LEARNING NAVIGATION CORRECTIONS FROM A WEAK RECOVERY POLICY

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning to recover from navigation errors can disrupt actions that an agent already performs well. Yet a recovery policy that is unsuitable for deployment may still offer useful local supervision. We propose RCSR-D, a training procedure that separates the usefulness of a policy's action proposals from its ability to navigate complete trajectories. A frozen recovery policy supplies candidate turns. An offline rule selects proposals under empirical retention criteria, and a student initialized from the original policy learns the selected first actions while matching reference behavior on the remaining states. Parameter interpolation moderates the learned update before deployment. On the same reserved 128-episode cohort in the navigation-tuned 9B setting, the original and recovery policies achieve 33.59% and 6.25% success, respectively, while RCSR-D students achieve 42.97–51.56% across three training seeds. On the full 1,839-episode R2R-CE validation-unseen split, the primary student improves success from 37.25% to 46.93% and success weighted by path length from 29.36% to 36.28%. These results show that a recovery checkpoint can remain a useful training resource despite poor navigation performance. RCSR-D provides a way to convert its local proposals into improved navigation while deploying a single student.

1 INTRODUCTION

Corrective post-training offers a way to improve a navigator's decisions in states where it fails. In vision-and-language navigation (VLN), the agent follows an instruction through a sequence of observations and actions (Anderson et al., 2018; Krantz et al., 2020). An incorrect action changes the observations that follow, motivating supervision on states reached by the policy itself (Ross et al., 2011). Learning from these errors must also preserve useful behavior: an update that encourages corrective turns may disrupt forward motion or change when the agent stops.

A recovery checkpoint's overall success does not fully describe the value of its local decisions. Success measures what happens when the checkpoint controls an entire trajectory; a training target asks whether one proposed action helps another policy learn. Our navigation-tuned 9B recovery continuation exposes this distinction. When deployed directly, it succeeds on only 6.25% of a reserved 128-episode cohort, compared with 33.59% for the original policy. That poor deployment result motivates a different question: can some of its proposals improve a student that starts from the original navigator? Figure 1 summarizes this empirical contrast.

We investigate this question with Retention-Constrained Selective Recovery Distillation (RCSR-D). The method separates three decisions: which recovery proposals to use, how the student learns them, and what behavior it should retain elsewhere. An offline rule selects turn proposals under empirical action-wise retention criteria. A student initialized from the original policy learns these proposals through first-action cross-entropy and matches the original distribution on the remaining training states. Fixed parameter interpolation limits the final update. The deployed student predicts its own actions; neither the recovery policy nor the selection rule runs at inference.

The 9B results support this separation between deployment quality and supervision value. The primary student improves success by 9.68 percentage points and path-length-weighted success by 6.92 points on all 1,839 public R2R-CE validation-unseen episodes. Students trained with all three seeds improve on the shared 128-episode subset containing the weak-policy comparison. On a fixed 512-episode subset, the student also outperforms the recovery policy directly interpolated toward

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

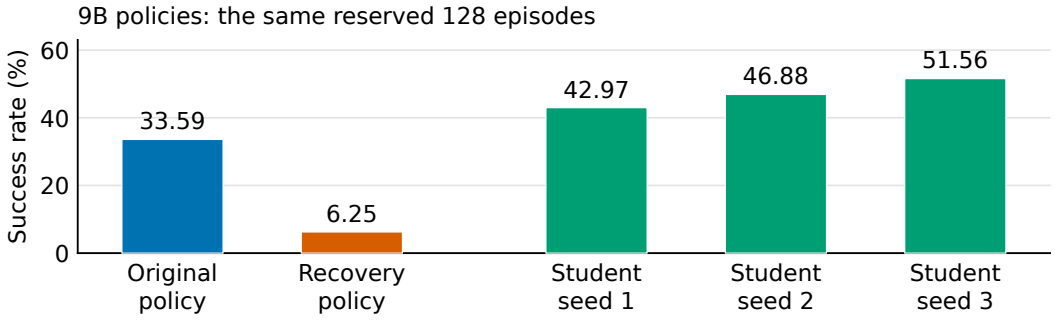

Figure 1: **A weak navigator can supply useful training proposals.** All five policies are evaluated on the same reserved 128-episode cohort. The recovery policy is deployed directly in its own evaluation; the three students learn through RCSR-D using its proposals. Each student bar represents a separate training seed. The figure shows the observed training outcomes; matched component comparisons appear in Section 5.4.

the original at the same coefficient. Component controls test more specific explanations for these gains, and a complementary StreamVLN study examines recovery and retention using a different repair source and native action interface.

RCSR-D provides an explicit procedure for reusing recovery training: extract candidate supervision, transfer selected proposals to a student initialized from the original policy, and match reference behavior on the remaining states. The empirical results show that this procedure improves closed-loop navigation using a proposal source that performs poorly when deployed directly. This makes local corrections a reusable product of recovery training, even when its final checkpoint is unsuitable for deployment.

2 Related Work

Learning from imperfect models. Useful supervision from a weaker model has precedents beyond navigation. Weak-Driven Learning uses weak checkpoints and entropy dynamics to improve reasoning and code models (Chen et al., 2026), while Armandpour et al. (2026) analyze when on-policy distillation gradients help or hurt reasoning models. Our study brings this perspective to navigation recovery: it examines how a recovery policy's proposals can improve the original navigator using fixed offline states and action-level distillation.

Corrective supervision in navigation. Learning from policy-induced errors addresses the state-distribution shift that makes sequential prediction difficult (Ross et al., 2011). CorrectNav turns navigation failures into further perception/action supervision (Yu et al., 2026); SACA identifies useful prefixes and divergence points in imperfect trajectories to form scenario-dependent learning groups (Li et al., 2026). Our focus is the value of the resulting recovery policy as a proposal source, even when deploying it directly is unsuccessful. NaVIDA's inverse-dynamics supervision and action chunking (Zhu et al., 2026) and StreamVLN's streaming slow–fast context (Wei et al., 2025) provide distinct native interfaces for the complementary studies.

