COVERAGE BEFORE CONTROL: ROUTE-INSTRUCTION GROUNDING AND STEERING FOR CONTROLLABLE RETROSYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-step retrosynthesis models are commonly evaluated by their ability to recover recorded reactions. In practice, chemists may need to choose among several precursor sets for the same product, for example to preserve a particular motif. Recovering a recorded answer alone does not establish this ability to follow a preference. Satisfying such requests requires both coverage of relevant alternatives and control over which alternatives are favored. We introduce Route-Instruction Grounding and Steering (RIGS), a two-stage framework for instruction-conditioned retrosynthesis. Stage A trains a language projector, teaching it which alternatives an instruction favors or discourages. Stage B uses the projector learned in Stage A to steer a frozen generative model through lightweight residual adapters. We construct nested one-to-many training supports by pairing each product with increasing numbers of candidate precursor sets. Extensive experiments demonstrate that broader support helps the model generate a wider range of alternatives, and RIGS can learn to guide generation according to instructions. The relationship between coverage and control is consistent across model scales but non-monotone. Code and materials are available at `https://anonymous.4open.science/r/RIGS-9695/`.

1 INTRODUCTION

Retrosynthesis involves deciding which disconnection to pursue (Thakkar et al., 2023). A target molecule can admit several plausible precursor sets (Chen et al., 2019), and a chemist may prefer an alternative that preserves a particular motif (Westerlund et al., 2025). Such preferences call for a model that can generate alternatives for the same product and make its output responsive to a natural-language request. We study this choice at the single-step level; throughout this paper, a route denotes one precursor-set alternative.

Top-k exact-match accuracy is widely used to evaluate single-step retrosynthesis models (Igashov et al., 2024; Wang et al., 2025; 2026). It measures recovery of a recorded precursor set, but can miss other plausible alternatives (Chen et al., 2019; Zagribelnyy et al., 2026). It also does not directly test whether a model can select among alternatives according to a user's request. Preference-guided retrosynthesis helps align route choices with practical goals such as material cost and yield (Liu et al., 2024a). Natural language lets chemists express requirements for each target, such as which transformations to favor or avoid, as explored by Synthegy and Synthelite (Bran et al., 2026; Nguyen et al., 2025). Disconnection prompts and inference-time reward guidance also provide ways to influence generated routes (Thakkar et al., 2023; Yadav et al., 2025). However, learning to ground free-form instructions in preferences among precursor alternatives for the same product and transfer this grounding into direct single-step generation remains less studied. We therefore ask how a generator can learn such preferences without depending on a fixed candidate list at inference.

This question depends on candidate coverage. In a motivating analysis, we construct the USPTO multi-route collection (USPTO-MR) from USPTO-Full (Lowe, 2017) by retaining products with at least two distinct recorded precursor sets (Appendix H.2). Even this multi-route subset contains only 2.37 observed precursor sets per product on average (Table 24). An instruction can promote an alternative only if the generator can produce it. Synthetic expansion supplies additional candidates

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.

for base-model training and preference supervision, but broader support alone does not establish responsiveness to language. We therefore distinguish *coverage*, the alternatives recovered under finite sampling, from *control*, the instruction-dependent choice among those alternatives near the top of the returned list. This motivates our organizing principle, *coverage before control*.

We introduce Route-Instruction Grounding and Steering (RIGS) to turn candidate preferences into instruction-conditioned generation. RIGS contains two stages: Stage A trains a language projector using offline comparisons among candidates for the same product, teaching it which alternatives an instruction favors or discourages; Stage B transfers the projector to a lightweight instruction pathway connected through residual adapters to the frozen GraphDiT module of RetroDiT (Wang et al., 2026). At inference, RIGS samples precursor sets directly.

We conduct a systematic experimental evaluation of RIGS to examine how candidate coverage and instruction control interact. Broader training support gives the generator more alternatives to choose from, while instruction guidance can move preferred alternatives toward the top of the returned list. The results reveal a consistent but non-monotone interaction between coverage and control. For the 65M Top30 model, correct instructions improve preferred-set Top-1 recovery by 18.52 percentage points over shuffled instructions. Stage A grounding further helps suppress alternatives that conflict with the instruction, supporting the value of learning preferences before steering generation.

Our contributions are threefold:

- We introduce RIGS, which grounds language in product-local candidate preferences and transfers the learned representation to residual adapters on a frozen discrete-flow generator, enabling direct instruction-conditioned generation without test-time retrieval or reranking.
- We formulate and empirically examine the coverage-before-control principle. A shared-inventory study across four nested synthetic supports, an observed-reaction reference, and two model capacities separates candidate availability from instruction-dependent ranking.
- We provide controlled evaluations that test whether generation follows the instruction's content, isolate the contribution of Stage-A grounding, and assess instruction following with case-specific structural instructions and recorded reaction alternatives from Pistachio.

2 Related Work

Single-step retrosynthesis and candidate coverage. RetroBridge models a product–reactant Markov bridge (Igashov et al., 2024), while RetroDiff decomposes generation into stages (Wang et al., 2025) and DiffAlign addresses permutation alignment (Laabid et al., 2025). Existing methods broaden candidate coverage through latent variables (Chen et al., 2019), reaction-class prompts (Toniato et al., 2023), and training-data augmentation. RetroWISE uses synthetic reactions (Zhang et al., 2024), while the Triple Transformer Loop applies templates with predictive filtering (Grandjean et al., 2026). Tran et al. (2026) distinguish failures to propose candidates from failures to rank them highly. We examine how nested training supports affect instruction-dependent generation in RetroDiT's GraphDiT backbone (Wang et al., 2026).

Grounding language in precursor preferences. MolT5 studies molecule–language translation (Edwards et al., 2022), MoleculeSTM learns joint structure–text representations (Liu et al., 2023), and Mol-Instructions develops biomolecular instruction tuning (Fang et al., 2024). In retrosynthesis, T-Rex uses generated descriptions to rank reaction centers and precursors (Liu et al., 2024b), while RetroInText combines descriptions with molecular and route context (Kang et al., 2025). RIGS grounds instructions in preferences among precursor sets for the same product, learning which alternatives to favor or avoid.

Transferring preferences into generation. Multistep planners augment search (Segler et al., 2018; Chen et al., 2020) with Pareto objectives (Hastedt et al., 2026), while Ariadne conditions route generation on depth and starting materials (Morgunov & Batista, 2026). More generally, discrete diffusion models support classifier-based and classifier-free guidance (Schiff et al., 2025). Beyond Search steers autoregressive retrosynthesis with inference-time classifier guidance (Laabid & Garg, 2026), while RetroSynFlow uses Feynman–Kac reward steering (Yadav et al., 2025). RIGS trans-

fers learned precursor preferences to a frozen GraphDiT through residual adapters, enabling direct instruction-conditioned generation.

3 RIGS: Route-Instruction Grounding and Steering

RIGS learns which candidates an instruction favors, then transfers that representation into a graph generator. We train a separate no-instruction Base model on the corresponding candidate support for each candidate budget and model capacity. Its backbone remains frozen during the two guidance stages. Stage A (Sec. 3.2) learns a language projector from offline preferences; Stage B (Sec. 3.3) connects this projector to generation through residual adapters (Figure 2).

3.1 Candidate support and preference supervision

Retrosynthesis templates serve as reusable chemical reaction rules: they encode local substructure patterns and associated bond changes. Matching and applying these rules to a product x with RDChiral (Coley et al., 2019) yields complete precursor molecules by carrying over molecular context outside the matched substructures (Appendix H.4). The resulting precursor sets form the candidate pool $\mathcal{E}(x)$. We rank these candidates using a heuristic quality score and precursor diversity. The rank $\mathrm{rank}_x(r)$ is candidate r's first position in this greedy ordering (Appendix H.4, Eq. 11), with smaller ranks selected first. For an integer budget $K \geq 1$, we take the first K ranks and apply shared graph-construction checks and deduplication, denoted by Filter, to obtain the Base training support:

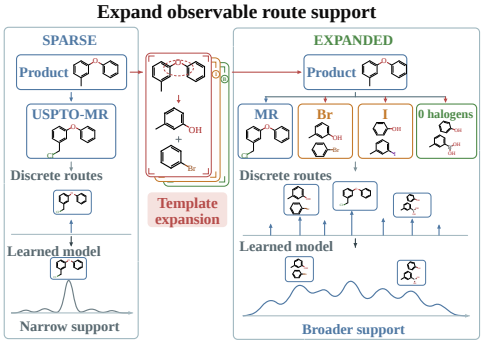


Figure 1: Support expansion. Template-derived candidates broaden training support before instruction grounding.

$$\begin{aligned} &\mathcal{A}_K(x) = \mathrm{Filter}(\{r \in \mathcal{E}(x) : \mathrm{rank}_x(r) \leq K\}), \\ &\mathcal{A}_{K_1}(x) \subseteq \mathcal{A}_{K_2}(x) \subseteq \cdots \subseteq \mathcal{A}_{K_m}(x), \qquad K_1 < \cdots < K_m. \end{aligned} \tag{1}$$

Within each support $\mathcal{A}_K(x)$, we form preferred and avoided candidate sets according to whether they follow a given guidance instruction. These sets provide preference supervision for Stage A grounding and Stage B instruction steering. Details of support construction and guidance labeling are given in Appendices H.4 and H.5.

3.2 Stage A: generator-free grounding

For a target product x, let $r_1, \ldots, r_n$ denote alternative precursor sets and c a natural-language instruction. Stage A learns to align instruction representations with those of preferred candidates. It uses frozen Qwen3-Embedding-8B (Zhang et al., 2025) as the language encoder E_T and a frozen reaction-pair encoder E_R for each product–precursor pair (x, r_i) (Appendix I.1). The language encoder converts each instruction into a sequence of token embeddings. We compute these text features offline and cache them for reuse in both stages. Stage A trains only a projector P_ϕ to map the text features into the reaction representation space. We denote the instruction token embeddings by $H_c = E_T(c)$, the candidate reaction embedding by $v_i = E_R(x, r_i)$, and the projected instruction embeddings by $Z_c = P_\phi(H_c)$.

We use Z_c and $\{v_i\}$ to predict a preference distribution $\widehat{q}^c_\phi$ over candidates. We train the projector to match this prediction to the target preference distribution q^c constructed from offline supervision. The target distribution q^c assigns higher weights to candidates that better satisfy the instruction. For instruction c, $\mathcal{P}_c$ contains the indices of positive (preferred) candidates, and $\mathcal{N}_c$ contains those of negative (avoided) candidates (Appendix H.5). The objective of Stage A is:

$$\mathcal{L}_A = \lambda_{\text{list}}\ \mathrm{KL}(q^c \| \widehat{q}^c_\phi) \quad + \lambda_{\text{pair}}\, \mathcal{L}_{\text{pair}}(\mathcal{P}_c, \mathcal{N}_c). \tag{2}$$

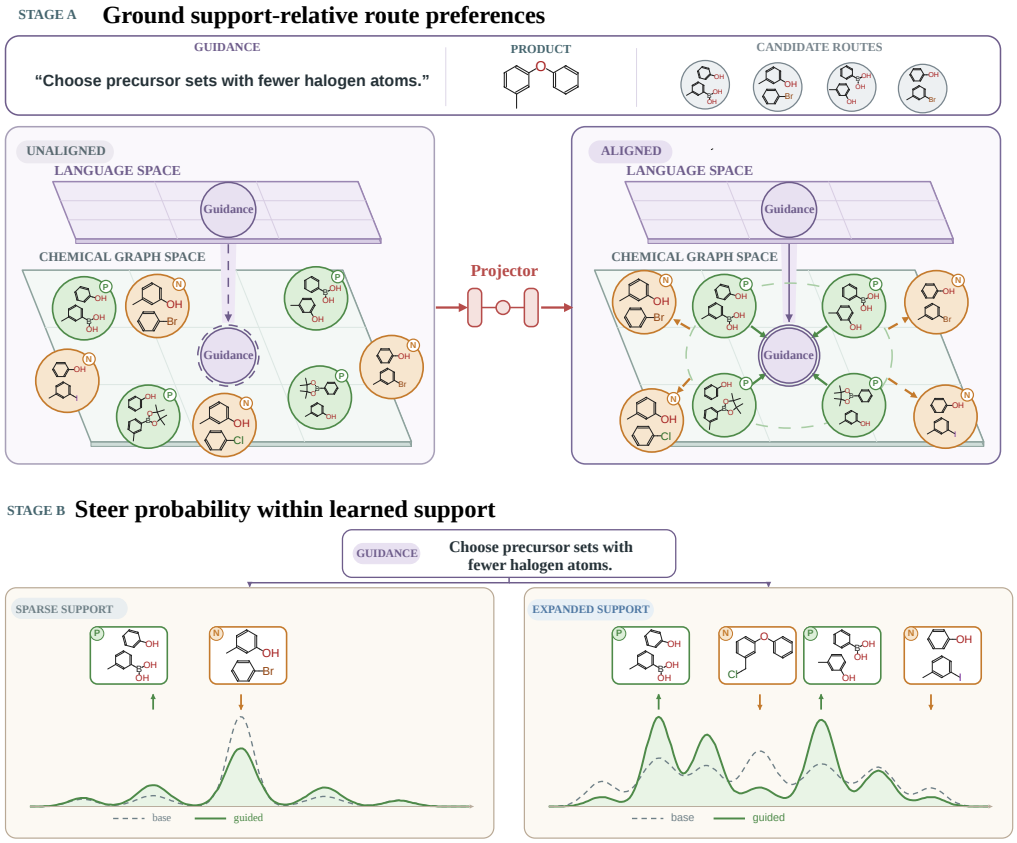


Figure 2: **RIGS-GraphDiT: grounding and steering.** Stage A grounds natural-language guidance against preferred and avoided routes. Stage B transfers this grounding into a residual instruction pathway that steers probability within the frozen generator's learned support.

The KL term matches the predicted distribution to the target preference weights, while the pairwise term encourages preferred candidates to receive higher scores than avoided candidates. Appendix I.3 specifies the candidate scoring rule and the pairwise loss. We minimize $\mathcal{L}_A$ by updating only P_ϕ. We select $P_{\phi^\star}$ on the validation set and use its parameters to initialize Stage B with a fresh optimizer. Algorithm 1 summarizes this process.

3.3 Stage B: Residual Instruction Steering

Stage B connects the instruction representation learned in Stage A to the frozen graph generator and reuses the cached text features $H_c = E_T(c)$. We initialize the projectors from $P_{\phi^\star}$. These projectors transform the cached text features into instruction representations for node-to-text cross-attention adapters at selected GraphDiT layers. Let $M_c^{(\ell)}$ denote the projected text features supplied to layer ℓ. We denote the collection of these projected features by M_c.

At each selected layer ℓ, the adapter $A_{\ell,\theta}$ uses cross-attention to convert the projector's instruction features $M_c^{(\ell)}$ into a residual update of the current graph-node states. For node states h_ℓ, frozen GraphDiT block B_ℓ, and steering strength $\alpha \geq 0$, the update is

$$h_{\ell+1} = B_\ell\Big(h_\ell + \alpha A_{\ell,\theta}(h_\ell, M_c^{(\ell)})\Big). \tag{3}$$

Setting $\alpha = 0$ recovers the Base model. The projector keeps a separate embedding for each instruction token, so each graph node can use different parts of the instruction during generation. The adapters update only node representations. The frozen backbone then uses these updates to adjust edge representations. Appendix I.4 specifies projector sharing and the full masked attention update.

Training objective. We optimize only the projector and adapter parameters θ, training the guided model to denoise preferred precursor sets while regularizing its departure from the frozen Base model, which serves as the reference. With steering strength fixed at $\alpha_{\text{train}} = 1$, the core objective is:

$$\mathcal{L}_{B,\text{core}} = \lambda_{\text{pos}}\mathcal{L}_{\text{pos}} + \lambda_{\text{rank}}\mathcal{L}_{\text{rank}} + \lambda_{\text{roll}}\mathcal{L}_{\text{roll}} + \lambda_{\text{guard}}\mathcal{L}_{\text{guard}}. \tag{4}$$

The positive flow-matching loss $\mathcal{L}_{\text{pos}}$ directly trains denoising on preferred candidates. Let G_i be the clean precursor graph of candidate r_i and $G_{i,t}$ its noisy precursor graph at time t. The reference f_0 and guided model $f_{\theta,1}$ use the same $G_{i,t}$, with denoising losses

$$\begin{aligned} D_i^{\text{ref}} &= \ell_{\text{den}}(f_0(G_{i,t}, x, t), G_i)\,, \\ D_i^{\text{student}} &= \ell_{\text{den}}(f_{\theta,1}(G_{i,t}, x, t, M_c), G_i)\,. \end{aligned} \tag{5}$$

Here, ℓ_{den} sums the mean node and weighted mean edge cross-entropies over valid positions. The gain $d_i = D_i^{\text{ref}} - D_i^{\text{student}}$ is positive when guidance reduces the denoising loss. The ranking loss $\mathcal{L}_{\text{rank}}$ penalizes violations of the margin $d_i - d_j \geq m_{\text{rank}}$ for $i \in \mathcal{P}_c$ and $j \in \mathcal{N}_c$, encouraging preferred candidates to benefit more from guidance than avoided candidates. Ranking alone can satisfy this margin by worsening denoising on avoided candidates; $\mathcal{L}_{\text{pos}}$ additionally provides a direct training signal to reconstruct preferred candidates.

Two KL penalties regularize deviations from the reference predictions. Let p_{ref} and p_{student} denote the models' categorical predictions at the same node or edge position of a noisy precursor graph. For each product–instruction group g, K_g is the weighted mean node-and-edge KL across its noisy precursor graphs. The penalties take the form

$$\mathcal{L}_{\text{roll}} = \langle \text{KL}(p_{\text{ref}} \| p_{\text{student}}) \rangle_{\text{roll}}\,, \mathcal{L}_{\text{guard}} = \langle [K_g - \delta_{\text{KL}}]_+ \rangle_g\,. \tag{6}$$

$\mathcal{L}_{\text{roll}}$ penalizes prediction drift on noisy precursor graphs encountered during generation, while $\mathcal{L}_{\text{guard}}$ penalizes only the amount by which a group's mean KL exceeds δ_{KL}. For instructions about numerical properties, we add $\lambda_{\text{teach}}\mathcal{L}_{\text{teach}}$ to the core objective. This loss encourages larger denoising gains d_i for candidates whose property values better satisfy the instruction. Appendix I.5 gives the detailed loss definitions, and Appendix D.2 reports their ablation. Algorithms 1 and 2 summarize the two training stages.

Algorithm 1: Stage A: generator-free grounding

Require: $\mathcal{D}_{\text{tr}}, \mathcal{D}_{\text{val}}$; frozen E_T, E_R

Cache $H_c \leftarrow E_T(c)$ and $v_i \leftarrow E_R(x, r_i)$ over $\mathcal{D}_{\text{tr}}$
for each training iteration **do**
  Sample groups with cached encoder outputs
  $Z_c \leftarrow P_\phi(H_c)$
  Form the predicted preference distribution $\widehat{q}_\phi^c$ from $Z_c, \{v_i\}$ via Eq. (12)
  Compute $\mathcal{L}_A$ from available labels via Eq. (2)
  Update only ϕ using $\nabla_\phi \mathcal{L}_A$
end for
return $P_{\phi^\star}$ selected on $\mathcal{D}_{\text{val}}$

Algorithm 2: Stage B: residual instruction steering

Require: $\mathcal{D}_{\text{tr}}, \mathcal{D}_{\text{val}}$; frozen f_0, E_T; $P_{\phi^\star}$

Initialize projectors from $P_{\phi^\star}$; zero the adapter output projections
for each training iteration **do**
  Sample candidate groups and shared noisy precursor graphs
  Compute denoising gains d_i from Eq. (5)
  Compute $\mathcal{L}_{\text{pos}}, \mathcal{L}_{\text{rank}}, \mathcal{L}_{\text{guard}}$
  Periodically sample guided states and replay them through both models to compute $\mathcal{L}_{\text{roll}}$
  Form $\mathcal{L}_B$ via Eq. (4); add $\lambda_{\text{teach}}\mathcal{L}_{\text{teach}}$ for metric instructions
  Update only θ using $\nabla_\theta \mathcal{L}_B$
end for
return $f_{\theta^\star, \alpha_{\text{eval}}}$ selected on $\mathcal{D}_{\text{val}}$

Inference. We reuse the Base continuous-time Markov chain (CTMC) sampler (Wang et al., 2026). For product x and instruction c, we cache the projected instruction memory M_c and use $f_{\theta^\star, \alpha_{\text{eval}}}$ to supply instruction-conditioned endpoint predictions at each sampling step. To obtain a ranked list of distinct precursor candidates, we canonicalize the valid outputs from B raw draws, merge duplicates, and rank candidates by how often they were sampled. The CTMC rates and update

are given in Eqs. (19)–(20); Appendix H.9 specifies the sampling protocol. For each model, we select the checkpoint $\theta^\star$ and guidance strength α_{eval} on the validation set, using instruction-following performance and output validity. We keep both fixed during test evaluation. The evaluated scales and selection rules are detailed in Appendix A and Supplementary Section S3.

