Resource-Adaptive Foundation Model Reasoning via Semantic Coverage

Anonymous Authors[1]

Abstract

Foundation-model reasoning is often scaled with fixed trace, branch, or token budgets, but fixed budgets waste inference compute when many rollouts collapse to the same answer class, proof route, evidence pattern, or fix strategy. We formulate test-time reasoning as resource-adaptive portfolio construction: under a compute budget, select unfinished rollout actions that are likely to add residual task-relevant support not already present in the current portfolio. We introduce semantic coverage, a portfolio objective over evaluator-induced support targets, and show that fixed semantic coverage is normalized, monotone, and submodular; under batch separability, the expected gain from bounded completion-level rollout actions inherits the same structure. A residual-threshold identity shows that additive trace scoring overcounts support exactly by the expected excess multiplicity of duplicate residual hits, motivating Semantic-Coverage Portfolio Search (SCPS), a receding-horizon controller that scores actions by predicted residual support, predicted overlap, and cost. On a 256-question held-out MMLU-Pro set with Qwen3.5-9B, an exact-answer SCPS variant reaches 91.0% portfolio Pass@16, compared with 75.0% for tree-prefix and 74.2% for semantic-pruning baselines, while using 31.6% of their realized tokens. This is a portfolio-acquisition claim, not a terminal-selector claim: SCPS directs inference compute toward support the portfolio lacks.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

1. Introduction

Foundation-model inference is still often rigid: each input receives the same number of samples, tree expansions, verifier calls, or decoding tokens, regardless of how much additional computation is likely to help. AdaptFM asks how inference can adapt to budgets such as compute, latency, energy, memory, or cost while preserving output quality. For test-time reasoning, the central resource question is not only how much computation to spend, but what the next unit of computation should buy.

Trace count is a poor proxy for reasoning progress. A model asked for mechanisms behind a phenotype may generate ten plausible traces, but seven may restate the same pathway and the rest may cover only one more mechanism. Similar collapse appears in exact-answer reasoning, proof search, coding, and evidence synthesis: many outputs can share an answer class, proof route, bug mechanism, or evidence type. Useful computation should add support the current portfolio lacks, not merely another fluent trace.

We call this missing task-relevant support residual semantic support. A completed artifact is a finished output returned by a rollout, such as an answer, proof attempt, hypothesis, evidence summary, experiment proposal, or patch. Semantic coverage credits such an artifact only for support not already supplied by the prompt, context, or current portfolio. This differs from diversity for its own sake, because irrelevant novelty should not score; it also differs from terminal selection, because a later verifier, voter, auditor, scientist, user, or decision rule may still need to choose from the portfolio.

Figure 1 illustrates the control problem. At a planning round, the system has a prompt x, a completed portfolio S_t, unfinished frontier states, and a remaining resource budget. Resources may be decode tokens, wall-clock time, dollar cost, energy, memory pressure, tool calls, or a vector of such quantities; our experiment instantiates the resource readout primarily with realized tokens. The controller chooses bounded rollout actions before seeing their completions. A good action is likely to add residual support, unlikely to du-

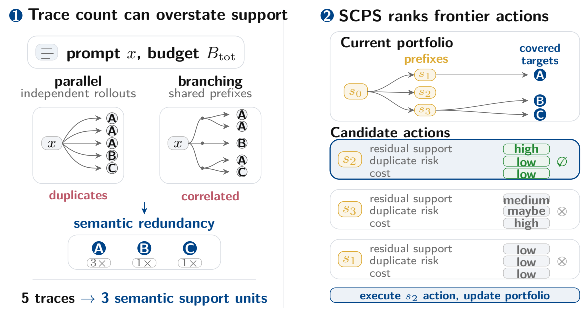

Figure 1. Trace count can overstate support. Independent and branching rollouts can collapse to fewer semantic support units; SCPS scores frontier actions by predicted residual support, duplicate risk, and cost before executing a budget-feasible batch.

plicate support acquired by other selected actions, and worth its cost.