Retaining reference behavior. Distillation transfers predictive distributions (Hinton et al., 2015) and can preserve earlier capabilities during adaptation (Li & Hoiem, 2016). Parameter interpolation provides another established way to moderate fine-tuning: WiSE-FT combines original and adapted weights (Wortsman et al., 2022b), and model soups combine fine-tuned models without inference-time ensembling (Wortsman et al., 2022a). These techniques supply tools for reference retention. RCSR-D combines them with proposal extraction to learn useful corrections from a weak navigation recovery policy.

Selection and fallback. Selective supervision also relates to expert selection and baseline fallback. SAME adapts expert specialization across navigation tasks (Zhou et al., 2024); our offline teacher

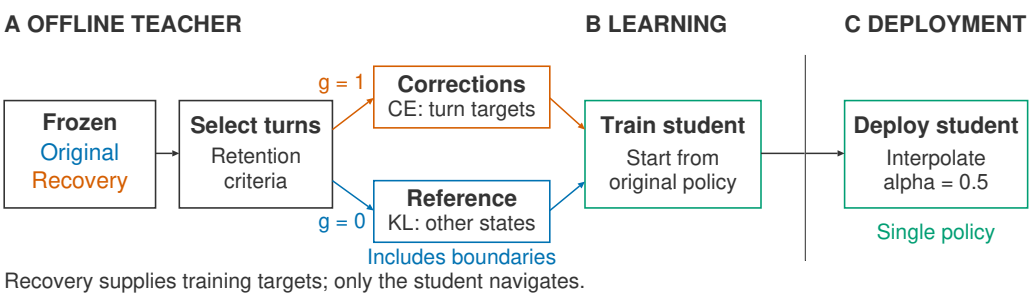

Figure 2: **Learning from proposals without deploying the recovery policy.** Offline selection assigns accepted turn proposals to corrective CE and the remaining states to reference KL. Both channels train a student initialized from the original policy. Fixed parameter interpolation precedes deployment; only the student predicts navigation actions. Boundary states always use KL.

draws supervision from two fixed policies during training. Safe policy improvement analyzes baseline fallback under explicit assumptions (Laroche et al., 2019). Here, action-wise development tests guide which recovery proposals enter the teacher. These connections motivate the use of selection and retention in the training procedure.

3 Extracting Recovery Supervision

3.1 Separate supervision from the deployed policy

Extracting useful proposals requires separating the source of supervision from the model that will navigate. We distinguish four roles (Figure 2). The *original policy* π_0, called Natural in the experiments, supplies the student's initialization and the reference behavior to retain. The *recovery policy* π_r supplies candidate corrections. An *offline teacher* combines supervision from these two frozen policies. The *student* π_θ learns from that supervision and is the only model deployed.

The recovery policy specializes in correcting errors made by the original navigator. It is trained on policy-visited states using cross-entropy on 2,048 original-policy errors and reference distillation on 2,176 competent or termination-boundary states. The student later uses the same 4,224-state source pool. Competent states are those where the original action agrees with the shortest-path teacher; boundary states concern stopping decisions. Each input x contains an instruction and RGB observation history; the first action belongs to $\mathcal{A} = \{\mathsf{Stop}, \mathsf{Forward}, \mathsf{Left}, \mathsf{Right}\}$. Shortest-path actions and distances define training strata and assess retention, but are not added to the student's input. This construction lets the recovery policy offer supervision without replacing the original policy as the student's starting point.

3.2 Select proposals under empirical retention criteria

Proposal selection addresses the mismatch between a specialist's local corrections and its overall behavior. We restrict delegation to confident turn proposals and retain the original policy's stopping decisions in the offline teacher. The fixed scorer first separates Stop from continuing, compares `Move` and `Turn`, and resolves left versus right conditionally on a turn. Let $a_0(x)$ be the original action and $a_r^{\mathrm{mov}}(x)$ the recovery policy's movement proposal. Its turn margin is

$$m_r(x) = z_r(\texttt{Turn} \mid x) - z_r(\texttt{Move} \mid x), \tag{1}$$

using logits at the fixed first-action prefix. With termination-boundary set $\mathcal{B}$, the offline delegation rule is

$$g_\tau(x) = \mathbf{1}[x \notin \mathcal{B}]\,\mathbf{1}[a_0(x) \neq \mathsf{Stop}]\,\mathbf{1}[a_r^{\mathrm{mov}}(x) \in \{\mathsf{Left}, \mathsf{Right}\}]\,\mathbf{1}[m_r(x) \geq \tau]. \tag{2}$$

The offline teacher $\pi_{T,\tau}$ chooses a_r^{mov} when $g_\tau = 1$ and a_0 otherwise. Boundary states and states where the original policy chooses Stop remain with the reference. This is a training-time assignment; the deployed student predicts its own Stop.

Calibration makes the tradeoff between corrective proposals and retained actions explicit. On 3,200 non-boundary training states, we choose τ to maximize recovery-turn macro accuracy subject to the following empirical constraint for each competent movement class k, where $\widehat{A}_k$ denotes empirical action accuracy:

$$\widehat{A}_k(\pi_{T,\tau}) - \widehat{A}_k(\pi_0) \geq -\epsilon_k, \qquad k \in \{\mathsf{Forward}, \mathsf{Left}, \mathsf{Right}\}, \tag{3}$$

with a five-percentage-point tolerance and a required recovery gain of at least 20 points. Ties favor higher preservation accuracy, then a larger value of τ. The selected threshold is $\tau = 2.25$. Oracle information enters the calibration criterion through the training labels; the routing decision in Eq. 2 does not query an oracle action. A rule is accepted only when both recovery and retention criteria are met. These criteria test the offline teacher on training states, not the student's future trajectories.

Calibration outcomes. At $\tau = 2.25$, the offline teacher's recovery-turn accuracy increases from 1.27% to 45.26% on the calibration states. Mean accuracy across the three competent movement strata changes from 92.01% to 92.10%; the largest class-specific decrease is 4.95 points for Forward, within the five-point tolerance. These are training calibration measurements. Selection applies to both error and competent strata, so the number of delegated rows is not the number of corrected errors.