4 EXPERIMENTS

We evaluate how training support affects candidate coverage and instruction alignment in RIGS. We then examine whether instruction grounding and generation-time conditioning improve control, and assess RIGS's applicability across datasets and guidance suites.

4.1 EXPERIMENTAL SETUP

Datasets. We compare recorded precursor alternatives in USPTO-MR with synthetic-only template expansions, USPTO-MR-Top$K$ (Top$K$), for $K \in \{15, 30, 45, 60\}$. USPTO-MR groups reactions from USPTO-Full (Lowe, 2017) by canonical product and retains products with multiple distinct recorded precursor sets. All expansions use a fixed retrosynthesis template library released with DESP (Yu et al., 2024), shared across disjoint product splits. Construction details and support statistics appear in Appendices H.2–H.4.

Natural-language guidance. We construct *RDKit5*, a natural-language guidance dataset that pairs instructions with preferred and avoided precursor sets. Instructions express preferences such as *Choose precursor sets with fewer halogen atoms*. We assign these labels using property values computed with RDKit (RDKit contributors, n.d.) within each product's candidate pool (Appendix H.5).

Evaluation setup. We adopt the 8M (Large) and 65M (X-Large) backbone sizes from RetroDiT (Wang et al., 2026). For each support, we evaluate both sizes on one shared Top100 inventory of 1,201 test products. Reference is the support-matched unguided Base; Matched and Shuffled use the same guided model with the case's instruction or a different instruction, respectively. Each arm uses $B = 100$ raw draws per group without backfilling. We select checkpoints and guidance scales α_{eval} using validation data and keep them fixed for testing (Appendix A).

Evaluation metrics. Using the first B samples from each product–instruction group, we report two hit rates. $\text{Hit}_{\text{all}}@B$ measures how often these samples contain at least one precursor set from the product's Top100 inventory. $\text{Hit}_{+}@B$ measures how often they contain at least one candidate from the instruction's preferred subset. For each test product, we combine outputs from all instruction groups and count how many distinct precursor sets from its Top100 inventory were generated. Pooled recovery sums these counts across test products. For each group, we canonicalize the 100 sampled outputs and rank the distinct valid precursor sets by decreasing sampling frequency. P@k ($\uparrow$) and N@k ($\downarrow$) measure how often the top k candidates in this ranking contain at least one preferred or avoided candidate, respectively. Raw validity is the percentage of all sampled outputs that pass RDKit's molecular structure checks, counting repeated outputs each time. For unique-valid count, we remove duplicate valid outputs within each group, then average the remaining counts across groups. Further metric definitions and sampling settings appear in Appendices A and H.9, respectively.

4.2 BROADER TRAINING SUPPORT IMPROVES CANDIDATE COVERAGE

With the same sampling budget, unguided Reference models trained on Top15 recover more distinct precursor alternatives from the shared Top100 test inventory than those trained on USPTO-MR. Pooled recovery increases from 2,348 to 12,880 for the 8M model and from 1,723 to 13,661 for the 65M model (Table 1). Expanding support from Top15 to Top30 and Top45 further increases pooled recovery, with smaller gains at each step. From Top45 to Top60, pooled recovery is nearly unchanged for the 8M model and decreases for the 65M model, although both models generate more distinct valid outputs (Figure 3 (b,f); Table 1). Thus, generating more different structures does not always mean recovering more candidates in the test inventory. We next examine whether RIGS can use instructions to make preferred alternatives appear earlier in the returned list.

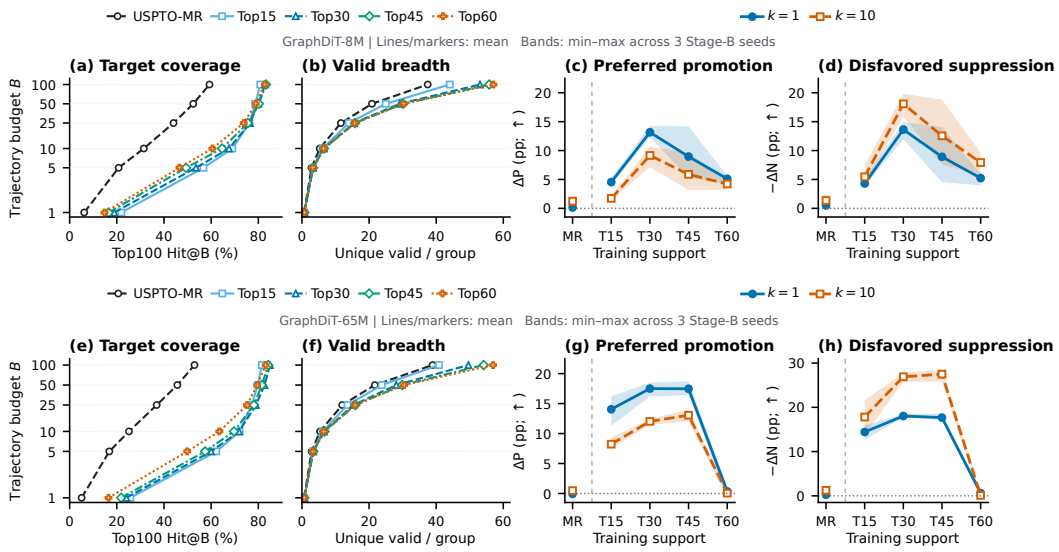


Figure 3: **Candidate coverage and instruction control across training supports.** GraphDiT-8M (top) and 65M (bottom) on the same Top100 test inventory, using equal-axis, case-balanced means. Left panels show Reference: (a,e) the rate of sampling at least one candidate from this inventory and (b,f) the number of distinct valid outputs per group, with raw-draw budget B on the vertical axis. Right panels compare Matched with Shuffled instructions at $B = 100$: (c,g) increases in preferred-set hit rate, $\Delta\mathrm{P}@k$, and (d,h) reductions in avoided-set hit rate, $-\Delta\mathrm{N}@k$, for $k = 1, 10$. Positive values indicate better instruction alignment. Control markers show means over three Stage-B training seeds; shaded bands span their minimum and maximum. Each seed uses its validation-selected checkpoint and guidance scale; Reference is fixed for each support. Result summaries and seed analyses can be found in Appendices B and C.4, respectively.

Table 1: **Candidate coverage and instruction following at validation-selected settings.** Results use $B = 100$ per group and Stage-B seed 42. Pooled recovery counts distinct product–candidate pairs recovered by Reference from the Top100 inventory across all instruction groups. Effects are in percentage points; positive values indicate improvement. Bold marks maxima in the effect columns within each model size. Absolute rates and Matched–Shuffled intervals appear in Tables 8 and 7, respectively. Matched–Reference intervals for P@1, P@10, and N@10 appear in Table 9. Three-seed summaries can be found in Appendix C.4.

Support	Reference Pooled recovery	Absolute P@1 (%)		Matched–Shuffled (pp)				Matched–Reference (pp)			
		Reference	Matched	ΔP@1	ΔP@10	−ΔN@1	−ΔN@10	ΔP@1	ΔP@10	−ΔN@1	−ΔN@10
GraphDiT-8M											
USPTO-MR	2,348	8.48	8.98	+0.08	+2.03	+0.74	+1.37	+0.51	+1.91	+0.86	+2.54
Top15	12,880	30.16	34.10	+4.10	+1.37	+3.83	+4.65	+3.95	+0.27	+4.06	+5.16
Top30	16,922	30.20	40.20	+12.89	**+10.70**	**+14.06**	+18.52	+10.00	-0.66	**+18.24**	**+27.73**
Top45	17,370	29.84	40.63	**+14.18**	+9.53	**+14.06**	**+18.79**	**+10.78**	**+2.62**	+15.20	+24.92
Top60	17,378	29.22	33.05	+5.63	+4.88	+5.94	+8.36	+3.83	+1.17	+5.55	+10.08
GraphDiT-65M											
USPTO-MR	1,723	7.70	8.16	+0.00	+0.23	-0.20	+1.60	+0.47	+2.30	-0.08	+0.70
Top15	13,661	30.23	41.25	+11.29	+7.50	+12.66	+13.83	+11.02	+2.42	+15.00	+19.57
Top30	20,075	32.93	48.83	**+18.52**	**+12.19**	**+17.89**	**+27.19**	**+15.90**	**+4.57**	+16.56	**+34.57**
Top45	21,562	30.59	45.74	+17.30	+11.99	+17.11	+25.86	+15.16	+3.79	**+18.32**	+32.97
Top60	19,241	30.20	29.14	+1.09	+0.35	+0.39	+0.63	-1.05	-0.43	-0.23	+1.72

4.3 Instructions help the model return candidates that match the request

To test whether instructions help preferred candidates rank higher, we compare three conditions: Reference generates without instructions, Matched uses the intended instruction, and Shuffled uses one expressing a different preference. We first compare Matched with Reference to measure the

benefit of instruction guidance. At 65M Top30, P@1 increases from 32.93% to 48.83%, while N@10 decreases by 34.57 percentage points (Table 1). RIGS more often ranks a preferred candidate first. Its top 10 outputs are also less likely to include an avoided candidate. We then compare Matched with Shuffled to test whether this benefit depends on the instruction's content. At 65M Top30, Matched improves P@1 by 18.52 percentage points over Shuffled. Compared with Shuffled, Matched generates about five more preferred and five fewer avoided outputs per 100 samples on average (Table 20). These comparisons show that RIGS can steer precursor generation according to the instruction's content.

Instruction-following gains vary with training support in a consistent but non-monotone pattern across model capacities. On observed USPTO-MR support, Matched–Shuffled P@1 gains are near zero. They increase at Top15 and are largest at Top30/Top45 (Table 1). At Top30, the gains over unguided generation are concentrated at Top-1: the 65M model improves by 15.90 percentage points at Top-1 and 4.57 at Top-10, while the 8M model improves at Top-1 with little change at Top-10 (Appendix B.2). At Top60, Matched–Shuffled gains decline at both capacities, with little improvement for the 65M model across all three Stage-B seeds (Appendix C.4). Additional analyses examine how guidance strength affects instruction control and output validity and compare supports under alternative training budgets (Appendices C.1–C.3 and B.3).

4.4 Roles of grounding and generation-time guidance

We show what Stage A grounding adds to direct Stage-B training and demonstrate the benefits of guiding generation over simply reranking unguided outputs.

Stage-A grounding. We compare 8M Top30 models with Stage A grounding (RIGS) and without it (Direct Stage-B), using the same Stage-B training and inference settings. On a separate test cohort, Stage A grounding increases the Matched–Shuffled reduction in N@10 by 6.82 percentage points (Table 2). Stage A trains the projector to score preferred candidates above avoided ones for the same instruction. Consistent with this objective, grounding mainly helps suppress candidates that conflict with the request (Table 15).

Guidance during generation. We compare guidance during generation with post-hoc reranking of unguided Reference outputs, using 100 samples per group for both. RIGS ranks its guided outputs by sampling frequency; the reranker uses a scalar scorer trained on signed RDKit utility. A preferred candidate appears among the samples in about 64% of groups for both methods, but RIGS achieves higher P@1: 32.73% versus 22.23% (Table 3). The reranker lowers N@1, yet its P@1 also falls below the unguided baseline's 24.57%. Thus, this reranker suppresses avoided candidates at the cost of selecting fewer known preferred candidates. RIGS improves both outcomes over unguided generation. Appendix E.1 gives the selector details and full comparison.

Table 2: **Effect of Stage-A grounding.** On a separate test cohort, we compare 8M Top30 models with and without Stage-A pretraining. Each row shows Matched–Shuffled gains in percentage points; higher is better (Appendix D.1).

Variant	ΔP @1	$-\Delta$N @1	$-\Delta$N @10
RIGS	+12.77	+10.69	+21.21
Direct Stage-B	+11.96	+7.67	+14.39

Table 3: **Generation-time guidance versus reranking.** We use 100 raw draws per group for Top30. Hit_+@100 reports how often they include a preferred candidate. Appendix E.1 details the separate cohort.

Method	Hit_+@100 (%)	P@1 $\uparrow$ (%)	N@1 $\downarrow$ (%)
Reference	64.80	24.57	25.39
Reranking	64.80	22.23	**3.32**
RIGS	63.87	**32.73**	8.75

4.5 Applicability across datasets and guidance suites

We assess the generality of RIGS across reaction datasets and guidance types. We also present a case study illustrating how RIGS supports multistep retrosynthetic planning.

Cross-dataset evaluation on Pistachio. To assess the generality of RIGS across reaction datasets, we separately train a 65M Top30 model on Pistachio, a commercial reaction database (Mayfield,

Table 4: **Control across datasets and guidance suites.** We evaluate separately trained Top30 models for each dataset and guidance suite. Positive values indicate better instruction alignment. Cohort and aggregation details for Pistachio, Structure, and Metric4 can be found in Appendices G.1, F.1, and G.2, respectively.

			Matched–Shuffled (pp)		Matched–Reference (pp)	
Reaction data	Guidance	Backbone	ΔP@1 ↑	$-\Delta$N@10 ↑	ΔP@1 ↑	$-\Delta$N@10 ↑
Pistachio	RDKit5	65M	+1.52	+7.19	+0.98	+13.59
USPTO	Structure	8M	+3.55	+3.43	+5.54	+10.57
		65M	+8.09	+18.44	+14.74	+35.79
USPTO	Metric4	8M	+5.37	+7.41	+4.07	+8.04
		65M	+15.78	+21.92	+13.44	+27.52

2021). We construct the Top30 training support using the same template library as in the USPTO experiments. We evaluate on a fixed cohort of held-out Pistachio products with recorded precursor alternatives. Details of cohort construction can be found in Appendix G.1. Matched improves P@1 by 1.52 percentage points and reduces N@10 by 7.19 points relative to Shuffled, with improvements over Reference as well (Table 4). These results support the generality of RIGS across reaction datasets.

Case-specific structural instructions. We use the *Structure* guidance dataset to test whether RIGS can follow free-form natural-language instructions. We generate case-specific instructions from structural evidence, describing molecular features to retain or avoid. On the test guidance set of 4,606 distinct instructions (Appendix F.1), Matched instructions promote preferred candidates and suppress avoided candidates relative to Shuffled at both model capacities (Table 4). At 65M, P@1 improves by 8.09 percentage points and N@10 decreases by 18.44 points. These gains support RIGS's ability to follow case-specific structural requests expressed in free-form natural language.

Synthesis-related guidance. We use the *Metric4* guidance dataset to test whether RIGS can follow instructions about synthesis-related properties. Metric4 combines four metrics: reduction in synthetic complexity, procurement burden, halogen count, and ring count (Appendix H.5). The Top30 results show preferred-candidate promotion and avoided-candidate suppression at both capacities (Table 4). Across all synthetic supports, we observe improved instruction following in seven of eight settings (Appendix G.2).

Multistep planning. To examine how RIGS can support multistep retrosynthetic planning, we use it as the single-step precursor generator in a custom value-guided AND–OR tree-search planner inspired by Retro* (Chen et al., 2020). The planner recursively expands precursor molecules to construct a synthesis route. In this case study, conditioning each expansion on the instruction to use fewer halogen atoms yields a complete route with halogen-free starting materials. This illustrates how RIGS can guide precursor selection across multiple retrosynthetic steps (Figure 9B in Appendix F.3).

5 Conclusion

We studied how single-step retrosynthesis can generate precursor alternatives that respond to a chemist's preferences. RIGS grounds instructions in preferences among alternatives for the same product, then transfers the learned representation to a frozen graph generator through lightweight residual adapters. Across nested training supports and two model capacities, synthetic expansion improves recovery of known alternatives over observed-reaction training, but further expansion does not monotonically improve coverage or instruction control. Matched-versus-shuffled comparisons show that RIGS can promote preferred candidates and suppress conflicting ones, with Stage-A grounding providing its clearest additional benefit in conflict suppression. Evaluations with case-specific structural instructions further support control beyond predefined molecular descriptors. The results reveal a consistent but non-monotone interaction between coverage and control across model capacities.

Ethics Statement

This work is guided by the ICLR Code of Ethics. The study uses patent-derived reaction data, including USPTO-MR, its template-expanded variants, and Pistachio, and involves no human participants, animal experimentation, or personally identifiable information. Dataset access and redistribution are subject to the respective usage and licensing terms; Pistachio records are not redistributed. We recognize that patent coverage and automated annotations may introduce biases and limit generalization. Instruction alignment and graph validity do not establish chemical feasibility or safety; proposed reactions require qualified chemical and safety review before experimental use.

Reproducibility Statement

We provide detailed descriptions of RIGS and its experimental setup to support reproduction of the reported results. Sections 3–4 describe the two-stage grounding and steering framework and its evaluation. Appendices H, I.1, and I.7 document data construction, frozen encoder inputs, objective coefficients, training budgets, and inference settings. Appendix A specifies the shared evaluation inventory and frozen evaluation coordinates; Supplementary Section S3 details the validation-based selection rules. The accompanying reproducibility materials also provide diagnostics, numerical results, and case studies.

The accompanying USPTO data package supplies constructed supervision, fixed splits, graph inputs, and evaluation inventories, allowing reproduction to start from the saved data without rerunning template expansion or annotation. Separate archives supply selected RDKit5 model states for the five supports at both capacities, and execution code and saved language/route features for an initial Top30/8M/RDKit5 recipe. A bounded check from the packaged distribution completed generation and scoring for 20 signed groups and 6,000 trajectories. The documentation distinguishes this interface check and verified data/state loading from full retraining and reproduction of the complete experimental matrix. Pistachio data are excluded; licensed access and the study-specific derived inputs are required for exact reproduction, with acquisition and input requirements documented separately.

AI Use Statement

We used generative AI tools to assist literature exploration, code review, experiment documentation, and language editing. The authors checked suggested citations against primary sources, reviewed and tested AI-assisted code, and verified analyses against saved experimental artifacts. The authors take responsibility for the final text, claims, code, and results.

References

Andres M. Bran, Théo A. Neukomm, Daniel Armstrong, Zlatko Jončev, and Philippe Schwaller. Chemical reasoning in LLMs unlocks strategy-aware synthesis planning and reaction mechanism elucidation. *Matter*, 9(5):102812, 2026. doi: 10.1016/j.matt.2026.102812. URL https://www.sciencedirect.com/science/article/pii/S259023852600175X.

Dominic M. T. Chan, Kevin L. Monaco, Ru-Ping Wang, and Michael P. Winters. New N- and O-arylations with phenylboronic acids and cupric acetate. *Tetrahedron Letters*, 39(19):2933–2936, 1998. doi: 10.1016/S0040-4039(98)00503-6. URL https://doi.org/10.1016/S0040-4039(98)00503-6.

Kavita H. Chandnani and Sampatraj B. Chandalia. Synthesis of m-phenoxybenzaldehyde starting from chlorobenzene and m-cresol: Some aspects of process development. *Organic Process Research & Development*, 3(6):416–424, 1999. doi: 10.1021/op990028z. URL https://pubs.acs.org/doi/10.1021/op990028z.

Benson Chen, Tianxiao Shen, Tommi S. Jaakkola, and Regina Barzilay. Learning to make generalizable and diverse predictions for retrosynthesis. *arXiv preprint arXiv:1910.09688*, 2019. URL https://arxiv.org/abs/1910.09688.

Binghong Chen, Chengtao Li, Hanjun Dai, and Le Song. Retro*: Learning retrosynthetic planning with neural guided A* search. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1608–1616. PMLR, 2020. URL https://proceedings.mlr.press/v119/chen20k.html.

Connor W. Coley, Luke Rogers, William H. Green, and Klavs F. Jensen. SCScore: Synthetic complexity learned from a reaction corpus. *Journal of Chemical Information and Modeling*, 58(2): 252–261, 2018. doi: 10.1021/acs.jcim.7b00622. URL https://doi.org/10.1021/acs.jcim.7b00622.

Connor W. Coley, William H. Green, and Klavs F. Jensen. RDChiral: An RDKit wrapper for handling stereochemistry in retrosynthetic template extraction and application. *Journal of Chemical Information and Modeling*, 59(6):2529–2537, 2019. doi: 10.1021/acs.jcim.9b00286. URL https://doi.org/10.1021/acs.jcim.9b00286.

Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 375–413. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.26. URL https://aclanthology.org/2022.emnlp-main.26/.