We make this view actionable in three steps. First, for fixed evaluator-induced coverage profiles over semantic targets, semantic portfolio value is normalized, monotone, and submodular. Second, we lift the objective to completion-level macro-actions and show that the expected one-round gain remains normalized, monotone, and submodular under batch separability. Third, we show that additive trace scoring overcounts semantic gain by the expected excess multiplicity of duplicate residual hits. SCPS operationalizes this accounting as a receding-horizon controller that scores actions by predicted residual support, predicted overlap, and cost.

Our empirical evidence is deliberately scoped. In exact-answer MMLU-Pro, semantic targets are answer classes and portfolio Pass@16 asks whether the gold answer appears anywhere in the portfolio. On 256 held-out questions with Qwen3.5-9B, SCPS reaches 91.0% portfolio Pass@16 while using 31.6% of the realized tokens used by matched-protocol tree-prefix and semantic-pruning baselines. This is a single quality-resource operating point, not a full Pareto frontier or equal-token comparison.

2. Related Work

Adaptive inference-time compute. Test-time reasoning methods spend additional inference compute through samples, branches, verifiers, rerankers, budget allocation, rollout recycling, or feedback loops (Wei et al., 2022; Wang et al., 2022; Brown et al., 2024; Yao et al., 2023; Hao et al., 2023; Zhou et al., 2024; Cobbe et al., 2021; Lightman et al., 2023; Snell et al., 2024; Zhang et al., 2025; Wang et al., 2025b; Madaan et al., 2023; Shinn et al., 2023). SCPS is an acquisition controller: it asks which unfinished rollout action is likely to add support absent from the current portfolio under redundancy and cost, before a terminal selector consumes the portfolio.

Diversity, redundancy, and coverage. Prior work encourages, measures, or collapses variation through relevance-novelty reranking, diverse decoding, determinantal objectives, semantic entropy, semantic self-consistency, semantic clustering, pruning, state merging, and guided sampling (Carbonell & Goldstein, 1998; Kulesza & Taskar, 2012; Vijayakumar et al., 2018; Meister et al., 2021; Farquhar et al., 2024; Knappe et al., 2024; Shi et al., 2025; Choi et al., 2025; Lee et al., 2025; Tu et al., 2026; Song et al., 2025; Wang et al., 2025a; Hooper et al., 2025; Handa et al., 2025; Wang et al., 2026). Coverage functions and submodular maximization are classical (Nemhauser et al., 1978; Lin & Bilmes, 2011; Bordeaux et al., 2014; Golovin & Krause, 2011). Our contribution is the conversion of evaluator-induced semantic support into a resource-adaptive inference objective, the lift from completed artifacts to bounded rollout actions, and the exact overcount identity for duplicate residual support.

3. Semantic Coverage

A completed portfolio should be valued by what it supports. For intuition, suppose an evaluator induces a finite checklist of answer classes, proof routes, mechanisms, evidence types, tests, or interventions. A completion helps if it checks off a weighted item the current portfolio has missed. The general definition keeps this structure while replacing the checklist with a measurable semantic target space.

Definition 1 (semantic portfolio value). Fix a prompt x and planning round t. Let Y_x be the space of completed artifacts. Let $(\Theta_x, \Sigma_x, \mu_t)$ be a finite-measure space of semantic targets, where μ_t assigns current importance. Let $k_t(y, \theta) \in [0, 1]$ be the evaluator-induced degree to which artifact y covers target θ, and let $b_t(\theta) \in [0, 1]$ be baseline coverage already supplied before the portfolio. For finite $S \subseteq Y_x$, define

$$m_{t,S}(\theta) := \max\{b_t(\theta), \sup_{y \in S} k_t(y, \theta)\}, \quad \sup \emptyset := 0,$$

$$F_t^{\text{sem}}(S) := \int_{\Theta_x} [m_{t,S}(\theta) - b_t(\theta)] \, d\mu_t(\theta). \tag{1}$$

The value is residual: a completion earns credit only by improving coverage beyond what is already available. The evaluator defines what counts as support; the theorem states the optimization structure after those profiles are fixed.

Theorem 1 (latent semantic coverage). If b_t and $k_t(y, \cdot)$ are measurable and bounded in $[0, 1]$, then F_t^{sem} is normalized, monotone, and submodular over finite completed portfolios.