3.3 Distill corrections and retain the reference

The student should learn accepted corrections without inheriting the recovery policy's other decisions. We initialize θ from θ_0 and supervise two disjoint sets of training states. Selected states use cross-entropy on the offline teacher's proposed action; the remaining states match the original policy's distribution. Let $I(x)$ contain the token positions of the first action in a canonical four-action response. Prefixes, the remaining three actions, and the response-ending token are masked out. The selected target y^T encodes the teacher's action, not an oracle label substituted after selection.

Both losses act on the same first-action positions. Writing $p_{\theta,t}$ for the student's next-token distribution under the fixed training prefix, the objective is

$$\ell_{\mathrm{CE}}(x) = -\frac{1}{|I(x)|} \sum_{t \in I(x)} \log p_{\theta,t}(y_t^T), \tag{4}$$

$$\ell_{\mathrm{keep}}(x) = \frac{1}{|I(x)|} \sum_{t \in I(x)} D_{\mathrm{KL}}(\bar{p}_{0,t} \,\|\, \bar{p}_{\theta,t}), \tag{5}$$

$$\mathcal{L}(\theta) = \mathbb{E}_{x \sim \mathcal{D}} \left[g_\tau(x) \ell_{\mathrm{CE}}(x) + (1 - g_\tau(x)) \ell_{\mathrm{keep}}(x) \right]. \tag{6}$$

Here the bar denotes renormalization of both distributions on the original policy's top-128 token support at each position. This is a top-k approximation to reference KL, with temperature one. Both loss coefficients are one, so the mixture follows the fixed row counts: 1,653 CE rows and 2,571 preservation rows. All 1,024 boundary states use preservation. Boundary metadata determine the training channel but are not part of the instruction or images. Of the selected rows, 1,086 come from the error stratum and 567 from the competent stratum; 1,105 change the original semantic action.

Distribution matching expresses retention differently from hard-label imitation. For a preserved token, KL has gradient $\bar{p}_{\theta,t} - \bar{p}_{0,t}$ with respect to the supported logits. Hard-label CE has gradient $p_{\theta,t} - e_{y_t^0}$, where $e_{y_t^0}$ is the one-hot reference target. KL is stationary when the supported distributions match; CE continues to sharpen its target. This local distinction motivates the matched preservation control in Section 5.4, where navigation outcomes determine its empirical value.

3.4 Interpolate parameters before deployment

Interpolation reduces the magnitude of the learned update before deployment. For student parameters θ and original parameters θ_0, we use

$$\theta^* = \theta_0 + \alpha(\theta - \theta_0), \qquad \alpha = 0.5, \tag{7}$$

The coefficient was chosen during development after the uninterpolated student reduced Right accuracy by 7.81 points, then fixed before the reported unseen evaluations and reused across seeds.

The separate 1,600-state validation-seen screen requires a recovery gain of at least 15 points with a positive paired interval. Per-action accuracy, terminal recall and the terminal–hard-negative Stop gap may decrease by at most five points; hard-negative false Stop may increase by at most five points. False Stop on movement strata must remain at most 3%. Interpolation scales the parameter difference by α; these empirical criteria do not make it an optimization projection with a proved behavioral guarantee.

The resulting policy π_{θ^*} navigates using the original input and action-selection interface. The teacher and selection rule are discarded. Single-student deployment removes the second policy, while the chosen actions and trajectory lengths still determine the inference workload.

4 Experimental Design

Models and training. The main study starts from a locally navigation-tuned Qwen3.5-9B checkpoint (Qwen Team, 2026), held fixed as Natural. Its 4,224 training states cover 635 recorded trajectories and 61 canonical training scans, disjoint from the 11 full-evaluation scans. Each student receives 48 updates with a global batch size of 128 and a learning rate of 2×10^{-6}; all vision, alignment and language parameters are trainable. Training uses the ms-swift/Megatron GKD path on fixed offline data and the final checkpoint for each seed. The inherited recipe configures one multi-token-prediction layer with coefficient 0.1, held constant across students. The learning rate follows a cosine schedule to 2×10^{-7} after a 5% warmup. Adam uses weight decay of 0.1 and gradient clipping at one. Training uses BF16, a microbatch size of one and four-way tensor parallelism on 128 GPUs. Appendix A records inherited settings and seed identities.

Evaluation populations. The full 1,839-episode public R2R-CE validation-unseen split measures the method's effect for one fixed primary 9B training seed. A reserved 128-episode subset supports the three-seed comparison, direct deployment of the recovery policy, and matched component controls. This subset comprises two precommitted disjoint 64-episode waves that exclude episodes accessed by earlier router/RGB tests. A fixed scene-stratified 512-episode subset tests direct interpolation of the recovery checkpoint. Both subsets overlap the full split and reuse its original/primary-student outcomes; they are not additional independent benchmarks. The 1,600-state validation-seen retention screen is a separate development evaluation. Native-action comparisons also use the full public split, with their own fixed reference policies.

Navigation interface. The 9B agent executes one first-action decision at a time through the same factorized scorer for Natural and every student. Inputs contain at most 12 RGB frames selected by the fixed `key_recent` sampler; forward actions move 0.25 m and turns rotate 30°. A 162-step cap triggers an evaluator-issued Stop. We distinguish endpoint success, which includes this forced Stop, from success requiring a model-chosen Stop. StreamVLN and NaVIDA retain their native interfaces, so comparisons are made within each setting.

Metrics and uncertainty. Success rate (SR) requires stopping within 3 m geodesic distance of the goal; oracle success (OS) requires reaching that region at any time. SPL weights success by the shortest-path length divided by the larger of the executed and shortest-path lengths. Navigation error (NE) is the final geodesic distance to the goal. SR, SPL and OS are percentages; NE is in meters. Intervals use 10,000 paired episode bootstrap draws. The main 9B and fixed-interpolation comparisons use 95% intervals; the prespecified 9B component and seed-specific StreamVLN KL–CE contrasts use 97.5%. Seed-specific StreamVLN comparisons with the parent and released model are secondary and use unadjusted 95% intervals. These intervals condition on the evaluated episodes and trained models. Three-seed deviations are sample standard deviations. StreamVLN seed means average paired per-episode outcomes before resampling; they neither resample training seeds nor describe an ensemble. Scene-cluster and leave-one-scene-out checks are post-hoc sensitivity analyses, reported separately from the episode intervals.