David A. Evans, Jeffrey L. Katz, and Theodore R. West. Synthesis of diaryl ethers through the copper-promoted arylation of phenols with arylboronic acids: An expedient synthesis of thyroxine. *Tetrahedron Letters*, 39(19):2937–2940, 1998. doi: 10.1016/S0040-4039(98)00502-4. URL https://doi.org/10.1016/S0040-4039(98)00502-4.

Yin Fang, Xiaozhuan Liang, Ningyu Zhang, Kangwei Liu, Rui Huang, Zhuo Chen, Xiaohui Fan, and Huajun Chen. Mol-Instructions: A large-scale biomolecular instruction dataset for large language models. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/d346d91999074dd8d6073d4c3b13733b-Abstract-Conference.html.

Yves Grandjean, David Kreutter, and Jean-Louis Reymond. Data augmentation in a triple transformer loop retrosynthesis model. *Digital Discovery*, 5:653–661, 2026. doi: 10.1039/D5DD00465A. URL https://pubs.rsc.org/en/content/articlelanding/2026/dd/d5dd00465a.

Friedrich Hastedt, Dongda Zhang, and Antonio del Rio-Chanona. From feasible to practical: Pareto-optimal synthesis planning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2605.07521v2.

Ilia Igashov, Arne Schneuing, Marwin Segler, Michael Bronstein, and Bruno Correia. RetroBridge: Modeling retrosynthesis with Markov bridges. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/acb94e709f02895fd98b5867f0b184f3-Abstract-Conference.html.

Andrei V. Iosub and Shannon S. Stahl. Palladium-catalyzed aerobic oxidative dehydrogenation of cyclohexenes to substituted arene derivatives. *Journal of the American Chemical Society*, 137(10):3454–3457, 2015. doi: 10.1021/ja512770u. URL https://doi.org/10.1021/ja512770u.

Chenglong Kang, Xiaoyi Liu, and Fei Guo. RetroInText: A multimodal large language model enhanced framework for retrosynthetic planning via in-context representation learning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/b2fbf1c9bc92e7ef2f6cab2e8a3e09af-Abstract-Conference.html.

Najwa Laabid and Vikas K Garg. Beyond search: Direct model guidance for steerable synthesis planning, 2026. URL https://openreview.net/forum?id=O4Tv5Cmpbv.

Najwa Laabid, Severi Rissanen, Markus Heinonen, Arno Solin, and Vikas Garg. Equivariant denoisers cannot copy graphs: Align your graph diffusion models. In *International Conference on Learning Representations*, 2025. URL

https://proceedings.iclr.cc/paper_files/paper/2025/hash/f8f78f8043f35890181a824e53a57134-Abstract-Conference.html.

Pengxiang Ling, Dao Li, and Xingyi Wang. Supported CuO/γ-Al_2O_3 as heterogeneous catalyst for synthesis of diaryl ether under ligand-free conditions. *Journal of Molecular Catalysis A: Chemical*, 357:112–116, 2012. doi: 10.1016/j.molcata.2012.01.028. URL https://www.sciencedirect.com/science/article/pii/S1381116912000489.

Shengchao Liu, Weili Nie, Chengpeng Wang, Jiarui Lu, Zhuoran Qiao, Ling Liu, Jian Tang, Chaowei Xiao, and Animashree Anandkumar. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5:1447–1457, 2023. doi: 10.1038/s42256-023-00759-6. URL https://www.nature.com/articles/s42256-023-00759-6.

Songtao Liu, Hanjun Dai, Yue Zhao, and Peng Liu. Preference optimization for molecule synthesis with conditional residual energy-based models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30929–30945. PMLR, 2024a. URL https://proceedings.mlr.press/v235/liu24n.html.

Yifeng Liu, Hanwen Xu, Tangqi Fang, Haocheng Xi, Zixuan Liu, Sheng Zhang, Hoifung Poon, and Sheng Wang. T-Rex: Text-assisted retrosynthesis prediction. *arXiv preprint arXiv:2401.14637*, 2024b. URL https://arxiv.org/abs/2401.14637.

Daniel Lowe. Chemical reactions from US patents (1976–Sep2016). figshare, 2017. URL https://doi.org/10.6084/m9.figshare.5104873.

John Mayfield. 13,118,970 reactions and counting. NextMove Software, 2021. URL https://nextmovesoftware.com/blog/2021/03/24/13118970-reactions-and-counting/.

Anton Morgunov and Victor S. Batista. Project Ariadne: Prompt-conditioned route generation for synthesis planning. *arXiv preprint arXiv:2606.24184*, 2026. URL https://arxiv.org/abs/2606.24184.

Xuan-Vu Nguyen, Daniel Armstrong, Milena Wehrbach, Andres M. Bran, Zlatko Jončev, and Philippe Schwaller. Synthelite: Chemist-aligned and feasibility-aware synthesis planning with LLMs. *arXiv preprint arXiv:2512.16424*, 2025. URL https://arxiv.org/abs/2512.16424.

RDKit contributors. RDKit: Open-source cheminformatics, n.d. URL https://www.rdkit.org.

Tatsuya Sagawa and Ryosuke Kojima. ReactionT5: A pre-trained transformer model for accurate chemical reaction prediction with limited data. *Journal of Cheminformatics*, 17(1): 126, 2025. doi: 10.1186/s13321-025-01075-4. URL https://doi.org/10.1186/s13321-025-01075-4.

Yair Schiff, Subham Sekhar Sahoo, Hao Phung, Guanghan Wang, Sam Boshar, Hugo Dalla-Torre, Bernardo P. de Almeida, Alexander Rush, Thomas Pierrot, and Volodymyr Kuleshov. Simple guidance mechanisms for discrete diffusion models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/6cc31b44d88dce8380d36e81485cd07f-Abstract-Conference.html.

Philippe Schwaller, Benjamin Hoover, Jean-Louis Reymond, Hendrik Strobelt, and Teodoro Laino. Extraction of organic chemistry grammar from unsupervised learning of chemical reactions. *Science Advances*, 7(15):eabe4166, 2021. doi: 10.1126/sciadv.abe4166. URL https://doi.org/10.1126/sciadv.abe4166.

Marwin H. S. Segler, Mike Preuss, and Mark P. Waller. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature*, 555:604–610, 2018. doi: 10.1038/nature25978. URL https://doi.org/10.1038/nature25978.

Keith Smith and Dennis Jones. A superior synthesis of diaryl ethers by the use of ultrasound in the Ullmann reaction. *Journal of the Chemical Society, Perkin Transactions 1*, pp. 407–408, 1992. doi: 10.1039/P19920000407. URL https://pubs.rsc.org/en/content/articlelanding/1992/p1/p19920000407.

Xiaozhuang Song, Xuanhao Pan, Xinjian Zhao, Hangting Ye, Shufei Zhang, Jian Tang, and Tianshu Yu. AOT*: Efficient synthesis planning via LLM-empowered AND-OR tree search. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34727–34758. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.1734. URL https://aclanthology.org/2026.findings-acl.1734/.

Yichen Tan, José María Muñoz-Molina, Gregory C. Fu, and Jonas C. Peters. Oxygen nucleophiles as reaction partners in photoinduced, copper-catalyzed cross-couplings: O-arylations of phenols at room temperature. *Chemical Science*, 5:2831–2835, 2014. doi: 10.1039/C4SC00368C. URL https://pubs.rsc.org/en/content/articlelanding/2014/sc/c4sc00368c.

Amol Thakkar, Alain C. Vaucher, Andrea Byekwaso, Philippe Schwaller, Alessandra Toniato, and Teodoro Laino. Unbiasing retrosynthesis language models with disconnection prompts. *ACS Central Science*, 9(7):1488–1498, 2023. doi: 10.1021/acscentsci.3c00372. URL https://pubs.acs.org/doi/10.1021/acscentsci.3c00372.

Alessandra Toniato, Alain C. Vaucher, Philippe Schwaller, and Teodoro Laino. Enhancing diversity in language based models for single-step retrosynthesis. *Digital Discovery*, 2:489–501, 2023. doi: 10.1039/D2DD00110A. URL https://pubs.rsc.org/en/content/articlelanding/2023/dd/d2dd00110a.

Suong B. A. Tran, Jihye Roh, and Connor W. Coley. Quantifying the failure modes of current one-step retrosynthesis models. *Chemical Science*, 17(28):14054–14063, 2026. doi: 10.1039/D6SC01323F. URL https://pubs.rsc.org/en/content/articlehtml/2026/sc/d6sc01323f.

Zhengkai Tu, Sourabh J. Choure, Mun Hong Fong, Jihye Roh, Itai Levin, Kevin Yu, Joonyoung F. Joung, Nathan Morgan, Shih-Cheng Li, Xiaoqi Sun, Huiqian Lin, Mark Murnin, Jordan P. Liles, Thomas J. Struble, Michael E. Fortunato, Mengjie Liu, William H. Green, Klavs F. Jensen, and Connor W. Coley. ASKCOS: Open-source, data-driven synthesis planning. *Accounts of Chemical Research*, 58(11):1764–1775, 2025. doi: 10.1021/acs.accounts.5c00155. URL https://pubs.acs.org/doi/10.1021/acs.accounts.5c00155.

Chenguang Wang, Zihan Zhou, Lei Bai, and Tianshu Yu. Order matters in retrosynthesis: Structure-aware generation via reaction-center-guided discrete flow matching. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://openreview.net/forum?id=xeH37plPS3.

Yiming Wang, Yuxuan Song, Yiqun Wang, Minkai Xu, Rui Wang, Hao Zhou, and Wei-Ying Ma. RetroDiff: Retrosynthesis as multi-stage distribution interpolation. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1918–1926. PMLR, 2025. URL https://proceedings.mlr.press/v258/wang25e.html.

Annie M. Westerlund, Lakshidaa Saigiridharan, and Samuel Genheden. Human-guided synthesis planning via prompting. *Chemical Science*, 16:14655–14667, 2025. doi: 10.1039/D5SC00927H. URL https://pubs.rsc.org/en/content/articlelanding/2025/sc/d5sc00927h.

Robin Yadav, Qi Yan, Guy Wolf, Avishek Joey Bose, and Renjie Liao. RETRO SYNFLOW: Discrete flow-matching for accurate and diverse single-step retrosynthesis. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/4f86833d5cc98ec32e470ef1c8cb82e3-Abstract-Conference.html.

Kevin Yu, Jihye Roh, Ziang Li, Wenhao Gao, Runzhong Wang, and Connor W. Coley. Double-ended synthesis planning with goal-constrained bidirectional search. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/cd091a4d8e97157d32940428f902c7b0-Paper-Conference.pdf.

Bogdan Zagribelnyy, Ivan Ilin, Maksim Kuznetsov, Nikita Bondarev, Mathieu Reymond, Roman Schutski, Thomas MacDougall, Rim Shayakhmetov, Zulfat Miftakhutdinov, Mikolaj Mizera, Vladimir Aladinskiy, Alex Aliper, and Alex Zhavoronkov. When single answer is not enough: Rethinking single-step retrosynthesis benchmarks for LLMs. *arXiv preprint arXiv:2602.03554v2*, 2026. URL https://arxiv.org/abs/2602.03554v2.

Xu Zhang, Yiming Mo, Wenguan Wang, and Yi Yang. Retrosynthesis prediction enhanced by in-silico reaction data augmentation. *arXiv preprint arXiv:2402.00086*, 2024. URL https://arxiv.org/abs/2402.00086.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025. URL https://arxiv.org/abs/2506.05176.

APPENDIX CONTENTS

A Common-inventory protocol and frozen coordinates . . . 16

B Common Top100 results and diagnostics . . . 17
- B.1 Absolute arm results . . . 17
- B.2 Returned-list contrasts with the unguided model . . . 17
- B.3 Common-update-ceiling support sensitivity . . . 18
- B.4 Relaxed round-trip coverage and accuracy across candidate ranks . . . 19

C Calibration and Stage-B seed robustness . . . 20
- C.1 Validation guidance-strength trade-off . . . 21
- C.2 Matched residual-scale sensitivity . . . 21
- C.3 Three-seed frozen-checkpoint inference-scale response . . . 22
- C.4 Complete three-seed analysis across supports and capacities . . . 23

D Controlled method alternatives . . . 24
- D.1 Direct Stage-B: effect of Stage-A-aligned initialization . . . 24
- D.2 Stage-B loss ablation . . . 25

E Proposal and selection counterfactuals . . . 26
- E.1 Budgeted proposal–selection counterfactual . . . 26

F Language and chemistry evidence . . . 26
- F.1 Free-form structural instructions and the frozen language representation . . . 26
- F.2 A chemistry-audited paired case: fewer precursor halogens . . . 28
- F.3 Downstream multistep planning witnesses . . . 29
- F.4 Direct empirical sampled-mass reallocation . . . 30

G External and heterogeneous robustness . . . 30
- G.1 Pistachio Top30: control on recorded reaction alternatives . . . 30
- G.2 Applicability to another metric suite . . . 32

H Data, checkpoint, and evaluation provenance . . . 33
- H.1 Identifiers, data units, and reporting scope . . . 33
- H.2 Construction of USPTO-MR . . . 34
- H.3 Base-generator data matrix . . . 34
- H.4 Synthetic precursor construction and ranking . . . 34
- H.5 Guidance suites and construction . . . 36
- H.6 Guidance-data matrix . . . 36
- H.7 Frozen evaluation cohorts . . . 36
- H.8 Base-generator checkpoint provenance . . . 38
- H.9 Sampling and statistical protocol . . . 38

I Architecture, objectives, and implementation . . . 39
- I.1 Frozen Stage-A representations . . . 39
- I.2 Structure annotation and quality checks . . . 39
- I.3 Stage-A grounding and exact route scoring . . . 40
- I.4 Stage-B residual adapter path . . . 40
- I.5 Stage-B loss components and metric teacher . . . 40
- I.6 Notation . . . 42
- I.7 Implementation and reproducibility details . . . 42

J Limitations and Future Work . . . 43

A COMMON-INVENTORY PROTOCOL AND FROZEN COORDINATES

Controlled support sequence. Top15/30/45/60 are nested synthetic-only prefixes of one product-local candidate ordering. USPTO-MR is a separate natural-data reference. The primary training recipe uses within-family data passes, so realized update counts differ with support size. Exact budgets appear in Appendix I.7; the common-update-ceiling sensitivity appears in Appendix B.3.

Validation calibration. Calibration targets high Matched P@1, with trajectory validity informing scale and checkpoint choice. The primary study retains each run's recorded validation-selected coordinate (Table 5). Supplementary Section S3 documents the support-specific selection rules, including validation gates, fallback decisions, and recorded amendments.

Table 5: Frozen coordinates for the ten USPTO–Guide(RDKit5)–8M–[Support] and USPTO–Guide(RDKit5)–65M–[Support] models in the Common Top100 study. We selected all coordinates on validation before opening the common test; every row uses Reference, Matched, and Shuffled arms.

Guided model ID	Stage-B epoch	α_{eval}
USPTO–Guide(RDKit5)–8M–USPTO-MR	14	.50
USPTO–Guide(RDKit5)–8M–Top15	6	.20
USPTO–Guide(RDKit5)–8M–Top30	12	.50
USPTO–Guide(RDKit5)–8M–Top45	14	.50
USPTO–Guide(RDKit5)–8M–Top60	6	.25
USPTO–Guide(RDKit5)–65M–USPTO-MR	14	.50
USPTO–Guide(RDKit5)–65M–Top15	1	.50
USPTO–Guide(RDKit5)–65M–Top30	14	.50
USPTO–Guide(RDKit5)–65M–Top45	8	.50
USPTO–Guide(RDKit5)–65M–Top60	4	.05

Frozen inventory and sampling. The common test contains 1,201 cases, 1,280 paired semantic families, and 2,560 signed groups balanced over five RDKit axes and both polarities. All arms use the shared sampler in Appendix H.9. The ten-model, three-arm study contains 7,680,000 trajectories. Table 28 gives construction-stage accounting.

Evaluation metrics. For each product–instruction group, the raw budget B counts every draw, including invalid outputs. $\text{Hit}_{\text{all}}@B$ is the rate of groups whose first B draws recover any candidate in the frozen inventory; $\text{Hit}_{+}@B$ restricts this event to the preferred set. Reference coverage tables abbreviate $\text{Hit}_{\text{all}}@B$ as Hit@B; the proposal–selection diagnostic uses $\text{Hit}_{+}@B$. Budget curves use $B \in \{1, 5, 10, 25, 50, 100\}$. For P@$k$ and N@k, we use all 100 raw draws per group, canonicalize and deduplicate valid outputs, and rank the resulting precursor sets by decreasing draw frequency. P@k and N@k are the rates of groups whose top k ranked candidates contain at least one preferred or avoided candidate, respectively; they are set-hit rates, not the fraction of matching candidates in the list.

Raw validity V is the fraction of draws yielding a precursor set that passes RDKit parsing and canonicalization. If n_g draws are valid and u_g are distinct, the group's duplicate rate is $(n_g - u_g)/B$; unique-valid count averages u_g with the stated cohort weights. Pooled recovery counts distinct product–candidate pairs recovered from the known inventory across all groups. Validity concerns molecular representation, not experimental feasibility.

Contrasts and uncertainty. We report rates as percentages and their differences as percentage points; counts retain their natural units. In three-arm comparisons, Δ subtracts the named comparator from Matched: Shuffled tests instruction identity, while Reference measures the net guidance effect. Positive ΔP and negative ΔN are favorable; $-\Delta$N reverses the suppression sign for readability. ΔV retains the signed validity difference. Aggregation and case-cluster confidence intervals (CIs) follow Appendix H.9; Stage-B seeds 42–44 form a separate robustness analysis.

Data provenance and test overlap. The candidate audit confirms product-wise nesting of the synthetic supports. We retain a 2.03% Top60 catalog-version extension in the provenance records. We inspected earlier Axis4 test reports during protocol development. The named earlier cohort contains 988 products and shares 154 with the 1,201-product Common Top100 cohort (12.82% of Common Top100). Thus the later inventory and generated outputs are new, but the test products are not wholly independent of that exploratory evaluation. We fixed the reported model coordinates before inspecting the Common Top100 outputs.

B COMMON TOP100 RESULTS AND DIAGNOSTICS

A more diverse sampler need not recover known alternatives more often in its first draw: Top60 produces the most distinct valid outputs at $B = 100$ but has the lowest Hit@1 among synthetic-support models at each capacity (Table 6).

Table 6: Reference coverage at the endpoints of the Common Top100 budget curves. Hit, validity and duplicate rate are percentages; pooled recovery counts distinct case–candidate pairs. Rates and mean counts use equal-axis, case-balanced aggregation at both capacities. All columns except Hit@1 use $B = 100$, without backfilling.

Support	Hit@1	Hit@100	Pooled	Unique valid /group	Validity	Duplicate rate
GraphDiT-8M						
USPTO-MR	6.17	59.30	2,348	37.475	63.49	26.01
Top15	21.91	80.82	12,880	44.044	68.38	24.34
Top30	18.91	82.89	16,922	53.074	68.09	15.01
Top45	16.56	83.24	17,370	55.828	67.36	11.53
Top60	14.77	82.77	17,378	57.067	66.14	9.07
GraphDiT-65M						
USPTO-MR	5.08	52.93	1,723	38.938	63.47	24.53
Top15	25.74	81.48	13,661	40.836	69.56	28.72
Top30	24.02	84.96	20,075	49.615	69.21	19.59
Top45	21.84	84.18	21,562	54.159	69.18	15.02
Top60	16.48	83.16	19,241	56.956	67.10	10.14

Table 7 retains the early- and late-rank instruction-content contrasts. Top30 and Top45 show clear preferred-route promotion and avoided-route suppression at both capacities; 65M Top60 remains statistically unresolved at these operating points. For the primary 8M comparisons, Top30–Top45 paired intervals include zero, whereas Top30 improves over Top15 and Top60 weakens relative to Top45. Per-axis effects are not uniform, so the primary estimand gives each axis equal weight. Matched–Shuffled validity intervals span zero across all supports; at $B = 100$, intervals for any-inventory availability and unique-valid breadth also span zero for every expanded support. Together, these results locate the instruction effect mainly in the composition and ranking of sampled outputs.

B.1 ABSOLUTE ARM RESULTS

The absolute arms in Table 8 expose costs shared by Matched and Shuffled: similar validity between those arms can coexist with a substantial loss relative to Reference. Entries use the primary equal-axis, case-balanced aggregation and the sampling protocol in Appendix H.9. Native Structure results appear in Appendix F.1; Metric4 results appear in Appendix G.2.

B.2 RETURNED-LIST CONTRASTS WITH THE UNGUIDED MODEL

Table 9 expands the Matched–Reference columns of Table 1 with paired intervals at the frozen primary coordinates, using the aggregation and sampling protocol in Appendix H.9. Early-rank gains need not extend to Top-10: guidance can move an already available preferred route to the front. Avoided-route suppression can remain strong across the list even when later-rank promotion is small.

