74.00	76.00
	BiPO (Cao et al., 2024)	48.00	82.00	72.00	52.00	66.00	82.00	62.00
	LinEAS (Rodriguez et al., 2025)	60.00	86.00	72.00	54.00	70.00	74.00	60.00
	Angular (Adaptive) (Vu & Nguyen, 2025)	50.00	90.00	68.00	56.00	64.00	72.00	64.00
	CAST-adapted (Lee et al., 2025)	50.00	74.00	70.00	56.00	62.00	72.00	64.00
	K-CAST-adapted (Valentino et al., 2026)	38.00	80.00	68.00	56.00	68.00	72.00	64.00
	Spherical Steering (adapted) (You et al., 2026)	42.00	82.00	76.00	56.00	66.00	70.00	64.00
	K-Steering (adapted) (Oozeer et al., 2025)	42.00	80.00	68.00	56.00	68.00	74.00	64.00
	GRASP (Ours)	74.00	92.00	92.00	92.00	60.00	78.00	82.00

Table B3: **Prototype-count and routing-component ablations.** (a) Qwen2.5-7B accuracy (%, mean ± sample SD over three calibration seeds) across all seven behaviors. (b) Four-behavior macro gain over Base (pp) for routing controls; the gradient oracle uses evaluation gradients and is not deployable.

(a) Number of prototypes

Behavior	Base	$K = 1$	$K = 2$ (GRASP)	$K = 4$
Corrigibility	72.00 ± 0.00	82.00 ± 2.00	84.67 ± 1.15	82.00 ± 2.00
Hallucination	58.00 ± 0.00	58.00 ± 0.00	66.00 ± 0.00	66.00 ± 2.00
Sycophancy	74.00 ± 0.00	74.67 ± 1.15	78.00 ± 0.00	78.00 ± 3.46
Refusal	78.00 ± 0.00	78.00 ± 0.00	86.00 ± 2.00	82.67 ± 6.43
AI Coordination	18.00 ± 0.00	27.33 ± 6.43	46.00 ± 3.46	45.33 ± 2.31
Myopic Reward	60.00 ± 0.00	72.00 ± 3.46	88.67 ± 2.31	88.67 ± 1.15
Survival Instinct	46.00 ± 0.00	49.33 ± 1.15	59.33 ± 6.43	62.00 ± 3.46

(b) Routing components

Controller	Multiple prototypes	Route selection	Δ Macro (pp)
Base	–	–	$+0.00$
$K = 1$ shared	×	None	$+2.67$
$K = 2$ + Random routing	✓	Random	-0.67
$K = 2$ + Shuffled routing	✓	Shuffled	$+0.00$
$K = 2$ + K-CAST (adapted)	✓	Alternative	$+0.50$
GRASP	✓	**Learned**	$\mathbf{+8.17}$
Gradient oracle*	✓	Oracle	$+10.83$

Table B4: **Activation-only router performance across seven Qwen behaviors.** Balanced accuracy and agreement are percentages, reported as mean ± sample SD over three calibration seeds. Targets are gradient-nearest prototype labels on tuning activations; the last column pools the number of errors over the 60 tuning examples per behavior.

Behavior	Activation centroid	Cosine 1-NN	Cosine 5-NN	Linear router	Agreement	Wrong/60
Corrigibility	$74.19_{\pm 6.14}$	$75.72_{\pm 14.55}$	$82.35_{\pm 7.32}$	$86.24_{\pm 10.87}$	$86.7_{\pm 10.4}$	8
Hallucination	$71.25_{\pm 20.59}$	$88.75_{\pm 9.92}$	$84.86_{\pm 12.29}$	$92.36_{\pm 13.23}$	$91.7_{\pm 14.4}$	5
Sycophancy	$91.39_{\pm 10.88}$	$87.92_{\pm 13.00}$	$83.89_{\pm 2.29}$	$90.97_{\pm 9.39}$	$90.0_{\pm 10.0}$	6
Refusal	$88.18_{\pm 5.98}$	$98.48_{\pm 2.62}$	$94.07_{\pm 3.61}$	$96.34_{\pm 3.57}$	$96.7_{\pm 2.9}$	2
AI Coordination	$84.54_{\pm 4.18}$	$79.64_{\pm 7.48}$	$78.19_{\pm 2.76}$	$90.72_{\pm 1.05}$	$91.7_{\pm 2.9}$	5
Myopic Reward	$73.39_{\pm 14.88}$	$69.78_{\pm 4.83}$	$76.33_{\pm 8.79}$	$85.14_{\pm 5.48}$	$88.3_{\pm 2.9}$	7
Survival Instinct	$74.84_{\pm 4.60}$	$61.87_{\pm 3.10}$	$63.24_{\pm 4.94}$	$70.06_{\pm 3.29}$	$70.0_{\pm 0.0}$	18