Table 1: RCSR-D improves navigation on all 1,839 public R2R-CE validation-unseen episodes. Results use one fixed primary 9B training seed. Differences are student minus Natural with paired 95% intervals; percentage metrics use percentage-point differences.

Metric	Natural	RCSR-D	Difference [95% CI]
SR ↑	37.25	**46.93**	$+9.68$ $[7.40, 11.96]$
SPL ↑	29.36	**36.28**	$+6.92$ $[5.22, 8.62]$
OS ↑	65.63	**73.19**	$+7.56$ $[5.49, 9.68]$
NE (m) ↓	7.63	**5.90**	-1.73 $[-2.06, -1.42]$

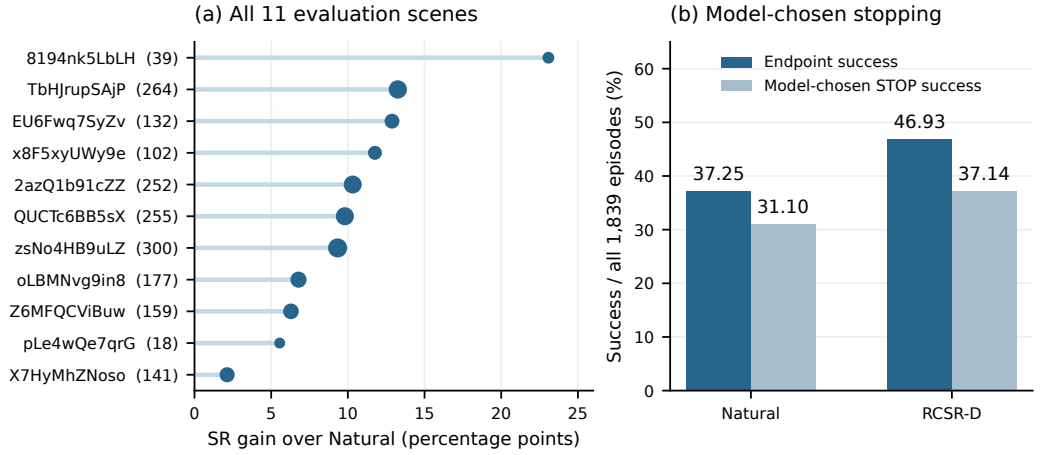

Figure 3: **Does the gain depend on one scene or forced stopping?** Left: SR increases in each of the 11 evaluated scenes; labels give scene identities and episode counts. These are descriptive scene effects. Right: the student improves both endpoint success and success requiring model-chosen Stop. Both rates use all 1,839 episodes as the denominator. The panels reanalyze the same primary-seed comparison.

5 Results

5.1 RCSR-D improves closed-loop navigation

The first question is whether extracting recovery supervision improves the policy that actually navigates. On the full public split, the primary student raises SR from 37.25% to 46.93% and SPL from 29.36% to 36.28% (Table 1). It succeeds on 863 episodes rather than 685, a net gain of 178. NE falls by 1.73 m, and all four paired metric intervals exclude zero. The gain therefore concerns successful completion and path efficiency, as well as reaching the goal region.

The improvement persists when success requires the model itself to stop. Model-chosen-Stop success rises from 31.10% to 37.14%, a 6.04-point gain with a 95% episode interval of $[4.08, 7.99]$. All 1,839 episodes remain in its denominator. Endpoint SR also benefits from successful budget-forced stops; Natural records 572 successes with model-chosen Stop and 113 after the forced Stop, compared with 683 and 180 for the student. The premature model-Stop rate falls from 14.19% to 11.20%, while the timeout rate falls from 54.70% to 51.66%. Thus both model-chosen completions and successful timeout endpoints contribute to the SR gain.

The gain is also distributed across the observed scenes (Figure 3). SR increases in all 11, with a 95% scene-cluster interval of $[7.63, 11.53]$ points. Leaving out any one scene gives gains from 9.08 to 10.31 points. Model-chosen-Stop success has a scene-cluster interval of $[3.60, 8.46]$. These post-hoc checks show that the result is not concentrated in one observed scene; they do not establish generalization beyond the 11 evaluated scenes.

Table 2: Seed-wise navigation results on the shared 128 episodes. SR gains are relative to Natural, with paired 95% intervals. SPL is a percentage and NE is in meters; Natural obtains 28.12% SPL and 8.12 m NE. Seed 1 is the fixed primary model used for the full-split evaluation.

Student	ΔSR [95% CI]	SPL $\uparrow$	NE $\downarrow$
Seed 1	$+9.38$ $[0.78, 17.97]$	34.42	6.09
Seed 2	$+13.28$ $[3.91, 22.66]$	38.51	5.73
Seed 3	$+17.97$ $[9.38, 26.56]$	39.56	5.46

Table 3: The student exceeds direct recovery interpolation on the fixed 512-episode subset. Both adapted policies use $\alpha = 0.5$. SR and SPL are percentages; the last column is student minus interpolated recovery (paired 95% interval).

Policy	SR	SPL	ΔSR [95% CI]
Natural	40.04	30.59	—
Recovery + interpolation	28.91	26.82	—
RCSR-D	48.24	37.08	$+19.34$ $[+14.45, +24.41]$

5.2 A POOR RECOVERY NAVIGATOR SUPPLIES USEFUL SUPERVISION

Direct deployment and supervision extraction give sharply different outcomes for the same recovery checkpoint. On the reserved 128 episodes, the recovery policy reaches 6.25% SR against Natural's 33.59%, whereas the three students trained from its proposals reach 42.97%, 46.88% and 51.56% (Figure 1; full metrics in Appendix A). Each seed-specific SR and NE interval excludes zero in the direction of improvement. Across seeds, the mean SR gain is 13.54 ± 4.30 points and the mean NE reduction is 2.36 ± 0.32 m. The fixed primary seed, rather than the best seed on this cohort, supplies the full-split result. Table 2 reports the seed-specific uncertainty and path-quality metrics alongside the SR comparison in Figure 1.