Table 7: Matched–Shuffled effects at returned ranks 1 and 10 on Common Top100. Entries are percentage-point estimates with 95% paired case-cluster bootstrap intervals, using equal-axis, case-balanced aggregation. Positive ΔP and negative ΔN are favorable. The registered primary coordinates use Stage-B seed 42.

Support	ΔP@1	ΔP@10	ΔN@1	ΔN@10
GraphDiT-8M				
USPTO-MR	$+0.08$ $[-0.66, +0.82]$	$+2.03$ $[+1.05, +3.05]$	-0.74 $[-1.48, 0.00]$	-1.37 $[-2.42, -0.35]$
Top15	$+4.10$ $[+2.85, +5.31]$	$+1.37$ $[+0.55, +2.19]$	-3.83 $[-4.92, -2.70]$	-4.65 $[-5.70, -3.59]$
Top30	$+12.89$ $[+11.02, +14.77]$	$+10.70$ $[+9.22, +12.19]$	-14.06 $[-15.70, -12.42]$	-18.52 $[-20.27, -16.80]$
Top45	$+14.18$ $[+12.46, +15.94]$	$+9.53$ $[+8.12, +10.94]$	-14.06 $[-15.70, -12.42]$	-18.79 $[-20.47, -17.11]$
Top60	$+5.63$ $[+4.06, +7.15]$	$+4.88$ $[+3.67, +6.13]$	-5.94 $[-7.46, -4.45]$	-8.36 $[-9.73, -6.95]$
GraphDiT-65M				
USPTO-MR	$+0.00$ $[-0.63, +0.63]$	$+0.23$ $[-0.63, +1.09]$	$+0.20$ $[-0.39, +0.78]$	-1.60 $[-2.46, -0.78]$
Top15	$+11.29$ $[+9.49, +13.13]$	$+7.50$ $[+6.21, +8.79]$	-12.66 $[-14.30, -11.02]$	-13.83 $[-15.31, -12.31]$
Top30	$+18.52$ $[+16.56, +20.43]$	$+12.19$ $[+10.78, +13.59]$	-17.89 $[-19.61, -16.21]$	-27.19 $[-28.91, -25.47]$
Top45	$+17.30$ $[+15.39, +19.22]$	$+11.99$ $[+10.47, +13.56]$	-17.11 $[-18.79, -15.43]$	-25.86 $[-27.62, -24.06]$
Top60	$+1.09$ $[-0.23, +2.42]$	$+0.35$ $[-0.70, +1.45]$	-0.39 $[-1.72, +0.94]$	-0.63 $[-1.76, +0.55]$

B.3 Common-update-ceiling support sensitivity

To examine sensitivity to support-dependent training budgets, we retrained Top15, Top30, and Top45 with common Stage-A/B training ceilings of 125,850/58,730 optimizer updates (effective batch 32, seed 42). We selected each deployment checkpoint and residual scale on validation before inspecting the corresponding Common Top100 outputs. Top60 reuses its primary checkpoint. This sensitivity equalizes adapter-training ceilings, not the selected update counts or the costs of base training and historical reaction-pair pretraining.

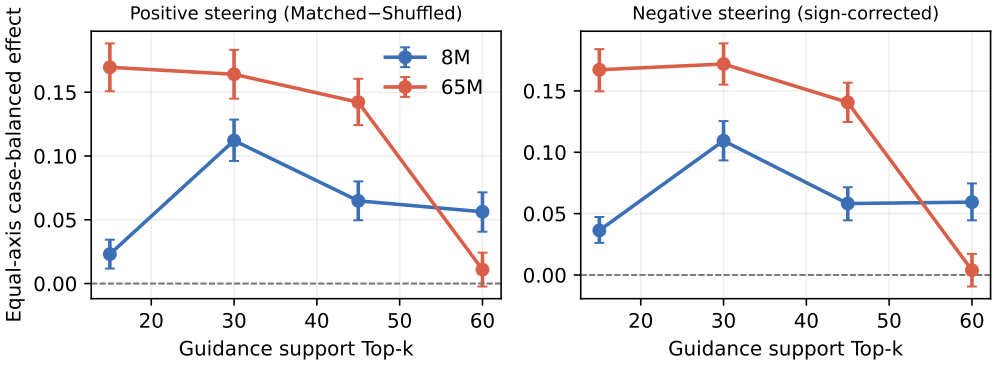


Figure 4: **Common-update-ceiling support-scaling sensitivity.** Points are equal-axis, case-balanced Matched–Shuffled effects on the frozen Common Top100 test, shown as probability differences (0.10 = 10 percentage points). Error bars are 95% shared case-cluster bootstrap intervals. The right panel plots $-\Delta N@1$ so that larger values are favorable. Top15/30/45 use the common update ceilings above; Top60 reuses its primary checkpoint.

Table 8: Key absolute-arm Common Top100 results (%) at the coordinates in Table 5. Each support lists all three conditions at the same raw-draw budget $B = 100$.

Support	Arm	P@1	P@10	N@1	N@10	Validity
GraphDiT-8M						
USPTO-MR	Reference	8.477	25.547	8.555	25.391	63.488
	Matched	8.984	27.461	7.695	22.852	60.816
	Shuffled	8.906	25.430	8.438	24.219	60.913
Top15	Reference	30.156	58.789	30.117	58.242	68.382
	Matched	34.102	59.062	26.055	53.086	67.084
	Shuffled	30.000	57.695	29.883	57.734	67.148
Top30	Reference	30.195	58.281	29.844	56.484	68.086
	Matched	40.195	57.617	11.602	28.750	52.605
	Shuffled	27.305	46.914	25.664	47.266	52.712
Top45	Reference	29.844	56.328	28.789	55.664	67.356
	Matched	40.625	58.945	13.594	30.742	62.611
	Shuffled	26.445	49.414	27.656	49.531	62.457
Top60	Reference	29.219	55.312	27.539	53.867	66.138
	Matched	33.047	56.484	21.992	43.789	64.696
	Shuffled	27.422	51.602	27.930	52.148	64.700
GraphDiT-65M						
USPTO-MR	Reference	7.695	21.562	7.422	21.602	63.470
	Matched	8.164	23.867	7.500	20.898	63.163
	Shuffled	8.164	23.633	7.305	22.500	63.104
Top15	Reference	30.234	61.055	31.406	61.367	69.555
	Matched	41.250	63.477	16.406	41.797	60.367
	Shuffled	29.961	55.977	29.063	55.625	60.230
Top30	Reference	32.930	63.281	29.727	62.734	69.209
	Matched	48.828	67.852	13.164	28.164	68.012
	Shuffled	30.312	55.664	31.055	55.352	67.897
Top45	Reference	30.586	61.211	30.273	60.234	69.182
	Matched	45.742	65.000	11.953	27.266	65.178
	Shuffled	28.438	53.008	29.063	53.125	65.086
Top60	Reference	30.195	55.586	27.695	55.391	67.099
	Matched	29.141	55.156	27.930	53.672	63.466
	Shuffled	28.047	54.805	28.320	54.297	63.362

Positive control survives the common training ceilings (Table 10), but the 8M peak shifts to Top30. Support choice therefore depends on training and selection; candidate count alone does not determine control strength. Reference and Top60 use the primary coordinates summarized in Table 8.

B.4 Relaxed round-trip coverage and accuracy across candidate ranks

This auxiliary analysis evaluates the Top30 65M reference, matched-guidance, and shuffled-guidance arms on the same 200 frozen test groups. The guided arms use the validation-selected epoch-14 checkpoint with $\alpha = 0.5$. For each group and arm, we reuse 100 reverse samples generated with 50 sampling steps, canonicalize and deduplicate valid precursor sets, and rank them by sample frequency, breaking ties by canonical SMILES. We evaluate the first $K \in \{1, 3, 5, 10\}$ available reverse candidates with a fixed ReactionT5 forward model (Sagawa & Kojima, 2025), using our locally fine-tuned checkpoint.

For candidate precursor set $r_{g,i}$ in group g, let x_g be the target product and $\mathcal{R}_g^{\text{ref}}$ the frozen collection of canonical positive-reference precursor sets associated with that semantic group. Let $F_1(r)$ be the top-ranked product from the fixed five-beam forward decoder. We accept only this Forward Top-1

Table 9: **Returned Top-k changes relative to unguided generation.** Cells give Matched–Reference effects in percentage points with 95% paired case-bootstrap intervals. We reverse the sign of avoided-set recovery; larger values are favorable in every column. Both capacities show early-rank preferred gains at Top15/30/45, while higher-rank effects vary.

Support	ΔP@1	ΔP@10	−ΔN@10
GraphDiT-8M			
USPTO-MR	$+0.51$ $[-0.23, +1.25]$	$+1.91$ $[+0.86, +2.97]$	$+2.54$ $[+1.45, +3.63]$
Top15	$+3.95$ $[+2.66, +5.16]$	$+0.27$ $[-0.55, +1.09]$	$+5.16$ $[+4.14, +6.17]$
Top30	$+10.00$ $[+8.09, +11.87]$	-0.66 $[-2.03, +0.74]$	$+27.73$ $[+25.90, +29.53]$
Top45	$+10.78$ $[+9.06, +12.50]$	$+2.62$ $[+1.33, +3.91]$	$+24.92$ $[+23.24, +26.72]$
Top60	$+3.83$ $[+2.27, +5.39]$	$+1.17$ $[-0.08, +2.42]$	$+10.08$ $[+8.71, +11.48]$
GraphDiT-65M			
USPTO-MR	$+0.47$ $[-0.27, +1.25]$	$+2.30$ $[+1.25, +3.40]$	$+0.70$ $[-0.31, +1.72]$
Top15	$+11.02$ $[+9.41, +12.66]$	$+2.42$ $[+1.29, +3.52]$	$+19.57$ $[+18.01, +21.09]$
Top30	$+15.90$ $[+14.14, +17.66]$	$+4.57$ $[+3.40, +5.78]$	$+34.57$ $[+32.77, +36.37]$
Top45	$+15.16$ $[+13.20, +17.11]$	$+3.79$ $[+2.38, +5.16]$	$+32.97$ $[+31.09, +34.80]$
Top60	-1.05 $[-2.50, +0.39]$	-0.43 $[-1.64, +0.78]$	$+1.72$ $[+0.47, +2.93]$

product, so the relaxed indicator for an available reverse candidate is

$$\chi_{g,i} = \mathbf{1}\big[F_1(r_{g,i}) = x_g \ \vee \ r_{g,i} \in \mathcal{R}_g^{\mathrm{ref}}\big]. \tag{7}$$

We use the same forward checkpoint, empty-reagent interface, and deterministic `num_beams=5`, `num_return_sequences=5` decoding throughout, and ignore beams below rank one. We fix the reverse-candidate ranking before forward evaluation.

For each arm, let $n_g(K)$ be the number of available candidates among the first K ranks in group g, and define $z_{g,i} = \chi_{g,i}$ when rank i is available and $z_{g,i} = 0$ otherwise. Table 11 reports fixed-slot RT-Accuracy and group-level RT-Coverage across the $G = 200$ groups:

$$A_{\mathrm{RT}}^{\mathrm{slot}}(K) = 100\,\frac{\sum_{g=1}^{G}\sum_{i=1}^{K} z_{g,i}}{GK}, \qquad C_{\mathrm{RT}}(K) = 100\,\frac{\sum_{g=1}^{G}\mathbf{1}[\sum_{i=1}^{K} z_{g,i} > 0]}{G}. \tag{8}$$

We count missing or invalid candidate ranks as failures and keep them in the accuracy denominator. At $K = 1$, RT-Accuracy and RT-Coverage are necessarily identical. Every entry retains the reference-precursor shortcut in Eq. (7).

Guidance concentrates relaxed round-trip hits at the first returned candidate: the $K = 1$ gain has a paired stratified bootstrap 95% interval of $[3.00, 13.50]$ points (10,000 resamples; seed 1042). This advantage fades deeper in the list. At $K = 10$, equal group coverage but lower fixed-slot accuracy means that guidance reaches as many groups with a hit while returning fewer successful candidates per group. This interpretation remains conditional on the relaxed rule, whose reference-precursor shortcut can accept a candidate without forward-model closure.

C Calibration and Stage-B seed robustness

We examine calibration at fixed checkpoints (Appendices C.1–C.3) and report the full support–capacity matrix across Stage-B seeds (Appendix C.4). Supplementary Section S3 records the selection rules and validation constraints; Appendix A lists the frozen evaluation coordinates.

Table 10: Common-update-ceiling deployment coordinates and effects. Selected B gives optimizer updates (Top60-equivalent epoch index); *reused* denotes the primary Top60 coordinates in Table 5. Effects are Matched minus Shuffled in percentage points [95% shared case-cluster CI]; larger ΔP@1 and smaller ΔN@1 are favorable. V denotes raw-draw validity.

Support	Selected B (coord.)	α	ΔP@1	ΔN@1	ΔV
8M					
Top15	58,730 (14)	.15	$+2.30$ $[+1.17, +3.44]$	-3.63 $[-4.73, -2.62]$	$+0.07$ $[-0.06, +0.20]$
Top30	33,560 (8)	.50	$+11.21$ $[+9.61, +12.85]$	-10.94 $[-12.54, -9.34]$	-0.05 $[-0.30, +0.20]$
Top45	50,340 (12)	.25	$+6.48$ $[+4.96, +8.01]$	-5.82 $[-7.15, -4.45]$	$+0.07$ $[-0.11, +0.26]$
Top60	reused	.25	$+5.62$ $[+4.06, +7.15]$	-5.94 $[-7.46, -4.45]$	-0.00 $[-0.21, +0.20]$
65M					
Top15	8,390 (2)	.50	$+16.95$ $[+15.08, +18.83]$	-16.72 $[-18.40, -14.96]$	$+0.10$ $[-0.26, +0.46]$
Top30	58,730 (14)	.50	$+16.41$ $[+14.49, +18.32]$	-17.19 $[-18.87, -15.51]$	$+0.16$ $[-0.18, +0.49]$
Top45	8,390 (2)	.50	$+14.22$ $[+12.42, +16.05]$	-14.06 $[-15.66, -12.46]$	$+0.09$ $[-0.30, +0.49]$
Top60	reused	.05	$+1.09$ $[-0.23, +2.42]$	-0.39 $[-1.72, +0.94]$	$+0.10$ $[-0.06, +0.27]$

Table 11: Relaxed Forward Top-1 round-trip results (%) for Top30 65M on 200 frozen test groups. Reverse K is the candidate-rank cutoff. Coverage is the fraction of groups with at least one round-trip hit; Accuracy uses the fixed $200K$ candidate-slot denominator, counting missing or invalid ranks as failures. Guidance denotes matched guidance, and Shuffled is the shuffled-guidance control. Δ is Guidance minus Reference in percentage points. Bold marks the highest point estimate for each metric and row. All results include the reference shortcut in Eq. (7).

	Reference		Guidance		Shuffled		Δ (pp)	
Reverse K	Cov.	Acc.	Cov.	Acc.	Cov.	Acc.	Cov.	Acc.
1	63.00	63.00	**71.00**	**71.00**	57.50	57.50	+8.00	+8.00
3	**76.50**	59.50	75.00	**62.83**	73.50	56.83	−1.50	+3.33
5	**79.00**	**57.90**	76.50	**57.90**	76.00	52.60	−2.50	0.00
10	**80.00**	**51.25**	**80.00**	47.85	77.00	42.90	0.00	−3.40

C.1 Validation guidance-strength trade-off

At inference, $\alpha = \alpha_{\text{eval}}$ scales the instruction residual applied to the frozen backbone; $\alpha = 0$ recovers the unguided Base model. Stronger intervention can favor preferred routes while disrupting valid generation (Figure 5), motivating calibration using both preference recovery and output validity.

These 65M sweeps use each support's original epoch-14 validation cohort and aggregates. Stars annotate final deployment scales evaluated here at epoch 14; deployed checkpoints may differ. The curves characterize each model's sensitivity to guidance strength, but do not isolate a causal effect of training support. They are separate from the common-inventory test diagnostics below.

C.2 Matched residual-scale sensitivity

We compare all five 8M supports at $\alpha_{\text{eval}} = .25$, retaining their seed-42 validation-selected checkpoints. This controls inference strength while preserving the support-specific trained models.

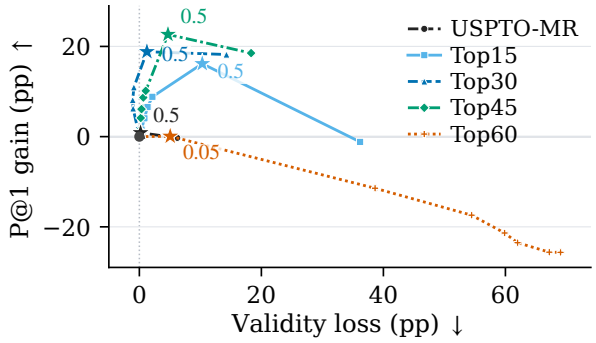


Figure 5: **Guidance-strength trade-off at fixed checkpoints.** GraphDiT-65M on each support's original validation cohort at epoch 14. P@1 gain and raw-validity loss are relative to $\alpha = 0$. Stars mark final selected scales evaluated at this checkpoint; deployed checkpoints may differ.

Table 12: Matched-scale 8M sensitivity at $\alpha_{\text{eval}} = .25$. Values are equal-axis, case-balanced Matched–Shuffled effects in percentage points. Positive ΔP and negative ΔN indicate better instruction alignment; ΔV is the Matched–Shuffled trajectory-validity difference.

Support	ΔP@1	ΔP@10	ΔN@1	ΔN@10	ΔV
USPTO-MR	+0.04	+0.63	-0.16	-1.25	-0.04
Top15	+4.73	+2.27	-4.69	-5.00	-0.05
Top30	+4.80	+3.28	-4.22	-8.67	-0.08
Top45	+6.60	+4.41	-6.37	-9.61	+0.01
Top60	+5.63	+4.88	-5.94	-8.36	+0.00

Matching α narrows the primary Top45–Top60 contrast: their paired 95% intervals include zero at all four recovery endpoints. The selected support ranking therefore depends partly on deployment strength; this comparison alone does not establish an intrinsic advantage for either support.

C.3 Three-seed frozen-checkpoint inference-scale response

For 8M Top45 and Top60, we hold each seed's validation-selected checkpoint fixed and vary $\alpha \in \{0, .05, .10, .15, .20, .25, .50, 1\}$, using seeds 42–44 and the paired test schedule from Appendix A. The grid is a post-hoc diagnostic and does not revise validation-selected operating points. Its archived estimand weights distinct cases equally after averaging their signed groups; the selected-point tables instead give each axis equal weight.

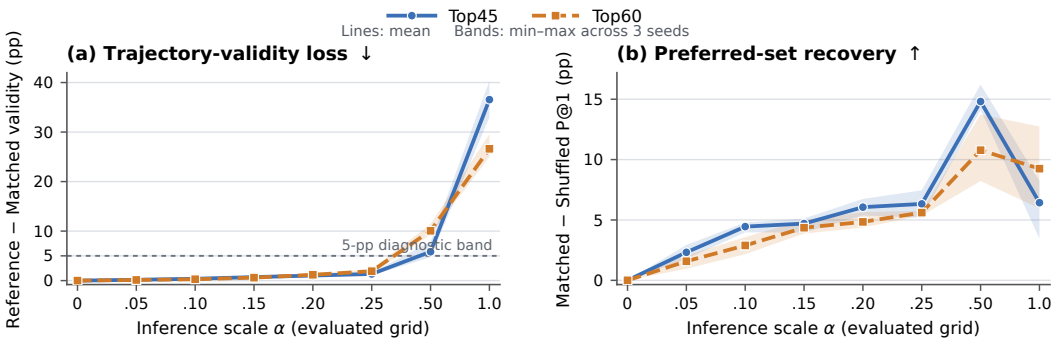


Figure 6: **Fixed-checkpoint scale response across three Stage-B seeds.** GraphDiT-8M Top45 and Top60: (a) Reference-minus-Matched trajectory-validity loss; (b) Matched–Shuffled P@1 gain, both in percentage points. Lines are seed means and bands span the seedwise minimum and maximum, not confidence intervals. The dashed five-point line is a diagnostic validity-loss reference.

Weak guidance yields similar responses. Raising α from .25 to .50 costs Top60 more validity without a larger P@1 gain, indicating a narrower calibration margin at these checkpoints (Figure 6). Both supports lose substantial validity under strong guidance.