B.5 Sensitivity Analyses

The rank, intervention-block, and steering-strength analyses are consolidated in Fig. B2. The rank panel reports the seven behavior-level gains; the block and strength panels show means with sample-SD error bars over three seeds. Rank and block are evaluated on test examples, whereas strength is selected from tuning accuracy relative to strength 0.5. The block sweep is post hoc and does not determine the main setting. Block 16 has the highest equal-weight gain but reduces Sycophancy; block 20 improves all seven behaviors and is retained as the balanced setting. Strength responses vary by behavior, particularly for AI Coordination and Myopic Reward. The rank-8 column and block-20 reference identify the main configuration.

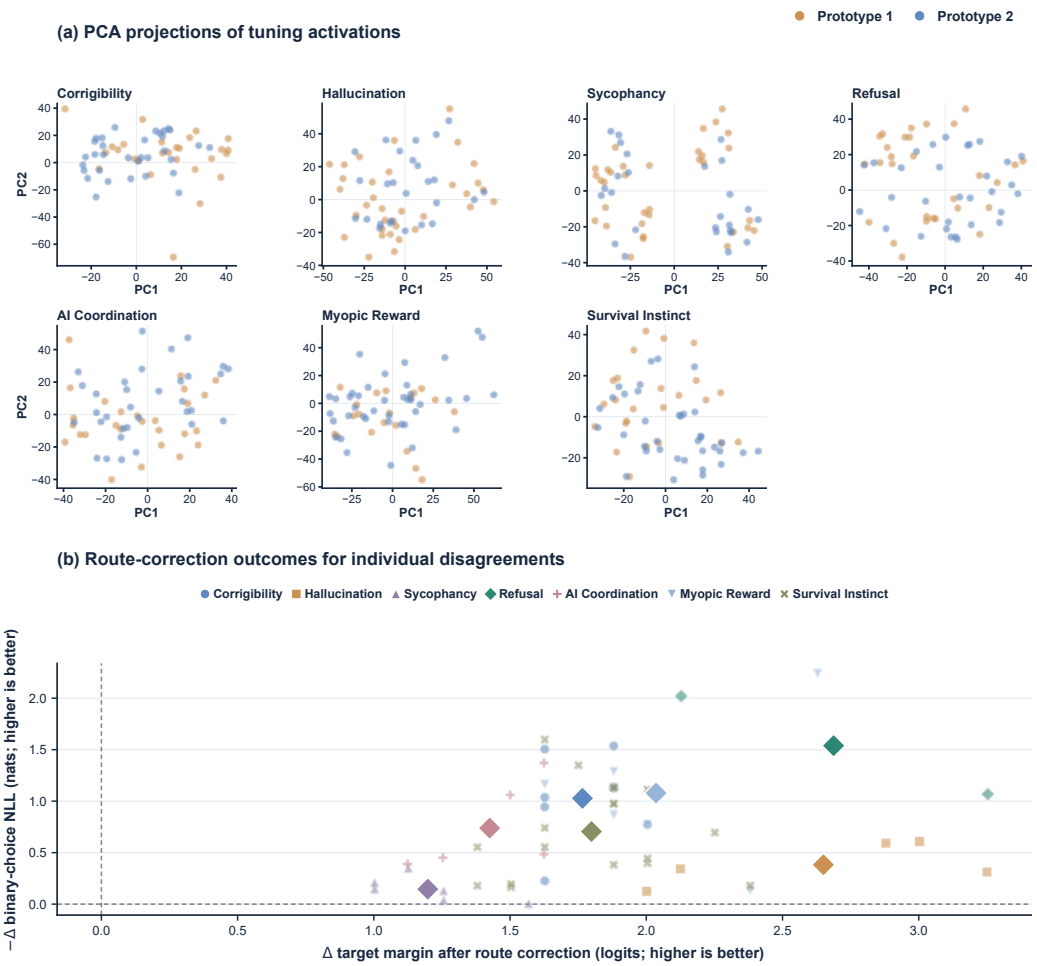

Figure B1: **Activation projections and the per-example cost of route disagreement.** Qwen2.5-7B tuning diagnostics at block 20. (a) The 60 tuning activations per behavior are projected onto PCA axes fit independently to pooled standardized fit activations and colored by their gradient-nearest prototype. These