The recovery continuation shares the students' initialization, source-state identities and 48-update budget, but uses oracle-error CE targets and different preservation assignments. It times out on 96.09% of the reserved episodes when deployed directly. Its proposals nevertheless support students that improve over Natural. This contrast shows why deployment performance alone can miss a recovery checkpoint's value for training.

5.3 DISTILLATION VERSUS DIRECT WEIGHT INTERPOLATION

Directly interpolating the recovery checkpoint offers a simpler explanation to test: perhaps reducing its update is sufficient to obtain the student's gain. We compare that alternative with the primary student on the fixed 512-episode subset, using the same $\alpha = 0.5$, decoder and step limit. The student reaches 48.24% SR against 28.91% for interpolated recovery (Table 3). The paired differences are $+19.34$ SR points $[14.45, 24.41]$ and $+10.26$ SPL points $[6.04, 14.38]$ (95%). Distillation therefore yields higher SR and SPL than direct weight interpolation at this fixed coefficient.

5.4 EFFECTS OF PROPOSAL SELECTION AND REFERENCE RETENTION

Two matched controls test the roles of selection and retention on the reserved 128 episodes, holding the training budget and interpolation fixed. Hard preservation replaces KL with reference-token CE at the same teacher-forced positions. Stratified random delegation changes the selected states while matching supervision counts, source roles, candidate actions and semantic action-change status. Table 4 gives the navigation results; Appendix B specifies the interventions and prespecified SR criteria.

Soft preservation improves NE in this comparison, while its SR advantage remains unresolved. The primary student reaches 42.97% SR versus 37.50% for hard preservation: $+5.47$ points with a 97.5% interval of $[-3.91, 14.84]$. Its NE difference is -0.97 m $[-1.91, -0.05]$. The comparison supports an NE improvement; the primary SR contrast remains inconclusive.

Table 4: Matched 9B controls, 128 paired episodes. SR and SPL are percentages; NE is in meters. Differences are primary student minus each control, with 97.5% paired episode intervals.

Policy	SR	SPL	NE	ΔSR [97.5% CI]
Natural	33.59	28.12	8.12	—
Primary student	42.97	34.42	6.09	—
Hard preservation	37.50	29.44	7.07	$+5.47\ [-3.91, 14.84]$
Random delegation	42.19	35.20	5.88	$+0.78\ [-9.38, 10.94]$

Margin-based selection likewise has not demonstrated an SR advantage over stratified random delegation. The random control reaches 42.19% SR and 35.20% SPL, compared with 42.97% and 34.42% for the student. The SR difference is $+0.78$ points $[-9.38, 10.94]$ (97.5%). This interval supports neither superiority nor equivalence.

5.5 Recovered failures outweigh lost successes

Closed-loop improvement depends on both recovery and regression. In the full comparison, the student recovers 334 original failures but loses 156 original successes. For original success rate s_0, recovery rate r conditional on original failure, and regression rate h conditional on original success,

$$\Delta\mathrm{SR} = (1 - s_0)r - s_0 h = (G - H)/n. \tag{8}$$

Here $G = 334$, $H = 156$, $n = 1839$, $r = 334/1154$ and $h = 156/685$. This accounting identity explains the positive net SR change without implying that every successful behavior is preserved.

Local retention checks capture a different aspect of the update. On the 1,600-state development screen, the primary interpolated student improves recovery-turn macro accuracy by 24.80 points (95% CI $[21.09, 28.52]$) while meeting the movement and stopping tolerances. Across three seeds, the recovery gain is 24.02 ± 1.53 points. Expert-action macro accuracy improves by 3.13 points; Forward, Left and Right change by -3.13, $+9.38$ and $+3.13$ points. Terminal recall improves by 2.82 points, and the false-Stop rate on hard-negative states increases by 0.20 points, within the stated tolerances. These checks cover a fixed collection of states, whereas closed-loop evaluation involves histories generated by the student. Passing local checks therefore does not guarantee preservation of episode-level success. The recovery and regression counts describe the student's outcomes without isolating the effect of reference retention on regression.

5.6 Complementary evidence from stronger native-action policies

A complementary StreamVLN study asks whether recovery training and reference retention can improve an already adapted policy under its native action interface. This study uses simulator-verified repair targets and native four-action execution units, rather than the 9B specialist-margin teacher. Starting from a winner-CE parent at 58.13% SR, hard-target retention reaches 60.85% and 60.74% across two training seeds (Table 5). The parent-relative gains are $+2.72$ points $[0.60, 4.79]$ and $+2.61$ points $[0.60, 4.68]$ (secondary, unadjusted 95% intervals). The corresponding SPL gains are $+1.90$ points $[0.09, 3.70]$ and $+1.70$ points $[-0.08, 3.45]$ (95%); the second seed's SPL interval includes zero. Relative to the released model, the SR gains are $+5.27$ points $[3.10, 7.45]$ and $+5.17$ points $[3.05, 7.34]$ (95%). The gains extend the study of recovery and retention to an adapted parent under this native-action recipe.

The matched objective comparison tests whether soft retention improves on hard-target imitation in that recipe. Both ordinary-LoRA arms receive the same 4,096 exposures—1,024 repair examples and 3,072 preservation examples— and differ only in whether preservation uses KL or CE on the recorded parent response. Both arms use rank 32, scale 64, a learning rate of 2×10^{-5}, a global batch size of 128 and 32 updates. Initialization, repair targets and exposure order are matched within each seed. The primary KL-minus-CE SR contrasts are -1.09 points $[-3.15, 0.92]$ and -0.22 points $[-2.18, 1.69]$ (97.5%); the paired seed mean is -0.65 points $[-2.09, 0.79]$ (95%). Soft KL superiority is therefore not established. Without a repair-only control, gains over the parent also do not isolate the incremental effect of preservation.