C.4 Complete three-seed analysis across supports and capacities

Protocol and scope. All ten capacity–support settings in Table 1 have Stage-B seeds 42, 43, and 44. We fix the base generator, Stage-A projector, training corpus, and paired test sampling schedule; each run retains its own archived validation-selected epoch and α under the pipeline's recorded rule, with no test-based reselection. Table 13 uses the same equal-axis, case-balanced aggregation as Table 1. Sample SDs describe conditional Stage-B training-and-selection variation; they are not confidence intervals or estimates of full-pipeline uncertainty. Validity loss is $L_{\text{val}} = V_R - V_M$, where R and M denote Reference and Matched; positive values indicate a guidance cost.

Table 13: All-support Stage-B replication of Table 1, using seeds 42–44. Cells show the mean above $\pm$ sample SD ($n = 3$, denominator $n - 1$), in percentage points. All alignment contrasts are higher-is-better; validity loss $L_{\text{val}} = V_R - V_M$ is lower-is-better. [F] marks 65M Top15, where all three runs use fallback selections that failed the original validation hard gate. Reference is fixed across Stage-B seeds.

	Matched–Shuffled		Matched–Reference			Validity
Support	ΔP@1	$-\Delta$N@10	ΔP@1	ΔP@10	$-\Delta$N@10	L_{val}
GraphDiT-8M						
USPTO-MR	0.14 $\pm$0.18	1.35 $\pm$0.92	0.40 $\pm$0.40	1.16 $\pm$0.93	1.74 $\pm$1.44	2.08 $\pm$1.59
Top15	4.54 $\pm$0.61	5.46 $\pm$1.78	4.30 $\pm$0.58	0.69 $\pm$0.36	6.41 $\pm$1.84	2.14 $\pm$0.81
Top30	13.16 $\pm$1.02	18.09 $\pm$2.01	10.70 $\pm$1.29	0.26 $\pm$1.16	25.81 $\pm$2.42	12.85 $\pm$2.53
Top45	8.96 $\pm$4.64	12.55 $\pm$5.70	7.30 $\pm$3.24	2.96 $\pm$0.95	14.82 $\pm$8.93	2.46 $\pm$1.99
Top60	5.12 $\pm$0.81	7.93 $\pm$2.01	3.74 $\pm$0.19	1.09 $\pm$0.28	9.10 $\pm$2.36	1.63 $\pm$0.31
GraphDiT-65M						
USPTO-MR	-0.07 $\pm$0.06	1.25 $\pm$0.40	0.43 $\pm$0.22	2.04 $\pm$0.78	0.25 $\pm$0.57	-0.02 $\pm$0.35
Top15[F]	14.05 $\pm$2.52	17.81 $\pm$3.91	13.13 $\pm$1.99	3.19 $\pm$0.73	23.32 $\pm$4.07	8.95 $\pm$0.85
Top30	17.50 $\pm$1.20	26.90 $\pm$1.05	15.57 $\pm$0.44	5.00 $\pm$0.41	33.49 $\pm$2.11	1.76 $\pm$0.82
Top45	17.47 $\pm$1.16	27.50 $\pm$1.43	15.53 $\pm$0.69	4.57 $\pm$0.69	34.78 $\pm$1.58	3.90 $\pm$0.20
Top60	0.35 $\pm$0.91	0.10 $\pm$0.60	-1.65 $\pm$0.67	-0.25 $\pm$0.53	0.86 $\pm$0.75	2.37 $\pm$1.67

Control across supports. At 8M, all synthetic supports improve P@1 against both Shuffled and Reference in all three seeds, but Top45's primary maximum varies with training and scale selection. At 65M, Top30/Top45 retain strong gains, whereas Top60's net P@1 effect is negative in every seed. Thus the repeats support reliable control at intermediate supports without fixing a universal best support. The 8M Top30 results also distinguish moving preferred routes to the front from improving their recovery anywhere in the Top-10. USPTO-MR retains near-zero P@1 instruction-content effects at both capacities.

Validity costs. Stable rank gains can still be costly: 8M Top30 loses substantially more validity than Top45. All three 65M Top15 runs use the archived fallback rule after failing the original validation hard gate, and lose more validity than Top30/Top45. The support comparisons therefore reflect different validity costs and selection rules, not a shared validity budget.

D CONTROLLED METHOD ALTERNATIVES

D.1 DIRECT STAGE-B: EFFECT OF STAGE-A-ALIGNED INITIALIZATION

We isolate the contribution of Stage-A-aligned initialization. RIGS (called Joint in the original logs) copies the validation-selected epoch-26 Stage-A projector into each Stage-B projector group; Direct Stage-B (Scratch-B in the logs) uses the same constructor and seed at epoch 0, before any Stage-A update. The arms otherwise share the base checkpoint, adapter topology, trainable scope, Stage-B data and objective, optimizer, update budget, checkpoint coordinates, cohorts, and sampling protocol. The separately reported exact cosine and scalar-head controls in Appendix E.1 instead test frozen-output selection, not Stage-B initialization.

Table 14 retains the matched validation coordinates needed to interpret the ablation. Differences are RIGS minus Direct Stage-B in percentage points. For 8M, both variants select $\alpha_{\text{eval}} = .50$ and share the P@1-selected epoch 12 coordinate. For 65M, we fix epoch 14 and compare both variants at the RIGS operating scale $\alpha = .50$; comparing their independently selected scales would confound initialization with deployment selection.

Table 14: Matched-validation summary for the Stage-A initialization ablation. Δ is RIGS minus Direct Stage-B in percentage points. Larger ΔP@k, smaller ΔN@k, and positive ΔValidity favor RIGS. The 65M row is a matched-scale validation comparison, not a test result.

Capacity	$(e, \alpha_{\text{eval}})$	ΔP@1	ΔP@10	ΔN@1	ΔN@10	ΔValidity
8M	$(12, .50)$	$+1.25$	$+1.48$	-2.19	-5.39	$+1.12$
65M	$(14, .50)$	$+1.17$	$+0.86$	-0.63	-3.28	-1.16

Avoided-route suppression is the most consistent benefit in the 8M validation sweep: N@10 improves at every checkpoint, whereas preferred-route recovery does not improve uniformly across ranks. The 65M matched-scale sweep also strengthens control, with a modest validity cost. These single-seed diagnostics motivate checking whether the suppression advantage persists on held-out data.

We additionally evaluate the validation-frozen epoch-12/$\alpha = .50$ RIGS and Direct Stage-B checkpoints on exactly the same frozen test cohort and sampling schedule. The cohort contains 1,185 cases and 2,560 signed groups, with 100 trajectories per condition. Here "Top60" denotes the frozen evaluation candidate upper bound; we train both models with the Top30 RDKit5 recipe. For each model, the guidance effect is Matched minus Shuffled, and the difference-in-differences (DiD) is Direct Stage-B effect minus RIGS effect.

Table 15: Frozen-test Stage-A initialization difference-in-differences for USPTO–Guide(RDKit5)–8M–Top30. RIGS and Direct Stage-B use the same epoch-12 checkpoint coordinate, $\alpha_{\text{eval}} = .50$, test cohort, and sampling schedule; M/S denote Matched/Shuffled. Absolute arms are case-balanced percentages; effects and DiD are percentage points. Intervals use 10,000 paired case-cluster bootstrap replicates. For P@k and validity, a negative Direct-minus-RIGS DiD favors RIGS; for N@k, a positive DiD favors RIGS because lower recovery is better.

Metric	RIGS M	RIGS S	Direct M	Direct S	RIGS M−S	Direct M−S	Direct−RIGS DiD [95% CI]
P@1	24.80	12.03	23.26	11.31	$+12.77$	$+11.96$	-0.82 $[-3.47, +1.89]$
P@3	37.09	21.56	34.70	21.27	$+15.53$	$+13.43$	-2.10 $[-4.90, +0.70]$
P@5	39.83	25.11	37.74	25.36	$+14.73$	$+12.38$	-2.35 $[-5.01, +0.37]$
P@10	42.60	29.37	41.39	28.73	$+13.24$	$+12.66$	-0.58 $[-3.25, +2.14]$
N@1	2.93	13.62	5.08	12.74	-10.69	-7.67	$+3.02$ **[$+0.89, +5.19$]**
N@3	5.46	22.42	9.76	22.49	-16.96	-12.73	$+4.23$ **[$+1.91, +6.62$]**
N@5	6.68	26.50	12.24	26.33	-19.82	-14.09	$+5.72$ **[$+3.39, +8.19$]**
N@10	9.00	30.21	15.61	30.00	-21.21	-14.39	$+6.82$ **[$+4.36, +9.42$]**
Validity	53.27	53.28	52.43	52.45	-0.01	-0.03	-0.02 $[-0.31, +0.30]$

The frozen 8M test supports Stage A chiefly as an initialization that helps the adapter reject instruction-incompatible routes. All four N@k DiD intervals favor Stage A, while preferred-route

and validity intervals cross zero (Table 15). The evidence does not establish a promotion gain. Intervals are per metric, without multiplicity correction or training-seed uncertainty; the 65M evidence remains validation-only.

D.2 Stage-B loss ablation

We evaluate the five Stage-B objective terms in a single-seed ablation for USPTO–Guide(RDKit5)–65M–Top30. A fresh complete $P/R/T/G/K$ arm and five leave-one-out arms share the same base checkpoint, the selected Stage-A A0 projector, zero-output adapter initialization, data order, optimizer, 14-epoch / 57,260-update budget, and fixed $\alpha_{\text{train}} = 1$. The complete objective is

$$\mathcal{L}_{B,\text{metric}} = 0.10\mathcal{L}_{\text{pos}} + 0.05\mathcal{L}_{\text{rank}} + 0.05\mathcal{L}_{\text{teach}} + 0.05\mathcal{L}_{\text{guard}} + 1.00\mathcal{L}_{\text{roll}}. \tag{9}$$

The arm labels $P/R/T/G/K$ abbreviate positive DFM, route ranking, metric-teacher KL, full-reference KL guard, and rollout-reference KL, respectively; they are not new loss symbols. Each leave-one-out arm changes exactly one coefficient to zero.

We evaluate only on a frozen validation cohort of 1,162 case clusters and 2,560 signed groups (256 per axis/polarity cell), at epoch 14 and $\alpha_{\text{eval}} = .50$. Each Matched or Shuffled condition contains 100 trajectories per group; we reuse the protocol-audited Reference surface. For ablation variant v, we define $E_v^P = (\text{P@1})_{v,\text{Matched}} - (\text{P@1})_{v,\text{Shuffled}}$ and $E_v^N = (\text{N@1})_{v,\text{Shuffled}} - (\text{N@1})_{v,\text{Matched}}$, so larger values mean stronger instruction-specific promotion or suppression, respectively. Figure 7 reports $\Delta E^P = E^P_{\text{full}} - E^P_v$, $\Delta E^N = E^N_{\text{full}} - E^N_v$, and $\Delta V = V_{\text{full,Matched}} - V_{v,\text{Matched}}$ in percentage points. Positive ΔV therefore means that deleting the term reduces validity. The full arm has $E^P = 14.89$, $E^N = 14.40$, and Matched validity of 68.97% (Reference: 70.22%), corresponding to a 1.25-point Reference–Matched validity loss. The reported p_{H} adjusts the five deletion comparisons by Holm's method separately for each endpoint. The supplementary document reports the full confidence intervals and adjusted p-values under "Stage-B loss ablation: full statistical results."

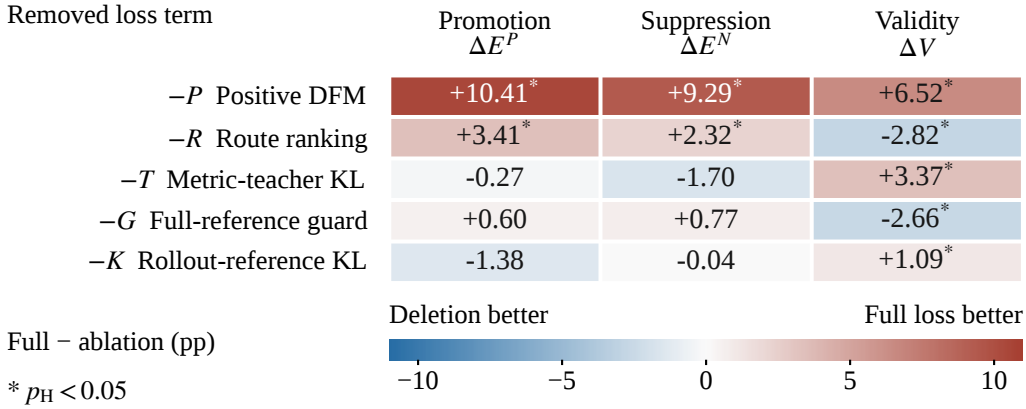


Figure 7: **Stage-B loss ablation.** Full-minus-ablation effects (pp) for USPTO–Guide(RDKit5)–65M–Top30 at epoch 14 and $\alpha_{\text{eval}} = .50$. Red favors the full loss; blue favors deletion. Asterisks mark $p_{\text{H}} < .05$, Holm-adjusted over the five deletions separately for each endpoint. Complete 95% paired-bootstrap intervals and adjusted p-values appear in the supplementary statistical table. This is a single-seed, validation-only comparison.

Positive DFM and route ranking provide the clearest control signal: removing either weakens promotion and suppression, with paired intervals excluding zero. Positive DFM also supports validity, whereas ranking trades validity for control. Metric-teacher and rollout-reference KL support validity without a resolved control gain. Deleting the guard improves validity with unresolved control changes, so this single-seed validation ablation does not establish its intended protective role.

E PROPOSAL AND SELECTION COUNTERFACTUALS

E.1 BUDGETED PROPOSAL–SELECTION COUNTERFACTUAL

The USPTO Top30 RDKit5 diagnostic uses the 8M backbone on 1,182 test cases and 2,560 signed groups. Reference generates without instruction guidance; Matched uses the correct instruction at Stage-B epoch 12 and $\alpha = .50$. Each arm contributes the literal first $B \in \{1, 5, 10, 25, 50, 100\}$ trajectories without invalid or duplicate backfilling. The following selectors reorder the valid distinct candidates in each realized pool, leaving Hit@B and Unique valid unchanged.

Selectors. **Frequency** ranks distinct candidates by descending occurrence count across the B raw draws.

Scalar head (s42) applies a width-256 MLP (GELU, dropout 0.1) to normalized instruction and route embeddings and their elementwise product, absolute difference, and cosine. The score also includes log frequency with a learned nonnegative weight. Here s42 denotes training seed 42.

Exact Stage-A cosine ranks candidates by the frozen Stage-A cosine similarity between projected instruction and route embeddings, without additional learned calibration or a frequency term.

RDKit oracle directly orders candidates by the instructed RDKit descriptor: larger values for "more" and smaller values for "fewer." It attains the best descriptor utility in the pool and hence zero RDKit regret; this need not maximize recovery of candidates in the frozen preferred set.

Readout fitting. The scalar head learns the requested RDKit preferences from Reference-generated training pools, using frozen instruction and route embeddings. We select it by P@1 on a separate validation set, then apply the same head across proposal pools. Table 17 reports sensitivity to the training seed. Scalar-head and oracle readouts are evaluation diagnostics; primary inference uses frequency ordering.

Selector metrics. Here Hit@B means preferred-set $\mathrm{Hit}_+@B$. P@1/Hit is $100\,\mathrm{P@1}/\mathrm{Hit}_+@B$, the percentage of groups selecting a preferred candidate first among those whose pool contains one. For signed raw RDKit utility $u_g(r) = s_g m_g(r)$, with polarity $s_g \in \{-1, +1\}$ and descriptor count m_g, regret is $\max_{r \in \mathcal{U}_g(B)} u_g(r) - u_g(r_{g,1})$: the gap between the best utility in the realized valid pool $\mathcal{U}_g(B)$ and the first selected candidate. Lower is better. We average regret over groups with nonempty pools and metric values for every candidate, using raw descriptor units rather than the standardized training utility.

Guidance improves which proposal reaches the front without a resolved increase in preferred-set availability. At $B = 100$, paired 95% intervals resolve higher P@1, lower N@1, and fewer unique-valid candidates under frequency ordering, while the Hit@100 interval includes zero. This pattern supports prioritization of preferred routes within the sampled pool.

Terminal selection exposes a different trade-off. All three scalar heads reduce N@1 and metric regret on both pools while lowering high-budget P@1 (Table 17); exact Stage-A cosine is weaker at selecting known positives. Even the RDKit oracle can miss those positives because descriptor optimality and frozen-set membership are different objectives. The trade-off's magnitude varies across head seeds.

F LANGUAGE AND CHEMISTRY EVIDENCE

F.1 FREE-FORM STRUCTURAL INSTRUCTIONS AND THE FROZEN LANGUAGE REPRESENTATION

Structure supplies case-specific retain/avoid instructions rather than a fixed descriptor registry or scalar metric teacher. Offline annotation combines product-local RDKit evidence with language-model judgments (Appendix I.2). The frozen test split contains 2,229 cases, 6,752 semantic groups, and 4,606 distinct instruction strings. Qwen3-Embedding-8B (Zhang et al., 2025) represents these strings as frozen token features; it does not generate their annotations (Appendix I.1). Sampling and aggregation follow Appendix H.9.

Table 16: Budgeted Top30 proposal and selector results. Panel (a) reports proposal support at representative budgets. Panel (b) reports all selectors used for the primary Reference and Matched pools at $B = 100$. Hit, P@1, N@1, P@1/Hit, and duplicate rate are percentages; Unique valid is a mean count, and RDKit regret uses raw descriptor units.

(a) Literal-prefix proposal support

Proposal	B	Hit@B	Unique valid	Duplicate rate
Reference	5	24.38	3.40	1.55
Reference	25	50.98	15.68	6.08
Reference	100	64.80	53.53	15.23
Matched	5	18.44	2.64	0.62
Matched	25	45.35	12.59	2.42
Matched	100	63.87	46.95	6.10

(b) $B = 100$ **proposal–selector combinations**

Proposal	Selector	Hit@100	Unique valid	P@1	N@1	P@1/Hit	RDKit regret
Reference	Frequency	64.80	53.53	24.57	25.39	37.91	1.332
Reference	Scalar head (s42)	64.80	53.53	22.23	3.32	34.30	0.510
Reference	Exact Stage-A cosine	64.80	53.53	5.35	12.89	8.26	1.627
Reference	RDKit oracle	64.80	53.53	26.99	0.35	41.65	0.000
Matched	Frequency	63.87	46.95	32.73	8.75	51.25	1.342
Matched	Scalar head (s42)	63.87	46.95	19.06	0.62	29.85	0.588
Matched	Exact Stage-A cosine	63.87	46.95	7.70	6.48	12.05	1.693
Matched	RDKit oracle	63.87	46.95	18.48	0.12	28.93	0.000

Table 17: Scalar-head seed sensitivity at $B = 100$ on the frozen test pools. Seed 42 is the registered primary head; seeds 43 and 44 are sensitivity runs. We select checkpoints independently on Reference validation and reuse them without adaptation across proposal sources. P@1 and N@1 are percentages.

Proposal	Head seed	Selected epoch	P@1	N@1	RDKit regret
Reference	42 (primary)	3	22.23	3.32	0.510
Reference	43	1	23.05	9.22	0.895
Reference	44	2	22.38	5.94	0.677
Matched (e12)	42 (primary)	3	19.06	0.62	0.588
Matched (e12)	43	1	21.37	1.13	0.814
Matched (e12)	44	2	19.88	0.86	0.689

We selected the following verbatim test instructions without consulting Matched–Shuffled outcomes. They span different constraint types:

- **Core retention and stereochemistry:** "Prefer precursors that retain the intact pyrrolidine core with the correct stereochemistry."
- **Bond connectivity:** "Prefer precursors that already contain the biaryl bond between the phenanthrene core and the pendant phenyl ring."
- **Substitution relation:** "Prefer precursors with a single halogen substituent ortho or para to the amino group."
- **Functional form and route preference:** "Prefer precursors that incorporate the tert-butyl carbamate protected amine directly rather than requiring azide or isocyanate intermediates."
- **Ring-system exclusion:** "Avoid precursors that introduce additional fused or spiro ring systems beyond the target cyclopropane."
- **Fragment/source integration:** "Avoid precursors that introduce the bromine atom as a salt or separate fragment."

Each instruction combines a case-specific molecular entity with a relation, polarity, or route-form preference. The learned projector transfers its frozen-language features to route scoring and node-to-text memory (Eqs. 12 and 15). Matched and Shuffled share the checkpoint and sampling schedule, changing only which instruction enters this pathway.

Table 18: Native-cohort Structure absolute-arm results (%) on 2,229 cases and 6,752 groups. Values average groups; paired case-balanced effects appear in Table 19.