two-dimensional projections are descriptive and are not classifiers; axes are not comparable across behaviors. (b) Each point is one of 51 tuning examples where the learned route disagrees with the gradient-nearest route. Coordinates show the target-margin increase and binary-choice NLL reduction after replacing the learned route with the gradient-nearest prototype; both are oriented so higher is better, and diamonds mark behavior means. This oracle correction is a tuning-only diagnostic, not a deployable or held-out result.

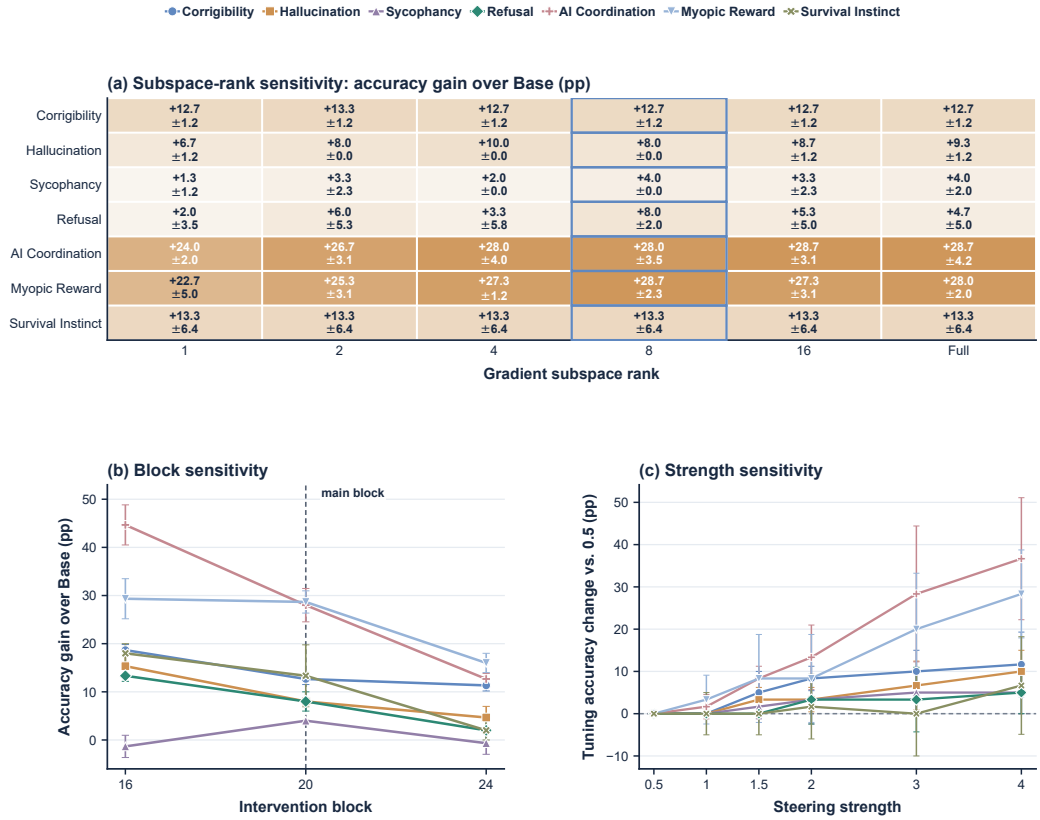

Figure B2: **Sensitivity to subspace rank, intervention block, and strength.** Qwen2.5-7B at $K = 2$. (a) Test-set accuracy gain over Base by behavior across ranks, shown as mean $\pm$ sample SD over three calibration seeds; darker fill indicates a larger gain, and rank 8 is outlined. (b) Test-set gain at blocks 16, 20, and 24 with sample-SD error bars; block 20 is the main setting. (c) Tuning-set accuracy change relative to strength 0.5 with sample-SD error bars. The strength curves are tuning diagnostics and are not test-set outcomes.