Table 5: A complementary StreamVLN recipe improves a stronger parent. Each arm uses all 1,839 public validation-unseen episodes. The ordinary-LoRA objectives share repair data; parent-relative comparisons are secondary (95% paired intervals). The primary KL–CE contrasts use 97.5% intervals, reported in the text.

Policy / preservation objective	SR	SPL	ΔSR vs. parent [95% CI]
Released model	55.57	49.55	—
Winner-CE parent	58.13	52.49	—
Seed 1: hard CE	60.85	54.39	$+2.72$ $[0.60, 4.79]$
Seed 1: soft KL	59.76	53.21	$+1.63$ $[-0.38, 3.59]$
Seed 2: hard CE	60.74	54.20	$+2.61$ $[0.60, 4.68]$
Seed 2: soft KL	60.52	54.18	$+2.39$ $[0.44, 4.40]$

Additional native-model comparisons describe the current scope of these adaptations. In an earlier StreamVLN repair-CE/reference-KL study, protected LoRA reaches 59.38% SR versus 55.57% for the released model, but its $+0.11$-point difference from ordinary LoRA has a 95% interval of $[-1.74, 1.96]$. An anchored NaVIDA variant reaches 60.41% SR versus 59.76% for its original checkpoint, a $+0.65$-point difference $[-1.36, 2.61]$ (95%). Its SFT controls reach 60.63% and 59.49%, leaving the relative ordering unresolved. Appendix C retains the complete results and distinct recipes; Appendix D reports their development outcomes.

6 Discussion

A recovery checkpoint's value for training should be assessed separately from its performance over complete trajectories. RCSR-D makes this distinction operational by extracting proposals, choosing which to transfer, and retaining the original policy as both initialization and reference. The 9B students improve navigation despite the recovery policy's poor deployment performance. Recovery training can therefore produce useful supervision even when its final policy is unsuitable for direct use. The component comparisons suggest treating the selection rule and preservation loss as empirical design choices (Section 5.4).

Local retention and closed-loop evaluation address different aspects of learning. Fixed-state checks describe how the update affects reference decisions, while paired navigation records measure the balance of recovered failures and lost successes. Assessing both connects the training objective to the behavior of the deployed student.

Deployment workload. The shared factorized decoder performs an additional conditional pass for a turn, so single-model deployment does not imply equal forward counts. The full evaluation records 251,158 Natural forwards versus 270,497 student forwards. Mean recorded episode wall time is 2,358.6 versus 2,542.4 seconds under the parallel evaluation configuration. These are workload-dependent observations, not isolated latency benchmarks or a claim of unchanged latency.

Scope. The study covers navigation-tuned 9B policies on public R2R-CE data and complementary native-model recipes. Establishing how reliably other recovery checkpoints provide useful supervision requires additional checkpoints, independent datasets and evaluation in physical environments.

7 Conclusion

RCSR-D converts local recovery proposals into improved navigation by training a student from the original policy and matching reference behavior on the remaining states. In the navigation-tuned 9B study, students learn from a recovery checkpoint that performs poorly as a complete navigator and improve both success and path efficiency. Recovery checkpoints can thus be evaluated as sources of corrective supervision as well as candidates for deployment. RCSR-D demonstrates one way to reuse those corrections while deploying a single student.

AI Use Statement

AI tools were used only for language polishing of the manuscript, figure preparation, and identifying bugs in code. The authors reviewed the AI-assisted work and remain responsible for the final submission.

Ethics Statement

The reported experiments use existing navigation datasets and simulator rollouts, not new human-subject or real-robot experiments. Better aggregate navigation metrics do not establish deployment safety; real-world use requires independent safety evaluation and appropriate data and model licenses.

Reproducibility Statement

Sections 3 and 4 specify the supervision rule, token masks, approximation, training budget and evaluation controller. Run manifests, checkpoint identities and per-episode records were retained. The metric supplement audits the original 9B full-split comparison and the earlier native-action comparisons in Table 7; its current scope excludes the new interpolation and two-seed objective studies. An anonymous release of the trained initialization and specialist is still required for independent reproduction; this draft does not claim that such a public release already exists.

References

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3674–3683, 2018. URL https://arxiv.org/abs/1711.07280.

Mohammadreza Armandpour, Fatih Ilhan, David Harrison, Ajay Jaiswal, Duc N. M. Hoang, Fartash Faghri, Yizhe Zhang, Minsik Cho, and Mehrdad Farajtabar. Unmasking on-policy distillation: Where it helps, where it hurts, and why. *arXiv preprint arXiv:2605.10889*, 2026. URL https://arxiv.org/abs/2605.10889v2.

Zehao Chen, Gongxun Li, Tianxiang Ai, Zixuan Huang, Xiaodong Liu, Yifei Li, Wang Zhou, Fuzhen Zhuang, Xianglong Liu, Jianxin Li, Deqing Wang, and Yikun Ban. Weak-driven learning: How weak agents make strong agents stronger. *arXiv preprint arXiv:2602.08222*, 2026. URL https://arxiv.org/abs/2602.08222v2.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the navgraph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, 2020. URL https://arxiv.org/abs/2004.02857.

Romain Laroche, Paul Trichelair, and Rémi Tachet des Combes. Safe policy improvement with baseline bootstrapping. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3652–3661, 2019. URL https://proceedings.mlr.press/v97/laroche19a.html.

Haoyuan Li, Rui Liu, Hehe Fan, and Yi Yang. Let's reward step-by-step: Step-aware contrastive alignment for vision-language navigation in continuous environments. *arXiv preprint arXiv:2603.09740*, 2026. URL https://arxiv.org/abs/2603.09740.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *European Conference on Computer Vision*, 2016. URL https://arxiv.org/abs/1606.09282.