(a) Preferred-route recovery

Arm	P@1	P@3	P@5	P@10
USPTO–Guide(Struct)–8M–Top30				
e22, $\alpha = .50$				
Reference	30.554	49.674	57.079	64.114
Matched	35.382	53.525	58.960	63.181
Shuffled	32.065	49.348	54.902	59.775
USPTO–Guide(Struct)–65M–Top30				
e26, $\alpha = 1$				
Reference	31.117	51.466	59.064	66.558
Matched	45.053	62.115	66.780	70.172
Shuffled	37.456	55.213	60.930	65.744

(b) Disfavored-route recovery and validity

Arm	N@1	N@3	N@5	N@10	Validity
USPTO–Guide(Struct)–8M–Top30					
e22, $\alpha = .50$					
Reference	19.535	36.893	44.283	51.555	69.982
Matched	12.678	26.052	33.797	41.410	60.567
Shuffled	16.321	31.398	38.315	44.787	60.500
USPTO–Guide(Struct)–65M–Top30					
e26, $\alpha = 1$					
Reference	24.097	42.210	49.985	58.175	71.231
Matched	6.680	14.662	18.824	23.978	64.555
Shuffled	13.907	27.636	34.864	41.484	64.706

Table 19: Case-specific structural control: case-balanced Matched–Shuffled effects in percentage points with 95% case-cluster intervals. Positive ΔP and $-\Delta$N indicate promotion and suppression, respectively.

Metric	8M, e22, $\alpha = .50$	65M, e26, $\alpha = 1$
ΔP@1	+3.554 [2.491, 4.598]	+8.089 [6.910, 9.243]
ΔP@10	+3.769 [2.939, 4.573]	+5.003 [4.155, 5.853]
$-\Delta$N@1	+3.908 [3.082, 4.755]	+7.799 [6.878, 8.753]
$-\Delta$N@10	+3.434 [2.525, 4.341]	+18.444 [17.245, 19.651]
ΔValidity	+0.051 [-0.086, +0.190]	-0.035 [-0.278, +0.212]

Both capacities promote preferred routes and suppress avoided routes at Top-1 and Top-10, with intervals excluding zero in the requested direction. These effects support case-specific structural control beyond the fixed metric registry. Using the same case-balanced aggregation, the Matched–Reference P@1 gains and N@10 reductions reported in Table 4 are 5.54 and 10.57 percentage points at 8M, and 14.74 and 35.79 at 65M, respectively. These point estimates average within-case contrasts; subtracting the group-averaged absolute values in Table 18 gives a different weighting.

F.2 A CHEMISTRY-AUDITED PAIRED CASE: FEWER PRECURSOR HALOGENS

Figure 8 illustrates how a compositional instruction can change the chosen disconnection for a fixed product. The manually audited USPTO Top30 case uses GraphDiT-65M at epoch 14 and the instruction "Choose precursor sets with fewer halogen atoms" for 3-phenoxytoluene, `Cc1cccc(Oc2ccccc2)c1`. Guidance shifts draws toward the preferred set; the caption reports the validity cost.

We rank non-invalid unique outputs by empirical frequency (ties use first trajectory index) and show five examples per arm that pass manual checks for atom-source consistency, meta regiochemistry, an identifiable reaction class, and primary-literature precedent. These are curated chemistry witnesses, not the unfiltered model Top-5; curation does not alter any quantitative count. The two Reference bromoarene routes have reaction-class precedent for ultrasound-assisted Ullmann coupling (Smith & Jones, 1992). The remaining three Reference routes have exact-product precedent (Ling et al., 2012; Chandnani & Chandalia, 1999; Tan et al., 2014). The Guided examples comprise two close-substrate Chan–Lam precedents and three reaction-class-only oxidative-dehydrogenation precedents (Chan et al., 1998; Evans et al., 1998; Iosub & Stahl, 2015).

We fix RC-predictor rank 2, the root choice, and trajectory seed 209236. The bottom row then changes from a one-halogen Ullmann disconnection with reaction-class precedent to a zero-halogen, close-substrate Chan–Lam disconnection. It is a qualitative witness of the requested compositional shift; we evaluate aggregate behavior separately.

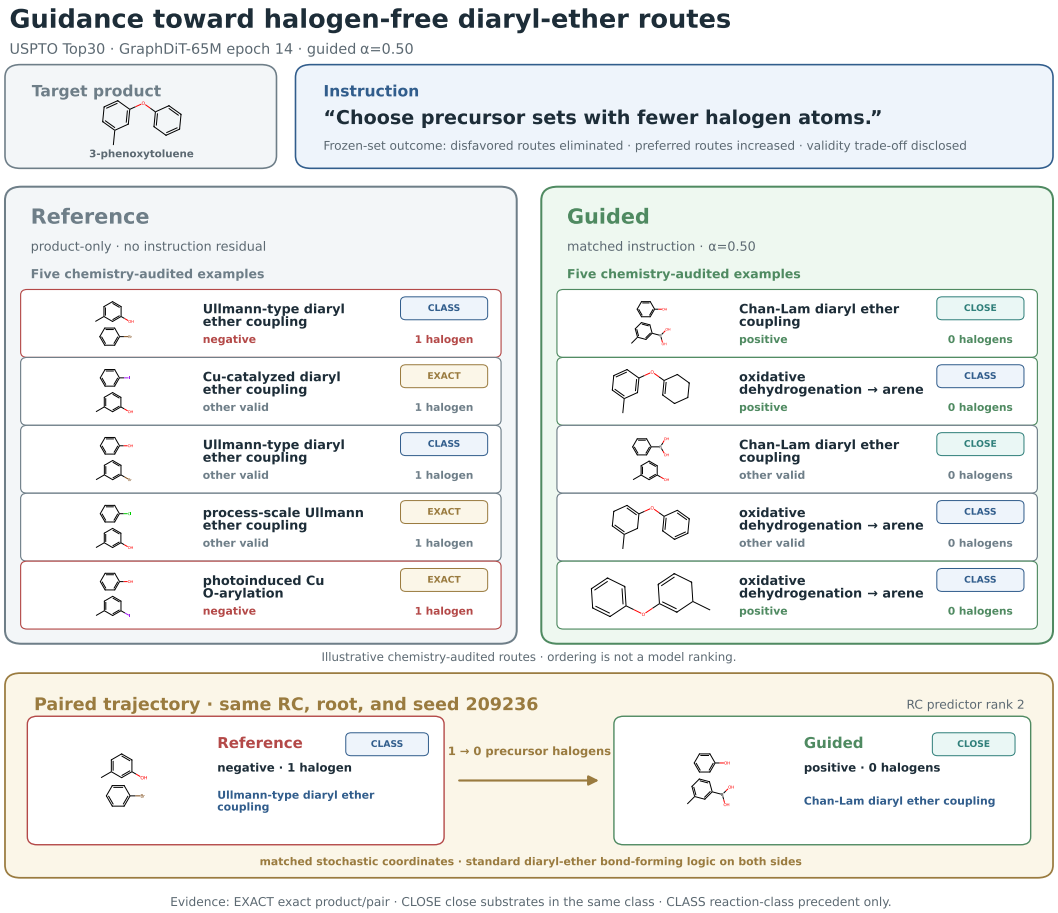


Figure 8: Chemistry-audited qualitative witness for “Choose precursor sets with fewer halogen atoms.” Each column shows five manually reviewed examples selected from frequency-ranked valid unique outputs, not the unfiltered model Top-5. Evidence badges denote exact-product experiments (Exact), close-substrate experiments (Close), and reaction-class-only precedent (Class). The bottom row pairs outputs at fixed RC schedule and seed. Positive/negative labels denote frozen-set membership, independently of the evidence badges. Across all 100 draws, Reference has 21 negative and 19 positive hits at 91% validity; Guided has 0 and 24 at 85% validity.

F.3 Downstream multistep planning witnesses

Figure 9 illustrates how local instruction following can propagate through a multistep search: changing precursor choices redirects later expansions and the resulting terminal precursors. The two paired witnesses establish this possibility; they do not estimate average planning gains, cross-seed stability, or experimental feasibility.

Search method. We use a custom value-guided AND–OR tree-search planner inspired by Retro* (Chen et al., 2020). Molecule nodes represent alternative reactions (OR), and reaction nodes require all their precursors (AND). A frozen value model estimates the remaining search effort for unexpanded molecules. Reaction costs sum precursor costs, and molecule costs take the minimum over candidate reactions. The planner selects an unresolved molecule in the current lowest-cost partial plan and expands it with RIGS, then updates these costs back toward the target. Guided searches apply the same instruction at every expansion. Search ends when all leaves of a complete route

belong to the building-block inventory, no expandable branch remains, or the search reaches 500 expansions per target.

Route-level measures. Route-level reaction-class ambiguity is $\max_j(1 - p_j)$, where p_j is the ASKCOS classifier's top-class confidence for reaction j (Tu et al., 2025); lower values indicate more confident classification. Proof depth is the longest chain of reaction steps. Selected reactions and terminal precursors count reaction nodes and leaf precursors, respectively; terminal halogen count sums the halogen atoms over those leaves. A solved proof reaches the planner's terminal-precursor criterion at every leaf.

Ambiguity-low selected proof. For target `pistachio100:038`, `CC(C)[C@H](C(=O)OCc1ccccc1)N1C(=O)N[C@@H](COc2ccc(Br)cc2)C1=O`, guidance reduces route-level reaction-class ambiguity while changing every selected reaction and terminal precursor, despite equal proof depth. This is a screen-selected computational witness: automatic RDKit parsing and route-continuity checks pass, but we have not manually audited its chemistry.

Halogen-low planning case. For target index 20, `COCO[C@@H](/C=C/C=O)CC[C@@H](C)O[Si](C)(C)C(C)(C)C`, we compare Reference and guided searches using RIGS-GraphDiT-65M. The guided arm uses "Choose precursor sets with fewer halogen atoms" with $\alpha = 0.5$ and polarity -1. Guidance changes the root and every later selected reaction, yielding halogen-free terminal precursors (Figure 9B). The local preference thus affects the whole selected plan, including its depth.

F.4 Direct empirical sampled-mass reallocation

Top-k recovery measures whether a route set appears in the returned list; empirical mass measures how often the sampler visits it. We reclassify the frozen Common Top100 trajectories into five mutually exclusive outcomes: preferred, disfavored, neutral Top100, valid outside Top100, or invalid. For semantic group g, let $\widehat{r}_g^{(b)}$ be the canonical precursor set decoded from draw b (or an invalid sentinel), and let $\mathcal{R}_g^+$ and $\mathcal{R}_g^-$ be the preferred and avoided precursor sets in its frozen evaluation inventory. The preferred empirical mass in the first B draws is

$$M_+(g;B) = \frac{1}{B}\sum_{b=1}^{B} \mathbf{1}\left[\widehat{r}_g^{(b)} \in \mathcal{R}_g^+\right], \tag{10}$$

with $M_-(g;B)$ defined using $\mathcal{R}_g^-$. These sets contain the canonical routes indexed by $\mathcal{P}_c$ and $\mathcal{N}_c$, respectively, in the evaluation group. Repeated occurrences count repeatedly; we neither deduplicate outputs nor backfill invalid draws. These quantities are literal frequencies under the frozen 100-draw protocol, not exact model probabilities. We report results at $B = 100$, using the equal-axis, case-balanced aggregation and paired bootstrap in Appendix H.9. The signed mass gap is $M_+ - M_-$.

Matched text reallocates repeated draws toward preferred routes and away from avoided routes at all ten coordinates, with paired intervals excluding zero in both directions (Table 20). Validity contrasts remain unresolved, and invalid draws stay in the denominator. The effect therefore extends beyond changing which route crosses a rank cutoff.

G External and heterogeneous robustness

G.1 Pistachio Top30: control on recorded reaction alternatives

We report a 65M Guide(RDKit5) model trained on Pistachio Top30. We select the evaluation cohort from the native Top30 RDKit5 test groups and retain their preference labels. It contains 1,035 products and 2,560 signed groups balanced across the five RDKit axes and both polarities. All three arms use the same fixed cohort and labels. We check the cohort's products and eligible precursor sets against the Pistachio Top100 test inventory of 3,516 products and 17,556 recorded candidate routes. In this data construction, we add template-generated candidates only to training and validation. The evaluation-inventory count differs from the 17,549 accepted graph routes in Table 25, which reports the model-bound graph view.

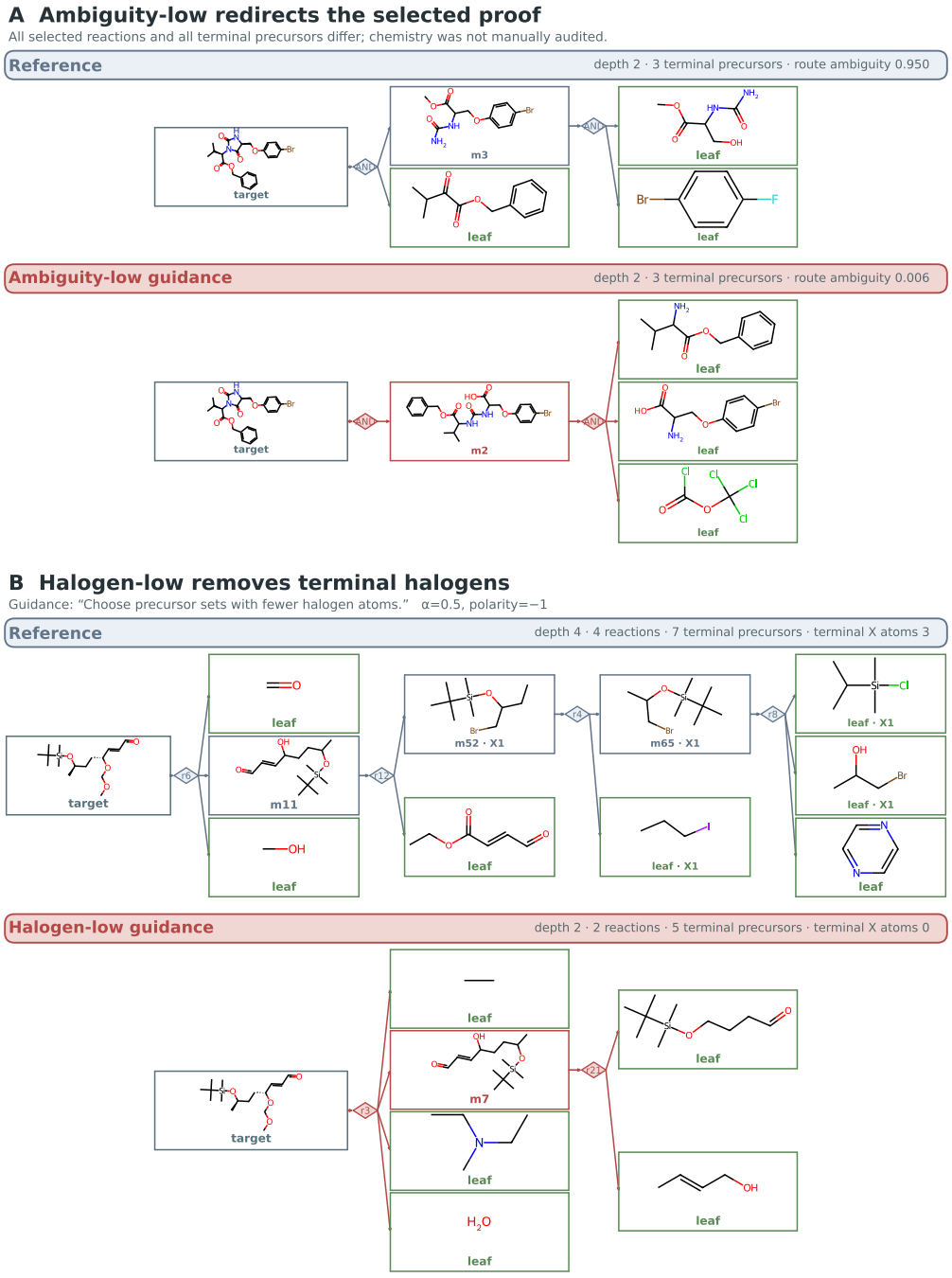


Figure 9: Selected proofs for two qualitative planning witnesses. (A) Ambiguity-low redirects the route at unchanged depth; chemical feasibility remains unverified. (B) Halogen-low redirects the selected plan; both searches find a complete plan, and both selected plans pass the structural audit. Cards are molecule OR nodes, diamonds are reaction AND nodes, and Xn marks n halogen atoms.

Reference, Matched, and different-axis Shuffled share the reaction-center top-5 proposal schedule and sampling seeds, using 100 raw draws per group and 50 steps. Root and trajectory seeds are 20260721 and 42000. Intervals use 10,000 paired case-cluster bootstrap replicates with equal axis weights (seed 1042). Table 21 gives the recorded checkpoint epoch and residual scale alongside all three arms.

Table 20: Direct empirical sampled-mass reallocation at $B = 100$. Each cell is the equal-axis, case-balanced Matched–Shuffled effect in percentage points, followed by its 95% paired-bootstrap interval. We reverse the sign of disfavored mass, so positive $-\Delta M_-$ is favorable. The ten rows are the selected Common Top100 checkpoints registered in Table 5.

Capacity	Support	ΔM_+	$-\Delta M_-$	$\Delta(M_+ - M_-)$	ΔValidity
8M	USPTO-MR	$+0.30$ $[+0.21, +0.39]$	$+0.38$ $[+0.30, +0.46]$	$+0.67$ $[+0.53, +0.82]$	-0.10 $[-0.33, +0.13]$
8M	Top15	$+0.94$ $[+0.82, +1.05]$	$+1.04$ $[+0.91, +1.17]$	$+1.98$ $[+1.78, +2.17]$	-0.06 $[-0.21, +0.09]$
8M	Top30	$+2.27$ $[+2.10, +2.44]$	$+2.26$ $[+2.09, +2.43]$	$+4.53$ $[+4.24, +4.82]$	-0.11 $[-0.43, +0.22]$
8M	Top45	$+2.67$ $[+2.48, +2.85]$	$+2.69$ $[+2.50, +2.89]$	$+5.36$ $[+5.04, +5.68]$	$+0.15$ $[-0.18, +0.48]$
8M	Top60	$+1.26$ $[+1.15, +1.37]$	$+1.20$ $[+1.09, +1.32]$	$+2.46$ $[+2.28, +2.64]$	0.00 $[-0.20, +0.20]$
65M	USPTO-MR	$+0.12$ $[+0.04, +0.19]$	$+0.16$ $[+0.10, +0.22]$	$+0.27$ $[+0.17, +0.38]$	$+0.06$ $[-0.13, +0.24]$
65M	Top15	$+3.28$ $[+3.01, +3.55]$	$+3.26$ $[+3.00, +3.53]$	$+6.54$ $[+6.11, +6.98]$	$+0.14$ $[-0.25, +0.53]$
65M	Top30	$+5.18$ $[+4.88, +5.49]$	$+5.06$ $[+4.74, +5.38]$	$+10.24$ $[+9.71, +10.77]$	$+0.11$ $[-0.20, +0.44]$
65M	Top45	$+4.07$ $[+3.82, +4.32]$	$+4.02$ $[+3.77, +4.27]$	$+8.09$ $[+7.67, +8.53]$	$+0.09$ $[-0.30, +0.48]$
65M	Top60	$+0.26$ $[+0.18, +0.35]$	$+0.21$ $[+0.13, +0.28]$	$+0.47$ $[+0.35, +0.58]$	$+0.10$ $[-0.06, +0.27]$

Table 21: **Pistachio 65M Top30 five-axis results on recorded reaction alternatives.** Stage-B checkpoint epoch $e = 10$; residual scale $\alpha = 0.50$. All entries are percentages. Hit$_+$@100 measures preferred-set availability; we compute validity over raw draws. P@k/N@k use frequency-ranked unique valid outputs.

		Preferred recovery ↑				Disfavored recovery ↓				
Arm	Hit$_+$@100	P@1	P@3	P@5	P@10	N@1	N@3	N@5	N@10	Validity
Reference	35.59	3.48	7.89	11.52	18.52	3.83	7.77	10.51	18.63	73.82
Matched	35.63	4.45	10.47	14.80	21.29	0.66	1.99	3.20	5.04	69.39
Shuffled	23.91	2.93	5.90	7.97	12.62	2.42	6.13	8.83	12.23	69.26

Control also appears on recorded reaction alternatives: paired 95% intervals resolve preferred-route promotion and avoided-route suppression at Top-10 against Shuffled. Against Reference, P@1 and P@10 improve and N@10 falls, while the preferred-set Hit@100 interval includes zero. This supports a change in which available routes the model returns first, with a validity cost visible in Table 21.