The proof is pointwise: for each target, adding artifact y gives marginal gain $[k_t(y, \theta) - m_{t,S}(\theta)]_+$, which can only decrease as S grows; integration preserves this inequality. Quality gates can zero or downweight invalid, unsupported, implausible, or irrelevant artifacts by replacing k_t with any fixed bounded admissibility-weighted kernel. Appendix B gives the full proof, local redundancy bounds, component coverage, concave evidence variants, and the strict-complementarity boundary.

Exact-answer quotient. Exact-answer tasks are the cleanest observable instance. Targets are answer classes, duplicate traces that canonicalize to the same class add no new coverage, and portfolio Pass@N asks whether search has acquired the true class. Let I_t be the information available at round t, $Z_x^{\downarrow}$ the completed trace space, A_x the answer-class space, $a : Z_x^{\downarrow} \to A_x$ the canonicalization map, and $A^\star \in A_x$ the latent correct class with posterior mass $\pi_t(\alpha) = \Pr(A^\star = \alpha \mid I_t)$. For finite or countable A_x, define

$$\begin{aligned} F_t^{\text{ans}}(S) &= \sum_{\alpha \in A_x} \pi_t(\alpha) \mathbf{1}\{\alpha \in A(S)\}, \\ A(S) &:= \{a(z) : z \in S\}. \end{aligned} \tag{2}$$

This is Equation (1) with $\Theta_x = A_x$, $b_t = 0$, and $k_t(z, \alpha) = \mathbf{1}\{a(z) = \alpha\}$. Appendix F treats answer banks when identity is latent, expensive, or only partially resolved.

4. Rollout Actions and Redundancy

A controller usually cannot choose completed artifacts directly. It chooses unfinished frontier actions before knowing which artifacts they will complete. A completion-level macro-action e is a bounded execution unit, such as sampling completions from a prefix, expanding a branch until horizon H, or spending a tool budget in one direction. Its semantic output $Z_e \subseteq Y_x$ is the random finite set of admissible completed artifacts returned during that execution; unfinished continuations return to the frontier and add no current portfolio value.

Fix round t, information state I_t, current portfolio S_t, and finite candidate action set E_t. A batch $B \subseteq E_t$ is selected in this frozen round before observing outputs. Define

$$\begin{aligned} U_B &:= \bigcup_{e \in B} Z_e, \\ G_t(B \mid S_t) &:= \mathbb{E}[F_t(S_t \cup U_B) - F_t(S_t) \mid I_t]. \end{aligned} \tag{3}$$

We assume batch separability: selecting other actions in the same batch does not physically change an action's current-round output, although outputs may be statistically dependent. If selected actions share a resource pool, the allocation must be part of the macro-action definition.

Theorem 2 (completion-level action lift). Under batch-separable macro-actions, if F_t is normalized, monotone, and submodular over completed portfolios and the expectation in Equation (3) is finite for every batch, then $G_t(\cdot \mid S_t)$ is normalized, monotone, and submodular over action batches.

For semantic coverage, the lift has an exact residual-threshold form. Let $u = (\theta, r)$, $d\nu_t(u) := dr\, d\mu_t(\theta)$, and let action e hit residual threshold u when

$$H_e(\theta, r) := \left\{ r > m_{t,S_t}(\theta) \text{ and } \sup_{y \in Z_e} k_t(y, \theta) \geq r \right\}.$$

Then

$$G_t^{\text{sem}}(B \mid S_t) = \int \Pr\left(\bigcup_{e \in B} H_e(u) \mid I_t \right) d\nu_t(u). \tag{4}$$

Each missing support threshold contributes once if at least one selected action hits it. With

$$\begin{aligned} v_t(e \mid S_t) &:= \int \Pr(H_e(u) \mid I_t)\, d\nu_t(u), \\ \omega_t(e, f \mid S_t) &:= \int \Pr(H_e(u) \cap H_f(u) \mid I_t)\, d\nu_t(u), \end{aligned}$$

the second-order Bonferroni lower bound is

$$G_t^{\text{sem}}(B \mid S_t) \geq \sum_{e \in B} v_t(e \mid S_t) - \sum_{\{e,f\} \subseteq B} \omega_t(e, f \mid S_t). \tag{5}$$