Cuong Anh Pham, Praneeth Vepakomma, and Samuel Horvath. Learning in the null space: Small singular values for continual learning. *arXiv preprint arXiv:2602.21919*, 2026. URL https://arxiv.org/abs/2602.21919.

Qwen Team. Qwen3.5-9B model card. Hugging Face model release, 2026. URL https://huggingface.co/Qwen/Qwen3.5-9B.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Meng Wei, Chenyang Wan, Xiqian Yu, Tai Wang, Yuqiang Yang, Xiaohan Mao, Chenming Zhu, Wenzhe Cai, Hanqing Wang, Yilun Chen, Xihui Liu, and Jiangmiao Pang. StreamVLN: Streaming vision-and-language navigation via slowfast context modeling. *arXiv preprint arXiv:2507.05240*, 2025. URL https://arxiv.org/abs/2507.05240.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998, 2022a. URL https://proceedings.mlr.press/v162/wortsman22a.html.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo-Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. *arXiv preprint arXiv:2109.01903*, 2022b. URL https://arxiv.org/abs/2109.01903.

Zhuoyuan Yu, Yuxing Long, Zihan Yang, Chengyan Zeng, Hongwei Fan, Jiyao Zhang, and Hao Dong. CorrectNav: Self-correction flywheel empowers vision-language-action navigation model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18737–18745, 2026. doi: 10.1609/aaai.v40i22.38942. URL https://ojs.aaai.org/index.php/AAAI/article/view/38942.

Gengze Zhou, Yicong Hong, Zun Wang, Chongyang Zhao, Mohit Bansal, and Qi Wu. SAME: Learning generic language-guided visual navigation with state-adaptive mixture of experts. *arXiv preprint arXiv:2412.05552*, 2024. URL https://arxiv.org/abs/2412.05552.

Weiye Zhu, Zekai Zhang, Xiangchen Wang, Hewei Pan, Teng Wang, Tiantian Geng, Rongtao Xu, and Feng Zheng. NaVIDA: Vision-language navigation with inverse dynamics augmentation. *arXiv preprint arXiv:2601.18188*, 2026. URL https://arxiv.org/abs/2601.18188.

Table 6: A poor navigator can supply useful supervision. All rows use the same reserved 128 episodes. The directly deployed recovery continuation is the frozen proposal source; the students learn from its proposals and use fixed parameter interpolation. Differences are student minus Natural (95% intervals).

Policy	SR	SPL	OS	NE	ΔSR [95% CI]
Natural	33.59	28.12	59.38	8.12	—
Direct recovery CE+KL	6.25	5.82	7.03	7.98	—
RCSR-D, seed 1	42.97	34.42	71.88	6.09	$+9.38$ $[0.78, 17.97]$
RCSR-D, seed 2	46.88	38.51	70.31	5.73	$+13.28$ $[3.91, 22.66]$
RCSR-D, seed 3	51.56	39.56	73.44	5.46	$+17.97$ $[9.38, 26.56]$

A Implementation and Evaluation Details

Training-state accounting. The 4,224-state pool comprises 2,048 original-policy error states and 2,176 competent/boundary states, including 1,024 boundary examples. Among the 1,653 delegated rows, 1,086 come from the error stratum and 567 from the competent stratum; 1,105 change the original semantic action. The remaining 2,571 rows receive reference KL. Thus neither delegation nor membership in an error stratum implies that the teacher supplies a new correct action. Recorded scan paths are normalized to 61 distinct training scans before checking their disjointness from the 11 evaluation scans.

Training seeds and inherited settings. The 9B seeds are 20260919, 20260920 and 20260921, applied to training and data order; seed 20260919 is the fixed primary model used for full-split evaluation. The StreamVLN retention-objective study uses seeds 20260926 and 20260927, with 16 training GPUs per arm, accumulation eight and global batch 128. Each arm uses its final 32-update adapter. These settings are distinct from the 9B all-parameter, 48-update procedure.

Paired records and repeated evaluations. The full 9B comparison contains 1,839 unique identities per arm, zero invalid rows and zero scorer errors. On the 128 identities shared with the earlier reserved-cohort run, every executed action and success outcome agrees for both Natural and the primary student. Worker assignments and measured execution times differ, supporting reproducibility across those run layouts. Repeated records are not counted as additional independent samples. For the fixed 512-episode interpolation control, a runtime interruption left 494 completed trajectories; those records were retained byte-for-byte and only the 18 missing outcomes were recovered. The reported comparison uses all 512 fixed identities. The two-seed StreamVLN comparison likewise contains all 1,839 unique episodes for each of the four newly trained arms, with the same original and parent records reused as paired references.

Training-seed comparison. Figure 1 presents SR for the same reserved 128 episodes. Table 6 also reports SPL, OS, NE and the paired SR intervals for all three training seeds. Seed 1 is the fixed primary model used for the full-split evaluation.

B Matched 9B Component Controls

Both controls use the primary seed's initialization, 4,224 source states, 48-update budget and $\alpha = 0.5$ interpolation. Hard preservation replaces reference KL with CE on the reference's argmax tokens at the same teacher-forced positions; it does not use independently generated reference responses. Random delegation preserves the 1,653/2,571 CE/preservation row counts while matching source role, candidate action and whether the semantic action changes; the reassignment changes 1,344 masks. All policies use the same reserved 128 episodes and controller. These are attribution comparisons on already accessed public validation, with original and primary-student trajectories reused.

Relative to hard preservation, the student improves SPL by $+4.98$ points $[-1.23, 11.24]$ and reduces NE by 0.97 m (student minus control: -0.97 $[-1.91, -0.05]$). Against random delegation, its SPL difference is -0.78 points $[-8.45, 6.56]$ and its NE difference is $+0.21$ $[-0.67, 1.09]$ (all intervals

Table 7: Additional full public validation-unseen comparisons using each backbone's native interface, 1,839 episodes per arm. Differences are percentage points relative to the panel's original checkpoint. Each recipe is specified separately; comparing scores across panels does not provide a controlled backbone comparison.