G.2 Applicability to another metric suite

Metric4 tests the same two-stage framework with a different supervision suite: synthetic-complexity delta, procurement burden, halogen count, and ring count. We train separate models with Metric4 supervision, so the experiment assesses applicability under metric-specific training. All ten settings share a strict-Top100 inventory of 982 products and 2,048 signed groups. Each arm uses RC Top-5 proposals, 100 raw draws per group, and 50 sampling steps; Matched and Shuffled share sampling seeds. We select checkpoints and guidance scales on validation and keep them fixed for testing. Estimates average groups within each case and then average cases; 95% intervals use 10,000 paired case-cluster bootstrap resamples.

Table 22 retains all five supports at both capacities. Seven of the eight synthetic-support settings show preferred Top-1 promotion and avoided Top-10 suppression against Shuffled, and preferred

Top-1 gains over Reference, with intervals excluding zero. The P@1 instruction effect remains unresolved for USPTO-MR and 65M Top60. Validity costs relative to Reference remain visible in the last column; Matched–Shuffled validity intervals include zero at all ten settings. These results support applicability beyond RDKit5, with gains that depend on the support and deployment setting. For the Top30 settings in Table 4, the Matched–Reference reduction in N@10 is additionally 8.04 percentage points (95% CI: $[6.52, 9.60]$) at 8M and 27.52 points ($[25.15, 29.91]$) at 65M, using the same case-balanced aggregation and paired bootstrap.

Table 22: Instruction control under Metric4 supervision. Entries are percentage-point effects with 95% confidence intervals. Matched–Shuffled tests instruction identity; Matched–Reference measures gains and validity costs over unguided generation. Positive ΔP and $-\Delta$N are favorable; negative ΔValidity indicates a loss. (e, α) gives the validation-selected Stage-B epoch and guidance scale.

		Matched–Shuffled		Matched–Reference	
Support	(e, α)	ΔP@1	$-\Delta$N@10	ΔP@1	ΔValidity
GraphDiT-8M					
USPTO-MR	$(14, .50)$	$+0.76$ $[-0.05, +1.63]$	$+2.62$ $[+1.48, +3.77]$	$+1.35$ $[+0.56, +2.19]$	-1.46 $[-1.72, -1.20]$
Top15	$(12, .25)$	$+3.46$ $[+2.14, +4.84]$	$+7.03$ $[+5.50, +8.61]$	$+3.06$ $[+1.55, +4.58]$	-2.88 $[-3.10, -2.65]$
Top30	$(6, .25)$	$+5.37$ $[+3.69, +7.08]$	$+7.41$ $[+5.91, +8.96]$	$+4.07$ $[+2.34, +5.78]$	-2.76 $[-2.97, -2.55]$
Top45	$(14, .50)$	$+12.17$ $[+10.08, +14.21]$	$+19.96$ $[+17.64, +22.30]$	$+9.04$ $[+6.98, +11.05]$	-3.62 $[-3.90, -3.34]$
Top60	$(14, .25)$	$+6.19$ $[+4.35, +8.02]$	$+10.77$ $[+8.94, +12.65]$	$+4.86$ $[+3.06, +6.67]$	-1.70 $[-1.94, -1.47]$
GraphDiT-65M					
USPTO-MR	$(12, .50)$	$+0.23$ $[-0.59, +1.04]$	$+1.81$ $[+0.74, +2.90]$	$+0.82$ $[+0.08, +1.58]$	$+1.11$ $[+0.78, +1.46]$
Top15	$(12, .25)$	$+7.71$ $[+5.98, +9.47]$	$+11.35$ $[+9.60, +13.21]$	$+7.10$ $[+5.50, +8.76]$	-2.02 $[-2.23, -1.80]$
Top30	$(10, .50)$	$+15.78$ $[+13.70, +17.90]$	$+21.92$ $[+19.50, +24.34]$	$+13.44$ $[+11.35, +15.48]$	-0.02 $[-0.31, +0.28]$
Top45	$(14, .50)$	$+15.81$ $[+13.57, +18.05]$	$+23.85$ $[+21.16, +26.60]$	$+11.25$ $[+9.06, +13.47]$	-5.09 $[-5.46, -4.70]$
Top60	$(2, .05)$	-1.15 $[-2.55, +0.20]$	$+0.76$ $[-0.36, +1.88]$	-0.84 $[-2.27, +0.61]$	$+0.08$ $[-0.07, +0.23]$

H Data, checkpoint, and evaluation provenance

H.1 Identifiers, data units, and reporting scope

Model identifiers follow Dataset–Role–Scale–Support, where Base denotes a no-instruction generator and Guide(Struct), Guide(Metric4), or Guide(RDKit5) identifies the adapter's supervision suite. Scale denotes the 8M or 65M backbone and Support its training candidate view. We report checkpoint, epoch, residual scale, and sampling budget separately. Reference, Matched, and Shuffled denote evaluation conditions, while Null reuses Reference at $\alpha = 0$.

Tables in this section distinguish the units in Table 23. In compact cells, **P/R** means usable products (or cases) / graph-executable routes, and **G/R** means signed semantic groups / graph-bound route keys or rows. The tables report model-bound supervision and base-generator training views. Test G/R denotes the native execution-view size. We describe sampled rollout cohorts, including the shared Top100 evaluations, separately.

Table 23: Counting units used in the data tables.

Unit	Definition
Product/case	One split-local canonical target product and its candidate pool.
Graph route	One candidate precursor set that survives canonicalization, mapping, graph-capacity, and route-binding checks.
Semantic group	One case–instruction–polarity comparison over several routes.
Paired family	The two opposite-polarity semantic groups for one case and axis.
Candidate occurrence	A route appearing inside a semantic group; not a distinct base route.

H.2 Construction of USPTO-MR

We construct USPTO-MR from USPTO-Full (Lowe, 2017) to capture multiple observed single-step retrosynthetic alternatives for the same product. We remove atom-mapping numbers for molecular identity comparison, canonicalize product and precursor SMILES while preserving stereochemistry, and group reactions by their canonical products. We collapse duplicate precursor combinations within each product group irrespective of fragment ordering and retain products with at least two distinct observed alternatives. We screen reactions for SMILES validity, atom-mapping consistency, identity transformations, and graph-size constraints. Following canonical normalization and deduplication, USPTO-MR contains 32,584 unique products and 77,110 product–precursor pairs. To prevent product overlap across splits, we assign all reactions associated with a canonical product to the same split. For products appearing in multiple original splits, we prioritize test over validation over training, resulting in 19,022 training, 6,098 validation, and 7,464 test products. Each alternative corresponds to a precursor combination recorded in the source dataset.

H.3 Base-generator data matrix

All synthetic supports use the same library of 270,794 retrosynthesis templates extracted from USPTO-Full and released with DESP (Yu et al., 2024), shared across disjoint product splits. Table 24 summarizes USPTO-MR and its synthetic expansions after preprocessing. Table 25 reports the exact model-bound, graph-executable views rather than upstream candidate rows. For USPTO, the 8M and 65M backbones reuse the same rows at each support. USPTO Top15/30/45/60 are synthetic-only nested prefix views; USPTO-MR is a separate natural-data baseline. Pistachio Top30 expands train and validation, while its registered test split contains only recorded reactions; the reported Pistachio model uses the 65M backbone.

Table 24: **USPTO-MR and synthetic expansions.** Totals after preprocessing; full split counts in Table 25.

Support	Products	Precursor sets	Avg./ product
USPTO-MR	32,584	77,110	2.37
Top15	31,912	470,308	14.74
Top30	31,912	922,977	28.92
Top45	31,912	1,360,737	42.64
Top60	31,912	1,788,168	56.03

We assign products with evaluation precedence, keeping train, validation, and test disjoint. The USPTO-MR graph cache removes one duplicate precursor-set route from the 77,111-row source. The Top$K$ rows are therefore not raw template counts: they are the routes the corresponding base model can actually sample during optimization.

H.4 Synthetic precursor construction and ranking

Candidate generation builds on the template-based reaction validation in AOT* (Song et al., 2026). Our implementation uses product-side fingerprint screening, exact SMARTS matching, and cached

Table 25: Base-generator data sizes. Each cell is usable products / accepted graph routes (P/R). *Both* means 8M and 65M use identical data rows; the two backbones still have separate checkpoints. USPTO Top60 reports the frozen model-training view, not the larger later annotation inventory. Pistachio's accepted test graph view (17,549 routes) is separate from the recorded-candidate evaluation inventory (17,556 routes; Appendix G.1).

Dataset	Support	Scale(s)	Train P/R	Validation P/R	Test P/R	Total P/R
USPTO	USPTO-MR	Both	19,022 / 41,488	6,098 / 14,740	7,464 / 20,882	32,584 / 77,110
USPTO	Top15	Both	18,595 / 274,018	5,975 / 88,142	7,342 / 108,148	31,912 / 470,308
USPTO	Top30	Both	18,595 / 537,777	5,975 / 172,969	7,342 / 212,231	31,912 / 922,977
USPTO	Top45	Both	18,595 / 793,099	5,975 / 254,984	7,342 / 312,654	31,912 / 1,360,737
USPTO	Top60	Both	18,595 / 1,042,205	5,975 / 335,110	7,342 / 410,853	31,912 / 1,788,168
Pistachio	Top30	65M	36,931 / 1,189,822	391 / 13,142	3,516 / 17,549	40,838 / 1,220,513

RDChiral template execution. A fixed heuristic selector balances quality and precursor diversity; its scores are not calibrated reaction-success probabilities. The accompanying USPTO package supplies the constructed supports and their ordering, so reproduction starts from these fixed data.

Instantiating complete precursor sets. We apply each screened template to the *complete* product molecule with RDChiral (Coley et al., 2019). Its product-side SMARTS identifies the local reaction environment; atom mappings anchor the precursor-side bond edits and template-specified atoms (e.g., leaving groups) to that match. RDChiral carries over molecular context outside the matched pattern from the product, restores non-reacted bonds, and handles stereochemical consistency. Each valid outcome therefore contains complete precursor molecules, whose disconnected components form one precursor set. We remove atom-map labels, canonicalize each component as isomeric SMILES, sort the components while preserving multiplicity, and deduplicate identical sets across template matches, retaining their template provenance. We discard invalid outcomes and the unchanged-product singleton. The quality–diversity selector below then ranks these complete precursor sets to define the Top$K$ supports.

Quality components. For product x and precursor set r, the quality score $Q_x(r)$ combines four heuristic signals: *template support*, favoring candidates proposed by multiple distinct templates; *template specificity*, favoring more detailed product-side reaction patterns; *precursor count*, favoring two-component sets; and *largest-fragment reduction*, favoring a smaller largest precursor relative to the product. We take a weighted average with weights 0.35, 0.30, 0.15, and 0.20, respectively, renormalized over the available signals.

Diversity and greedy selection. For the selected set S, diversity is $D(r \mid S) = 1 - \max_{u \in S} \mathrm{Tanimoto}(F(r), F(u))$, with $D(r \mid \varnothing) = 1$. F is a 2,048-bit, radius-two Morgan fingerprint of the full disconnected precursor set. At each step, the selector maximizes

$$s_x(r \mid S) = 0.60 Q_x(r) + 0.40 D(r \mid S) \tag{11}$$

over the remaining eligible candidates, then adds the chosen candidate to S and updates the similarities. The selector breaks ties by higher Q_x, then higher D, then lexicographically smaller canonical precursor SMILES. The first retained generating-template identifier is the candidate's primary template. A candidate is eligible while fewer than three previously selected candidates have that primary template; candidates without a primary identifier remain eligible. If no remaining candidate satisfies this quota, the selector relaxes it for that step.

The selector stops after K candidates or when no candidates remain. The selection order defines $\mathrm{rank}_x(r)$ in Section 3.1; each stored `selection_score` is the value of Eq. 11 when the selector chose that candidate. For a fixed input pool and configuration, changing K only changes the stopping

point, yielding nested prefixes. The shared graph checks and deduplication in Filter then determine the usable Base training support, whose size can be below K.

H.5 Guidance suites and construction

The three guidance suites are different supervision sources and are not interchangeable aliases. Structure guidance is case-specific, whereas Metric4 and RDKit5 use fixed metric registries with opposite signed directions. Table 26 gives the construction method, software provenance, and two verbatim canonical examples from the materialized text pools. The accompanying data schema documents unannotated candidate records (item C2). The supplementary training example (item C3) illustrates the normalized Structure view: we retain Positive/Negative candidates, omit Neutral candidates, and renormalize annotation probabilities over the retained candidates. Structure annotation details appear in Appendix I.2.

Metric values. For Metric4, complexity reduction is the target SCScore minus the mean precursor SCScore (Coley et al., 2018). Procurement burden combines precursor prices with penalties for missing quotes and longer lead times, using ASKCOS buyables data (Tu et al., 2025).

Metric preferences and target weights. Within each product's pool, we normalize RDKit counts using z-scores and complexity reduction and procurement burden using robust z-scores. For normalized values $\widetilde{d}_i$ and instruction direction s_c ($+1$ for higher, -1 for lower), the upper and lower quartile tails of $s_c\widetilde{d}_i$ define preferred candidates $\mathcal{P}_c$ and avoided candidates $\mathcal{N}_c$, including boundary ties. We exclude middle or missing-value candidates and groups without a valid contrast. The Stage-A target is a softmax over the retained candidates:

$$q_i^c = \frac{\exp(s_c\widetilde{d}_i)}{\sum_{j\in\mathcal{P}_c\cup\mathcal{N}_c}\exp(s_c\widetilde{d}_j)}, \qquad i \in \mathcal{P}_c \cup \mathcal{N}_c.$$

Appendices H.6 and I.5 describe support filtering and the Stage-B metric teacher, respectively.

H.6 Guidance-data matrix

Table 27 combines the three supervision suites. USPTO 8M and 65M models share each support-specific semantic view; their frozen route-semantic embedding widths differ. We sample the Structure validation and test cohorts once within their registered splits and reuse them across capacities.

After support restriction and graph filtering, RDKit5 and Metric4 retain only guidance groups containing both preferred and avoided candidates. Group counts therefore need not increase monotonically with Top$K$.

For RDKit5 USPTO Top15 and Top45, *R* counts retained subset-specific route keys after relabeling; for Pistachio, it counts referenced unique or endpoint routes in the exact product-condition view. These are graph-bound training identities, not candidate occurrences multiplied by semantic groups. Metric4 counts come from the support-specific execution manifests, including the complete frozen Top60 view. Appendix G.2 summarizes all ten settings on the shared strict-Top100 cohort.

H.7 Frozen evaluation cohorts

Complete execution views and formal rollout cohorts are distinct. Structure uses 6,752 validation groups from 2,262 cases and 6,752 test groups from 2,229 cases, with a variable number of groups per structural axis. We select the checkpoint and residual scale on validation and keep them fixed for testing. Appendix H.9 specifies the sampling protocol. We describe the RDKit5 Common Top100 cohort below; Appendix G.2 describes the Metric4 cohort.

RDKit5 strict-Top100 common-support scaling cohort. The primary support-scaling experiment uses one shared USPTO Top100 test universe across all training supports. The strict-100 filter precedes family sampling; Table 28 distinguishes the raw candidate inventory from the routes that survive graph binding. Each of the five axes contributes 256 paired families and 512 signed groups. All selected cases have exactly 100 raw candidates, and the frozen RC Top-5 table resolves every

Table 26: Guidance types, construction methods, and representative canonical instructions. We compute metric values before writing instructions and do not ask the language model to estimate Metric4 or RDKit5 values.

Suite	Signal	Construction and software	Two canonical examples
Structure	Product-local structural motifs, connectivity, fragment count, and retain/avoid criteria.	RDKit extracts candidate evidence; a frozen LLM listwise annotator proposes at most five relative criteria and route preferences. Deterministic filtering forms prefer/avoid views; RXN-Mapper (Schwaller et al., 2021) and RDKit bind surviving routes to executable graphs.	*Prefer precursors that already contain the imidazole ring directly attached to the propyl chain.* *Avoid precursors containing heteroatoms other than bromine and oxygen.*
Metric4	`mosyn.synthetic_complexity_delta`, `mosyn.procurement_burden`, `rdkit.halogen_atom_count`, and `rdkit.ring_count`.	MOSYN joins deterministic route metrics; SCScore supplies molecular complexity and ASKCOS buyables data supplies procurement inputs. RDKit computes the two structural counts. Pool-normalized signed tails create the two polarities.	*Choose disconnections that deliver a larger reduction in synthetic complexity.* *Choose precursor sets with a lighter procurement burden.*
RDKit5	RDKit halogen-atom, carbonyl-group, total-ring, aromatic-ring, and heterocycle counts.	RDKit atom/ring APIs and fixed SMARTS compute each descriptor over the complete precursor set. Pool-normalized, tie-safe signed tails produce labels; deterministic rule-authored templates produce wording.	*Choose precursor sets with fewer halogen atoms.* *Choose precursor sets with more aromatic rings.*

Table 27: Guidance-data sizes for the reported models. Numeric cells are semantic groups / graph-bound route keys or rows (G/R). Each USPTO row applies to both 8M and 65M; the Pistachio row applies to 65M only.

Dataset	Support	Train G/R	Validation G/R	Test G/R
RDKit5				
USPTO	USPTO-MR	60,404 / 41,488	20,610 / 14,740	25,598 / 20,882
USPTO	Top15	82,912 / 261,905	26,644 / 84,436	32,370 / 103,112
USPTO	Top30	130,880 / 528,659	42,116 / 170,286	51,522 / 208,613
USPTO	Top45	99,548 / 768,621	31,970 / 247,495	39,110 / 303,070
USPTO	Top60	134,226 / 1,046,897	43,100 / 336,212	52,564 / 410,175
Pistachio	Top30	180,460 / 1,034,142	1,820 / 10,994	13,542 / 14,738
Structure				
USPTO	Top30	54,012 / 537,777	6,752 / 65,473	6,752 / 64,422
Metric4				
USPTO	USPTO-MR	90,854 / 41,488	29,918 / 14,740	36,964 / 20,882
USPTO	Top15	121,628 / 274,018	39,084 / 88,142	47,850 / 108,148
USPTO	Top30	126,094 / 537,777	40,550 / 172,969	49,738 / 212,231
USPTO	Top45	131,156 / 793,099	42,262 / 254,984	51,668 / 312,654
USPTO	Top60	142,810 / 1,065,712	45,784 / 341,862	55,956 / 417,568

selected case. Thus "strict-100" fixes the raw inventory size even when fewer routes survive graph binding. Generation accounting appears in Appendix A.

Table 28: Size accounting for the strict-Top100 common-support test cohort. We recompute every column on the named case universe: raw routes count the frozen candidate inventory, graph-resolved route keys count routes retained by the frozen graph cache, and paired families and signed groups count the corresponding five-axis semantic records.

Case universe / view	Cases	Raw route records	Graph-resolved route keys	Paired families	Signed groups
Up-to-100 source test universe	7,342	733,017	666,086	29,320	58,640
Five-axis graph-bound execution view	7,269	725,812	659,056	29,314	58,628
Exact-100 eligible universe	7,320	732,000	665,146	29,262	58,524
Strict-Top100 formal cohort	1,201	120,100	107,801	1,280	2,560

H.8 Base-generator checkpoint provenance

For the USPTO Structure experiments (Appendix F.1), we select the 8M and 65M Base checkpoints by validation deduplicated Top-3 accuracy, at steps 73,728 and 88,105, respectively. Checkpoint selection does not use guidance test outputs.

H.9 Sampling and statistical protocol

Sampling and returned lists. The primary and native-cohort three-arm evaluations use ordered RC Top-5 proposals, 100 raw trajectories per group and physical arm, and 50 flow steps. Matched and Shuffled share reaction-center proposals, root choices, and trajectory seeds. We canonicalize and deduplicate valid outputs, then rank them by frequency, with canonical precursor SMILES breaking ties. Invalid draws remain in the sampling budget. We do not backfill draws or pad short returned lists. Metric Null is the $\alpha = 0$ view of Reference and adds no physical generation. The separately named selector and chemistry diagnostics specify their own readouts.

Shuffled instructions. For RDKit5 and Metric4, Shuffled samples a different axis within the same suite and split, followed by an available polarity and text pool, each uniformly, and uses canonical wording. These mappings (seed 1042) remain fixed across model settings on the same cohort. Structure selects a donor deterministically from a different product and semantic axis within the same split. Neither policy requires matching polarity. Evaluation retains the original instruction's preferred and avoided sets.

Supplementary cohorts. Appendix H.7 specifies the Structure validation and test cohorts, which have a variable number of groups per structural axis. We sample them without replacement before inference and keep them fixed across model scale, checkpoint, and residual strength. Appendix G.2 specifies the Metric4 cohort and its case-balanced evaluation. The RDKit5 initialization, loss, and proposal–selection studies specify their auxiliary cohorts in Appendices D.1, D.2, and E.1.