Proposition 3 (trace-additive overcount). Let $N_B(u) := \sum_{e \in B} \mathbf{1}\{H_e(u)\}$ be the number of selected actions that hit residual support unit u, and let $A_t^{\text{add}}(B \mid S_t) := \sum_{e \in B} v_t(e \mid S_t)$. Then

$$\begin{aligned} & A_t^{\text{add}}(B \mid S_t) - G_t^{\text{sem}}(B \mid S_t) \\ & \quad = \int \mathbb{E}[(N_B(u) - 1)_+ \mid I_t]\, d\nu_t(u). \end{aligned} \tag{6}$$

Thus additive trace scoring pays for every residual hit, while semantic portfolio value pays once per support unit. The gap is exactly expected duplicate residual-hit multiplicity. Appendix C gives proofs, higher-order bounds, and counterexamples for primitive token edges and interacting actions.

Algorithm 1 Semantic-Coverage Portfolio Search

```
1: Input: prompt x, budget B_tot, horizon H, rollout kernel, evaluator
2: Initialize portfolio S_0 ← ∅, frontier Q_0 ← {x}, memory M_0
3: for t = 0, 1, ... while budget remains do
4:     Build completion-level actions E_t from Q_t
5:     Estimate v̂_t(e | S_t, M_t), ĉ_t(e), and needed overlaps ω̂_t(e, f | S_t, M_t)
6:     Select an approximate maximizer B_t ⊆ E_t of Equation (7) over the feasible batch family
7:     if B_t = ∅ or L̂_t(B_t | S_t, M_t) ≤ 0 then
8:         break
9:     end if
10:    Execute B_t for up to horizon H; add admissible completions to S_t
11:    Update frontier, memory, and remaining budget
12: end for
13: return portfolio S_T to a terminal selector or report
```

5. Semantic-Coverage Portfolio Search

SCPS turns this accounting into a receding-horizon controller. At round t, it maintains a completed portfolio S_t, unfinished frontier $\mathcal{Q}_t$, and semantic memory M_t summarizing current coverage. It builds bounded actions, estimates their residual support, overlap, and cost, executes a feasible batch, updates the portfolio and memory, and replans. The resource cost may be a scalar token or latency estimate, or a scalarization of a vector budget; hard feasibility is enforced by the batch family.

The deployed score is the plug-in novelty-overlap-cost surrogate

$$\begin{aligned}\widehat{L}_t(B \mid S_t, M_t) := \sum_{e\in B} \widehat{v}_t(e \mid S_t, M_t) \\ - \beta_t \sum_{\{e,f\}\subseteq B} \widehat{\omega}_t(e, f \mid S_t, M_t) \\ - \lambda_t \sum_{e\in B} \widehat{c}_t(e).\end{aligned} \tag{7}$$

With true residual-hit probabilities, $\beta_t = 1$, and $\lambda_t = 0$, the first two terms are the Bonferroni lower bound in Equation (5). With learned estimates, tuned coefficients, and cost regularization, Equation (7) is a planning surrogate rather than an exact semantic-gain bound. In the exact-answer MMLU-Pro implementation, M_t is an option-answer bank; a frozen surrogate bundle predicts option/null distributions, answer collisions, and length/cost. SCPS returns a portfolio and does not choose the final answer.

Table 1. SCPS gives a strong quality-resource operating point. Results are on a 256-question held-out MMLU-Pro set with Qwen3.5-9B and maximum 16 rollouts. Pass@16 is portfolio coverage, not selected-answer accuracy. Bold marks the promoted portfolio/resource metrics.

Method	Pass@16 ↑	Mean tok. ↓	Unique ans. ↑	Dup. tok. frac. ↓
SCPS	91.0%	1120	3.47	55.4%
Tree-prefix	75.0%	3546	1.77	81.5%
Sem. pruning	74.2%	3548	1.77	81.3%

6. Experiments

We evaluate portfolio quality under a resource budget, not terminal answer selection. The main metric is Pass@16: whether the gold answer appears anywhere in the generated portfolio. This is the exact-answer coverage instance in Equation (2); selected-answer accuracy is reported only as a diagnostic.