Policy / post-training variant	SR	SPL	ΔSR
StreamVLN: outcome-verified post-training			
Original	55.57	49.55	—
Winner-CE parent	58.13	52.49	+2.56
Repair + reference KL, ordinary LoRA	59.27	53.08	+3.70
Repair + reference KL, protected LoRA	59.38	53.04	+3.81
NaVIDA: anchored selective-distillation variant			
Original	59.76	53.04	—
SFT control (earlier teacher calibration)	60.63	53.75	+0.87
SFT control (matched teacher calibration)	59.49	52.75	−0.27
RCSR + protected Natural CE	60.41	53.87	+0.65

97.5%). The planned SR criterion required a positive interval, at least two points of improvement, and nondecreasing SPL; neither comparison passes. The NE result favors soft retention in this setting, while the primary SR attribution remains inconclusive. The reported comparison includes all 128 fixed identities. Two missing hard-preservation outcomes were recovered while retaining the other 126 completed records unchanged.

C Native-Action Comparisons

Two-seed retention objectives. The primary KL-minus-CE SPL differences are -1.17 points $[-2.94, 0.56]$ and -0.01 points $[-1.69, 1.66]$ (97.5%). The planned criterion for a stable KL advantage required positive seed-specific 97.5% SR intervals, nonnegative SPL point differences in each seed, and mean SR gain of at least two points. These criteria are not met. The positive hard-CE parent comparisons reported in the main text are secondary, unadjusted comparisons, not a redefinition of that primary criterion.

For completeness, the mean SR across the KL seeds exceeds that of Original by $+4.57$ points $[2.31, 6.82]$ and that of the parent by $+2.01$ points $[-0.11, 4.11]$ (97.5%). The paired seed-mean KL–CE SR interval in the main text uses 95% coverage. These are different contrasts and confidence levels. Seed-mean statistics average the two fixed models' per-episode outcomes before resampling episodes; they neither resample newly trained models nor describe a deployed ensemble.

Earlier parameterization comparison. Table 7 retains the separate full-split StreamVLN and NaVIDA studies. The StreamVLN arms share the 4,096-exposure repair-CE/reference-KL objective, parent, rank 32, scale 64, learning rate 2×10^{-5}, global batch 128 and 32-update budget. Ordinary LoRA trains both factors. Protected LoRA fixes one factor to repair-input directions after removing dominant preservation-input directions, following feature-subspace constraints in continual learning (Pham et al., 2026). All three adapted checkpoints were frozen before the full-split comparison, and the Original reproduction uses the same evaluation source.

Protected LoRA improves SR over Original by $+3.81$ points $[1.69, 5.93]$ (97.5%). It improves SPL over Original by $+3.49$ points $[1.70, 5.31]$; ordinary LoRA improves SR by $+3.70$ points $[1.25, 6.04]$ (both 97.5%). Protected minus ordinary LoRA is $+0.11$ SR points $[-1.74, 1.96]$ (95%) and -0.04 SPL points. Protected minus parent is $+1.25$ SR points $[-0.60, 3.15]$ (97.5%). The parameterization and parent intervals include zero, leaving their incremental contributions unresolved. Ordinary LoRA shares the repair and preservation losses and therefore controls parameterization, rather than removing the retention objective.

Anchored NaVIDA recipe. The anchored variant adds original-policy-label CE on protected states to the selective-distillation objective. Relative to Original, it improves SR by $+0.65$ points $[-1.36, 2.61]$ and SPL by $+0.83$ points. Against SFT with the earlier teacher calibration, its SR difference is -0.22 points $[-2.07, 1.69]$; against the matched-calibration SFT control it is $+0.92$

points $[-0.82, 2.72]$ (all SR intervals 95%). Its highest observed SPL in this panel is a point estimate, without an established superiority claim. Earlier development outcomes for both backbones follow below.

D Development Comparisons and Recipe Changes

This appendix preserves earlier development outcomes alongside the final full-split comparisons. They involve different interventions and cohorts, and are not pooled with the final results or reclassified by later outcomes.

NaVIDA development. In a 256-episode validation-seen comparison, the initial selective-distillation adaptation reaches 67.58% SR versus 65.63% for Original and 69.53% for interpolated SFT. The 1.95-point difference from Original has a paired interval including zero. A separate scene-held-out LoRA repair reaches 84.58% versus 85.60% SR on 389 episodes. The later full-split anchored variant in Table 7 adds protected-state Natural-label CE and therefore has a different objective from the initial 9B recipe.

StreamVLN development. In the original specialist screen, no tested margin threshold jointly satisfies the fixed recovery and Forward-retention requirements. Subsequent experiments use outcome-verified targets and native action chunks. Table 8 gives the earlier decision-margin comparison and the development evaluation of the final LoRA variants. These are 512-episode held-out-TRAIN evaluations: the scenes were excluded from the additional post-training fit partition, but were available for upstream checkpoint training. They are not R2R validation-unseen results.

Table 8: Earlier StreamVLN development comparisons, 512 episodes per arm. The protected-repair and decision-margin studies have different objectives; the winner-CE parent is the shared reference. All rows retain their original development interpretation.

Policy / variant	SR	SPL
Source v2 SFT	88.67	83.80
Winner-CE parent	89.65	85.20
Matched edit-CE control	87.89	82.62
Decision-margin variant	87.30	81.49
Repair + reference KL, ordinary LoRA	87.70	82.61
Repair + reference KL, protected LoRA	88.67	83.50

The decision-margin variant recovers 36 source-v2-SFT failures and loses 43 successes, a net loss of seven. In the final repair development comparison, protected LoRA recovers 11 parent failures and loses 16 successes. Its SR change is -0.98 points, with a 95% interval of $[-2.93, 0.98]$. Ordinary LoRA reaches 87.70% versus the parent's 89.65%. Thus the two final LoRA variants did not improve development SR over the parent. The later full validation-unseen evaluation was a prospectively recorded change in evaluation scope with fixed checkpoints. Its positive original-model comparisons do not reverse those development outcomes or establish a causal rule for how repair and regression rates transfer across scene distributions. No alpha, rank or checkpoint sweep was selected based on the full target outcomes.