Aggregation and uncertainty. RDKit5 Common Top100 rates, mean counts, and paired effects use equal-axis, case-balanced aggregation: we average signed groups within each product–axis pair, products within each axis, and then the five axes equally. Pooled recovery counts distinct product–candidate pairs. For Structure, absolute metrics average groups, whereas paired effects average differences between arms within each case and then average cases. Confidence intervals use 10,000 shared case-cluster bootstrap resamples; Structure uses bootstrap seed 42. Metric4 retains the case-balanced aggregation specified in Appendix G.2. Bootstrap uncertainty describes variation across cases, separately from Stage-B training-seed variation.

I ARCHITECTURE, OBJECTIVES, AND IMPLEMENTATION

This section specifies the frozen inputs, Stage-A scoring, and Stage-B adapter and loss definitions. For each product–instruction group g, $\mathcal{C}_g$ denotes the candidate pool after filtering and deduplication, retaining its construction order. Appendix I.7 lists the training and inference settings.

I.1 FROZEN STAGE-A REPRESENTATIONS

Encoders. Frozen Qwen3-Embedding-8B (Zhang et al., 2025) supplies 4,096-dimensional instruction-token features. Reaction-pair features come from pretrained reaction-pair encoders (Top30 or Top60; 256-D) or the frozen GraphDiT Base model trained on the corresponding support (Native, 256-D or 512-D). The Top30 reaction-pair encoder was pretrained with Structure supervision; the Top60 version uses a separate pretrained checkpoint. We obtain Native features by pooling the complete product–precursor pair at $t = 0$ without instructions. We L2-normalize route embeddings and keep the language and route encoders frozen during projector fitting. Table 29 maps experiments to representations.

Table 29: Frozen route encoders in the reported experiments. Native features use the Base model trained on the corresponding support.

Guided experiments	Route encoder	Dimension
USPTO 8M Top30: RDKit5, Metric4, and Structure	Pretrained reaction-pair encoder (Top30)	256
USPTO 8M Top15/Top45/Top60: RDKit5 and Metric4	Pretrained reaction-pair encoder (Top60)	256
USPTO-MR 8M: RDKit5 and Metric4	Native	256
USPTO 65M: all RDKit5/Metric4 supports and Top30 Structure	Native	512
Pistachio 65M Top30: RDKit5	Native	512

Comparison and compute scope. The pretrained reaction-pair encoders use descriptor statistics fitted on their respective Top30 or Top60 training routes; the 8M Top15/Top45 runs reuse the Top60 encoder and its statistics. Guided scaling therefore varies route representations as well as candidate support, while Reference coverage is independent of these embeddings. *Generator-free* describes projector fitting on cached features: native cache construction runs the frozen GraphDiT backbone, and the reported Stage-A budget excludes cache construction and historical semantic-teacher pretraining.

I.2 STRUCTURE ANNOTATION AND QUALITY CHECKS

Annotation. The USPTO Structure annotator uses the recorded API alias `deepseek-v4-pro` (temperature 0.2; we did not record an immutable model revision). Given target and candidate precursor SMILES with RDKit evidence, it proposes up to five relative structural *Prefer*/*Avoid* instructions and labels candidates positive, negative, neutral, or excluded. Instructions omit candidate IDs, SMILES, and numeric descriptor values. We request rewards of $+1, -1, 0, 0$, respectively. To form the listwise target, we apply a temperature-one softmax to returned rewards over non-excluded candidates, then renormalize over the positive and negative candidates that remain after filtering. We supply the saved annotation system prompt as supplementary item P1 (`prompts/uspto_structure_v2.txt`).

Quality checks. Automated checks validate candidate coverage and label–reward consistency, and reject malformed annotations and detected copying of candidate literals. Mapping and graph-construction checks remove inconsistent or unrepresentable candidates; surviving groups must retain both positive and negative candidates. These checks establish executable preference contrasts. Chemical label correctness and distinctness of instruction axes remain LLM judgments without exhaustive independent expert adjudication.

I.3 Stage-A grounding and exact route scoring

Let $H_c = E_T(c)$ denote the instruction-token features, $Z_c = P_\phi(H_c)$ their projected tokens, and $v_i = E_R(x, r_i)$ the normalized reaction-pair embedding. Stage A averages the projected token features and normalizes the result to obtain the instruction representation z_c:

$$\begin{aligned} z_c &= \mathrm{Norm}(\mathrm{MaskedMean}(Z_c)), \\ \widehat{q}^{c}_{\phi,i} &= \frac{b_{g,i} \exp(\tau z_c^\top v_i)}{\sum_j b_{g,j} \exp(\tau z_c^\top v_j)}. \end{aligned} \tag{12}$$

Here τ scales the similarity logits, MaskedMean averages over non-padding instruction tokens, and Norm denotes L2 normalization. The instruction-independent candidate weights come from Structure's heuristic selection scores (Appendix H.4) or RDKit5's recorded route counts (one if absent). We normalize these weights over retained candidates, using uniform weights when all are zero, then floor each normalized value at 10^{-8} to obtain $b_{g,i}$.

The projector output and route embeddings have width 256 at 8M and 512 at 65M. Using the embeddings and listwise distribution in Eq. (12), the exact pairwise term compares raw similarity logits rather than prior-adjusted logits:

$$\begin{aligned} u_i(c) &= \tau z_c^\top v_i, \\ \mathcal{L}_{\mathrm{pair}} &= \frac{1}{|\mathcal{P}_c||\mathcal{N}_c|} \sum_{i \in \mathcal{P}_c} \sum_{j \in \mathcal{N}_c} [m_{\mathrm{pair}} - u_i(c) + u_j(c)]_+. \end{aligned} \tag{13}$$

The denominator averages all preferred–avoided pairs within a group. Transfer copies only the validation-selected projector parameters and resets optimizer state and training counters; the encoders remain frozen.

I.4 Stage-B residual adapter path

Let $\mathcal{I}$ index the adapted layers and $\mathcal{G}$ index the projector-sharing groups. The map $\gamma : \mathcal{I} \to \mathcal{G}$ assigns each adapted layer to a group. Each group $\kappa \in \mathcal{G}$ has a projector $P_{\kappa,\theta}$ initialized from $P_{\phi^\star}$. We write the group's instruction-token memory and the layer-wise memory from Section 3.3 as

$$M_{c,\kappa} = P_{\kappa,\theta}(H_c), \qquad M_c^{(\ell)} = M_{c,\gamma(\ell)}. \tag{14}$$

During inference, we cache the collection $M_c = (M_{c,\kappa})_{\kappa \in \mathcal{G}}$. Passing M_c in place of c evaluates the same conditional model with precomputed instruction features. The full masked adapter update is

$$\begin{aligned} \Delta h_\ell(c) &= A_{\ell,\theta}(h_\ell, M_{c,\gamma(\ell)}), \\ A_{\ell,\theta}(h, M) &:= m_G \odot \mathrm{Attn}_{\ell,\theta}(\mathrm{LN}(h), M, M), \\ \widetilde{h}_\ell &= h_\ell + \alpha \Delta h_\ell(c), \qquad h_{\ell+1} = B_\ell(\widetilde{h}_\ell). \end{aligned} \tag{15}$$

Here $\mathrm{Attn}_{\ell,\theta}$ is multi-head node-to-text attention and m_G masks generative graph positions. We omit the frozen block's edge-state, product, and time arguments from the notation. Using one-based layer indices, the 8M model shares one projector between layers $\{2, 4\}$ and another between $\{6, 8\}$. The 65M model additionally uses groups $\{10, 12\}$ and $\{14, 16\}$. Each listed layer has its own cross-attention adapter. Figure 10 shows the components and their pre-block assembly.

I.5 Stage-B loss components and metric teacher

For candidate i in group g, let $D^{\mathrm{student}}_{g,i}$ and $D^{\mathrm{ref}}_{g,i}$ be the guided and reference denoising losses on the same noisy precursor graph. Define $d_{g,i} = D^{\mathrm{ref}}_{g,i} - D^{\mathrm{student}}_{g,i}$; positive $d_{g,i}$ indicates improved denoising under guidance. The four terms in Eq. (4) are defined below.

Positive DFM ($\mathcal{L}_{\mathrm{pos}}$). $D^{\mathrm{student}}_{g,i}$ is the full categorical flow-matching loss over all valid node and edge positions. The positive term keeps only preferred candidates $i \in \mathcal{P}_c$, renormalizes the cached teacher probabilities within that subset, and averages their full losses across eligible groups. It is an absolute likelihood anchor: ranking alone could satisfy a relative comparison by damaging avoided routes without improving preferred ones.

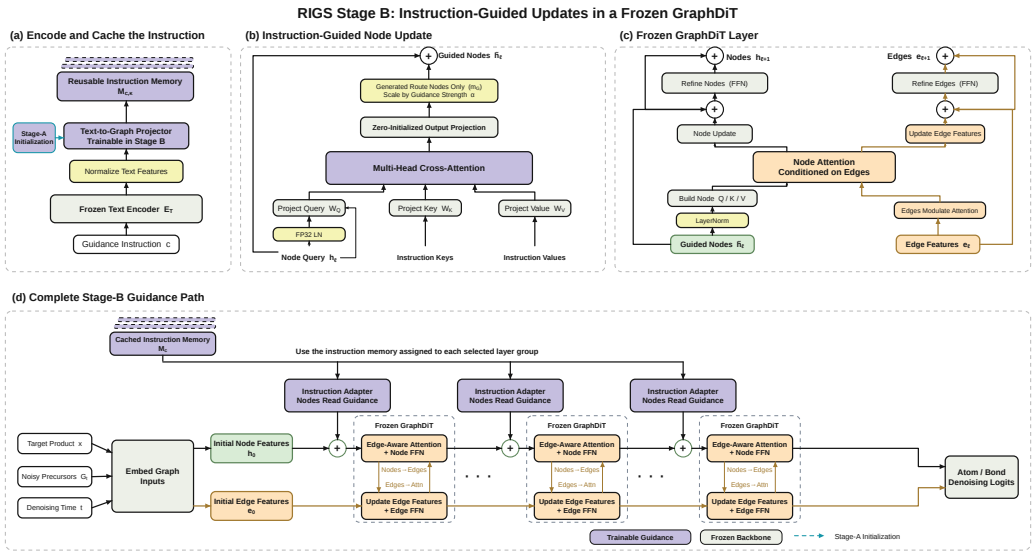


Figure 10: **Stage-B components and assembly.** (a) A frozen language encoder and a trainable Stage-A-initialized projector construct reusable instruction-token memory. (b) Node queries attend to that memory through a trainable adapter whose output projection starts at zero; masking, residual scaling, and the node-state skip are explicit. (c) The frozen GraphDiT block propagates the guided node update through its existing node–edge coupling. (d) We reuse the same components at selected pre-block injection sites. The adapters inject guidance directly only into node states; edge states respond inside the frozen backbone.

Raw route ranking ($\mathcal{L}_{\text{rank}}$). Within each eligible group, we compare every preferred–avoided pair (i, j) using the uncalibrated gains $d_{g,i}$ and $d_{g,j}$. We apply a one-sided Huber penalty to the margin violation $[m_{\text{rank}} - (d_{g,i} - d_{g,j})]_+$, then average the penalty over pairs and groups without rescaling scores within products.

Rollout Reference KL ($\mathcal{L}_{\text{roll}}$). During training, we periodically replay noisy precursor graphs visited by the guided sampler through both Student and frozen Reference. The loss averages $\mathrm{KL}(p_{\text{ref}} \| p_{\text{student}})$ over every valid node and edge category at those replay states. Unlike a teacher-forced penalty, it controls drift on states that the guided model actually reaches.

Full-reference KL guard ($\mathcal{L}_{\text{guard}}$). On shared noisy precursor graphs used for training, the full-state KL includes every valid node and edge position, including within-group disagreement positions. We average candidate KLs within each group, using endpoint probabilities as weights, to obtain K_g. Writing $\langle \cdot \rangle_g$ for the normalized, group-weighted training average, the unthresholded diagnostic and the guard objective are

$$\mathcal{L}_{\text{fullKL}} = \langle K_g \rangle_g, \qquad \mathcal{L}_{\text{guard}} = \big\langle [K_g - \delta_{\text{KL}}]_+ \big\rangle_g. \tag{16}$$

We apply the threshold before averaging: groups below the tolerance contribute zero. The guard constrains drift on noisy precursor graphs from training candidates; rollout KL covers those reached by the sampler.

The first two terms implement instruction steering and the latter two preserve the base generator. Structure guidance uses this core directly. For metric axis a, let $x_{g,i}^{(a)}$ be the deterministic route metric and let $s_g^{(a)} = +1$ request larger values and $s_g^{(a)} = -1$ request smaller values. Pool z-scoring subtracts the group mean and divides by its population standard deviation; constant-valued groups are ineligible. The deterministic teacher and Student improvement distribution are

$$\begin{aligned} z_{g,i}^{(a)} &= \mathrm{PoolZScore}(\{x_{g,j}^{(a)}\}_j)_i, \quad u_{g,i}^{(a)} = s_g^{(a)} z_{g,i}^{(a)}, \\ q_{g,i}^{(a)} &= \mathrm{softmax}_i(u_{g,i}^{(a)}/T_q), \qquad p_{g,i} = \mathrm{softmax}_i(d_{g,i}/T_s). \end{aligned} \tag{17}$$

The metric branch optimizes

$$\mathcal{L}_{\text{teach}} = \mathrm{KL}(q_g^{(a)} \| p_g), \qquad \mathcal{L}_{B,\text{metric}} = \mathcal{L}_{B,\text{core}} + \lambda_{\text{teach}} \mathcal{L}_{\text{teach}}. \tag{18}$$

T_q is the teacher temperature and T_s is the Student-score temperature; larger values flatten the corresponding distribution. The teacher is an offline deterministic transformation of route metrics, not another neural network, and is absent at inference.

I.6 Notation

Table 30 summarizes the notation used in the method and Algorithms 1 and 2. Within a fixed semantic group, we suppress the group subscript in $x, r_i, c, q^c, \mathcal{P}_c$, and $\mathcal{N}_c$; superscript $\star$ denotes a validation-selected parameter set. In tables and plots, α abbreviates α_{eval}. Attention diagrams retain the standard $Q/K/V$ meanings; κ denotes a projector-sharing group. Support size K, raw-draw budget B, and returned-list cutoff k are distinct. Contrast order and sign conventions follow the captions.

Table 30: Recurring notation across support construction, two-stage training, and evaluation.

Symbol	Definition
Support and supervision	
$x,\ r_i$	Product graph and candidate precursor-set graph.
$g = (x, c, \mathcal{C}_g, \mathcal{Y}_g)$	Product, instruction, indexed pool, and supervision record.
$\mathcal{D}_{\text{tr}}, \mathcal{D}_{\text{val}}$	Labeled groups for training and validation.
$K,\ K_x,\ \mathcal{A}_K(x)$	Nominal support size, realized size, and retained ordered template-expansion prefix.
$\mathcal{P}_c,\ \mathcal{N}_c,\ q^c$	Preferred/avoided index sets; listwise target distribution.
Stage A: grounding	
$E_T, H_c;\ E_R, v_i$	Frozen text/route encoders; token/route features.
$\phi, \phi^\star, P_\phi$	Projector parameters, selected parameters, and projector.
$Z_c, z_c, \widehat{q}^c_\phi, b_{g,i}, \tau$	Projected tokens, pooled embedding, predicted pool distribution, endpoint prior, and similarity-logit scale.
$\mathcal{L}_A, \mathcal{L}_{\text{pair}}; \lambda_{\text{list}}, \lambda_{\text{pair}}$	Stage-A loss, pairwise term, and listwise/pairwise weights.
Stage B: residual steering	
$\theta, \theta^\star;\ P_{\kappa,\theta}, A_{\ell,\theta}$	Trainable/selected parameters; projector; node-to-text adapter.
$\mathcal{I}, \mathcal{G}, \kappa, \gamma;\ M_{c,\kappa}$	Injection layers; projector groups, index and assignment; projected token memory.
$M_c^{(\ell)}, M_c$	Per-layer and grouped memories (Eq. 14).
$h_\ell, B_\ell;\ \alpha_{\text{train}}, \alpha_{\text{eval}}$	Node state, frozen block, and train/eval residual scales.
$d_{g,i} = D^{\text{ref}}_{g,i} - D^{\text{student}}_{g,i}$	Guidance-induced reduction in complete-route denoising cost.
$\mathcal{L}_{\text{pos}}, \mathcal{L}_{\text{rank}}, \mathcal{L}_{\text{roll}}, \mathcal{L}_{\text{guard}}$	Preferred-endpoint, ranking, rollout-KL and reference-guard terms of $\mathcal{L}_{B,\text{core}}$.
$\mathcal{L}_{\text{teach}}, \mathcal{L}_{B,\text{metric}}$	Metric-teacher KL and metric-augmented Stage-B loss.
Sampling and evaluation	
$B, S;\ \rho, G_t$	Raw draw budget, flow steps, RC proposal and graph at t.
$k;\ \mathrm{P@}k, \mathrm{N@}k$	Returned-list cutoff and preferred/avoided set hit rates.
$\Delta, \Delta\Delta$	Named contrast; difference of contrasts.

I.7 Implementation and reproducibility details

Objective coefficients. In the notation of Eq. (2), Stage A uses $\lambda_{\text{list}} = 1.0$, $\lambda_{\text{pair}} = 0.5$, pair margin $m_{\text{pair}} = 0.1$, and fixed logit scale $\tau = 10$. Stage B uses $\lambda_{\text{pos}} = 0.10$, $\lambda_{\text{rank}} = 0.05$, $\lambda_{\text{roll}} = 1.00$, $\lambda_{\text{guard}} = 0.05$, and $\delta_{\text{KL}} = 0.02$. Structure guidance uses this core objective directly, whereas the metric branches set $\lambda_{\text{teach}} = 0.05$ in Eq. (18) and use $T_q = T_s = 1$ in Eq. (17). We set candidate-score calibration to `none`; Appendix I.5 specifies the exact score and teacher reductions.

Training settings. Stage A and Stage B use learning rates 10^{-4} and 3×10^{-5}, respectively. For RDKit5 Common Top100, both model sizes train for 30/14 epochs (Stage A/Stage B), with effective batches of 512 semantic groups for USPTO-MR and 32 for synthetic supports. Training uses BF16 mixed precision and seed 42 (43 and 44 for Stage-B robustness repeats), with $\alpha_{\text{train}} = 1$ fixed throughout Stage B. Table 31 summarizes Structure training and selected checkpoints; RDKit5 Common Top100 and Metric4 evaluation settings appear in Tables 5 and 22.

Table 31: Structure Top30 training and validation-frozen checkpoints. Training epochs are Stage A/Stage B; the selected Stage-A epoch initializes Stage B, whose test setting is $(e, \alpha_{\text{eval}})$. Both capacities use Matched-P@1-first selection.

Suite / scale	Training epochs Stage A / B	Stage-A selected epoch	Stage-B test $(e, \alpha_{\text{eval}})$
Structure / 8M	30 / 30	28	$(22, .50)$
Structure / 65M	20 / 30	20	$(26, 1)$

Inference implementation. Inference scales only the cross-attention output residual by α_{eval}, not backbone states, Reference logits, or loss weights. Appendix H.9 specifies sampling, readout, and uncertainty.

CTMC sampling. We retain the Base proposal schedule, graph initialization, and CTMC sampler (Wang et al., 2026). Given product x, instruction c, and reaction-center proposal ρ, the guided model $f_{\theta,\alpha}$ predicts a factorized distribution $\widehat{p}_t$ over terminal precursor graphs from the current graph G_t at time $t \in [0, 1]$. The guided transition rates are

$$\widehat{R}_t(G_t, G') = \mathbb{E}_{\widetilde{G}_1 \sim \widehat{p}_t} \left[R_t(G_t, G' \mid G_0, \widetilde{G}_1) \right], \tag{19}$$

where R_t is the Base sampler's fixed conditional rate and $\widetilde{G}_1$ is a possible terminal graph marginalized under the model prediction, not a ground-truth input. The unchanged first-order update is

$$\Pr(G_{t+\Delta t} = G' \mid G_t, x, \rho, c) \approx \mathbf{1}\{G' = G_t\} + \Delta t \, \widehat{R}_t(G_t, G'). \tag{20}$$

Here $\Delta t = 1/S$, with $S = 50$ steps in the reported evaluations.

J Limitations and Future Work

Our evaluation focuses primarily on single-step retrosynthesis. The multistep case study illustrates how RIGS can guide a planner, but we have not systematically evaluated its benefits for multistep planning. Future work will integrate RIGS into LLM-guided planning and explore post-training to improve output validity and instruction following.