Protocol. We use a 256-question held-out MMLU-Pro set with Qwen3.5-9B (Wang et al., 2024; Team, 2026). All methods use the same questions, held-out seed 47, run seed 13, temperature 0.6, maximum 16 rollouts, max output 192 tokens per completion, decode-token limit 3072, total-token limit 30000, answer-only multiple-choice prompting, and vLLM serving (Kwon et al., 2023). The learned SCPS surrogate bundle was trained on a disjoint 512-question controller-training split and was not trained or tuned on the 256-question evaluation set. Tree-prefix expands ranked frontier prefixes without residual-support or overlap scoring. Semantic pruning uses the same evaluator and predictor bundle but skips already-seen predicted answers or high-overlap prefixes instead of optimizing the residual-support objective.

Table 1 reports the current matched-protocol evidence. SCPS finds the gold answer in 233/256 questions, compared with 192/256 for tree-prefix and 190/256 for semantic pruning. Paired bootstrap intervals over questions are separated from zero: +16.0 percentage points versus tree-prefix (95% CI $[10.6, 21.5]$) and +16.8 versus semantic pruning ($[11.3, 22.7]$). The intervals do not include generation randomness, controller-training randomness, surrogate-training randomness, or serving nondeterminism.

The resource readout is also scoped. SCPS uses 1120.25 mean total tokens, compared with 3546.10 for tree-prefix and 3548.08 for semantic pruning, or 31.6% of their realized token use. This supports the claim that the selected controller avoids duplicate support under shared hard limits; it is not an equal-realized-token comparison and not a full multi-budget Pareto frontier. Appendix H.4 reports ablations: removing targeted answer-diverse action generation drops Pass@16 to 82.4%, while no-overlap and no-cost score ablations reach 90.2% and 89.1%. Independent self-consistency and semantic-diversity root-sampling baselines reach 78.1% and 78.9%. Several intervals among SCPS variants overlap, so these are diagnostic rather than a decisive variant hierarchy.

Selector conversion remains the main bottleneck. On the same SCPS portfolios, common-selector accuracy is 41.8% and majority accuracy is 65.2%, while portfolio Pass@16 is 91.0%. SCPS improves acquisition of represented support; terminal selection remains a separate layer.

7. Limitations and Conclusion

The theory is round-wise. It applies after the evaluator-induced objective, coverage profiles, action set, and resource-feasible family are fixed for the current planning round. Replanning after observations creates a new fixed objective; the theorems do not certify evaluator correctness, primitive token-level search, interacting actions, or terminal selection. The objective covers semantic coverage, exact-answer coverage, component coverage, concave evidence accumulation, and nonnegative channel sums, but not arbitrary strict complementarities unless the complementary bundle is modeled as the completed artifact.

The experiment is a live exact-answer portfolio-construction result on one held-out MMLU-Pro slice, one model, and one serving setup. It reports a strong token-based quality-resource operating point, not latency, energy, memory, dollar-cost, or hardware-aware optimization. The supplement supports audit of reported rates from compact per-question outcomes, method settings, split identifiers, and recomputation scripts; fresh live reruns additionally require benchmark access, Qwen3.5-9B access, a compatible vLLM stack, and a compatible trained predictor bundle. The available logs do not include a portable GPU model name, exact package lockfile, exact serving command, or full hardware record.

Semantic coverage reframes resource-adaptive reasoning as portfolio construction: the unit of progress is not another trace, but another piece of task-relevant support the current portfolio lacks. Once evaluator-induced coverage profiles are fixed, semantic portfolio value has diminishing returns, and the same structure lifts to batch-separable completion-level actions. SCPS uses predicted residual support, predicted overlap, and cost to allocate inference compute. The broader lesson for adaptive foundation-model inference is structural: spend computation where it expands support under the current budget, and report portfolio acquisition separately from terminal selection.

References

Bordeaux, L., Hamadi, Y., and Kohli, P. (eds.). Tractability: Practical Approaches to Hard Problems. Cambridge University Press, Cambridge, 2014. ISBN 978-1-107-02519-6. doi: 10.1017/CBO9781139177801.

Brown, B., Juravsky, J., Ehrlich, R., Clark, R., Le, Q. V., Ré, C., and Mirhoseini, A. Large Language Monkeys: Scaling Inference Compute with Repeated Sampling, December 2024.

Carbonell, J. and Goldstein, J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '98, pp. 335–336, New York, NY, USA, August 1998. Association for Computing Machinery. ISBN 978-1-58113-015-7. doi: 10.1145/290941.291025.

Choi, D., Lee, J., Tack, J., Song, W., Dingliwal, S., Jayanthi, S. M., Ganesh, B., Shin, J., Galstyan, A., and Bodapati, S. B. Think clearly: Improving reasoning via redundant token pruning. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), Findings of the Association for Computational Linguistics: EMNLP 2025, pp. 21437–21451, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.1169.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training Verifiers to Solve Math Word Problems, November 2021.

Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. Detecting hallucinations in large language models using semantic entropy. Nature, 630(8017):625–630, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07421-0.

Golovin, D. and Krause, A. Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization. Journal of Artificial Intelligence Research, 42:427–486, November 2011. ISSN 1076-9757. doi: 10.1613/jair.3278.

Handa, D., Parmar, M., Rrv, A., Uddin, M. N., Palangi, H., and Baral, C. GuidedSampling: Steering LLMs Towards Diverse Candidate Solutions at Inference-Time. In The Fourteenth International Conference on Learning Representations, October 2025.

Hao, S., Gu, Y., Ma, H., Hong, J., Wang, Z., Wang, D., and Hu, Z. Reasoning with language model is planning with world model. In Bouamor, H., Pino, J., and Bali, K. (eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 8154–8173, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.507.

Hooper, C., Kim, S., Moon, S., Dilmen, K., Maheswaran, M., Lee, N., Mahoney, M. W., Shao, S., Keutzer, K., and Gholami, A. ETS: Efficient Tree Search for Inference-Time Scaling, June 2025.

Knappe, T., Li, R. L., Chauhan, A., Chhua, K., Zhu, K., and O'Brien, S. Semantic Self-Consistency: Enhancing Language Model Reasoning via Semantic Weighting. In The 4th Workshop on Mathematical Reasoning and AI at NeurIPS'24, October 2024.

Kulesza, A. and Taskar, B. Determinantal Point Processes for Machine Learning. Foundations and Trends in Machine Learning, 5(2-3):123–286, December 2012. ISSN 1935-8237. doi: 10.1561/2200000044.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J., Zhang, H., and Stoica, I. Efficient Memory Management for Large Language Model Serving with PagedAttention. In Proceedings of the 29th Symposium on Operating Systems Principles, SOSP '23, pp. 611–626, New York, NY, USA, October 2023. Association for Computing Machinery. ISBN 979-8-4007-0229-7. doi: 10.1145/3600006.3613165.

Lee, S., Kim, H., Hwang, J., Park, E., and Ok, J. Efficient latent semantic clustering for scaling test-time computation of LLMs. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), Findings of the Association for Computational Linguistics: EMNLP 2025, pp. 24126–24144, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.1310.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let's Verify Step by Step. In The Twelfth International Conference on Learning Representations, October 2023.

Lin, H. and Bilmes, J. A class of submodular functions for document summarization. In Lin, D., Matsumoto, Y., and Mihalcea, R. (eds.), Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 510–520, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., Gupta, S., Majumder, B. P., Hermann, K., Welleck, S., Yazdanbakhsh, A., and Clark, P. SELF-REFINE: Iterative refinement with self-feedback. In Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, pp. 46534–46594, Red Hook, NY, USA, December 2023. Curran Associates Inc.

Meister, C., Forster, M., and Cotterell, R. Determinantal beam search. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 6551–6562, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.512.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. An analysis of approximations for maximizing submodular set functions—I. Mathematical Programming, 14(1):265–294, December 1978. ISSN 1436-4646. doi: 10.1007/BF01588971.

Shi, W., Cui, Y., Wu, Y., Fang, J., Zhang, S., Li, M., Han, S., Zhu, J., Xu, J., and Zhou, X. Semantic-guided Diverse Decoding for Large Language Model. In The Thirty-ninth Annual Conference on Neural Information Processing Systems, October 2025.

Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: Language agents with verbal reinforcement learning. Advances in Neural Information Processing Systems, 36:8634–8652, December 2023.

Snell, C. V., Lee, J., Xu, K., and Kumar, A. Scaling LLM Test-Time Compute Optimally Can be More Effective than Scaling Parameters for Reasoning. In The Thirteenth International Conference on Learning Representations, October 2024.

Song, W., Dingliwal, S., Jayanthi, S. M., Ganesh, B., Shin, J., Galstyan, A., and Bodapati, S. B. Accelerated test-time scaling with model-free speculative sampling. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pp. 30623–30636, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1558.

Team, Q. Qwen3.5: Accelerating Productivity with Native Multimodal Agents. https://qwen.ai/blog?id=qwen3.5, February 2026.

Tu, S., Li, Y., Bai, Y., Hou, L., and Li, J. DeepPrune: Parallel Scaling without Inter-trace Redundancy, April 2026.

Vijayakumar, A. K., Cogswell, M., Selvaraju, R. R., Sun, Q., Lee, S., Crandall, D., and Batra, D. Diverse Beam Search: Decoding Diverse Solutions from Neural Sequence Models, October 2018.

Wang, A., Song, L., Tian, Y., Yu, D., Mi, H., Duan, X., Tu, Z., Su, J., and Yu, D. Don't get lost in the trees: Streamlining LLM reasoning by overcoming tree search exploration pitfalls. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 23946–23959, Vienna, Austria, July 2025a. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1167.

Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In The Eleventh International Conference on Learning Representations, September 2022.

Wang, X., Li, Y., Feng, S., Yuan, P., Zhang, Y., Shi, J., Tan, C., Pan, B., Hu, Y., and Li, K. Every Rollout Counts: Optimal Resource Allocation for Efficient Test-Time Scaling. In The Thirty-ninth Annual Conference on Neural Information Processing Systems, October 2025b.

Wang, X., Shi, J., Feng, S., Yuan, P., Li, Y., Zhang, Y., Tan, C., Zhang, J., Pan, B., Hu, Y., and Li, K. Do Not Waste Your Rollouts: Recycling Search Experience for Efficient Test-Time Scaling, May 2026.

Wang, Y., Ma, X., Zhang, G., Ni, Y., Chandra, A., Guo, S., Ren, W., Arulraj, A., He, X., Jiang, Z., Li, T., Ku, M., Wang, K., Zhuang, A., Fan, R., Yue, X., and Chen, W. MMLU-Pro: A More Robust and Challenging Multi-Task Language Understanding Benchmark. In The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track, November 2024.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q. V., and Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. Advances in Neural Information Processing Systems, 35:24824–24837, December 2022.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. In Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, pp. 11809–11822, Red Hook, NY, USA, December 2023. Curran Associates Inc.

Zhang, K., Zhou, S., Wang, D., Wang, W. Y., and Li, L. Scaling LLM inference efficiently with optimized sample compute allocation. In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pp. 7959–7973, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.404.

Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H., and Wang, Y.-X. Language Agent Tree Search Unifies Reasoning, Acting, and Planning in Language Models. In Proceedings of the 41st International Conference on Machine Learning, pp. 62138–62160. PMLR, July 2024.

A. Orientation, Notation, and Claim Boundaries

The appendices are organized around the same stack as the method: fixed semantic portfolio values, completion-level rollout actions, and the practical SCPS controller built on approximations to those objects. The goal is not only to prove the formal claims, but also to make clear which quantities are exact theorem-level objects and which are implementation or modeling layers.

A.1. Appendix Roadmap

The main text uses semantic coverage to separate three questions that are often conflated: what the completed portfolio supports, which frontier actions are likely to add support, and how a terminal selector later uses the realized portfolio. The appendices keep these layers separate. Table 2 gives the dependency map.

Appendix	Main role	Key dependencies
A	Notation, assumptions, and claim boundaries	Fixes the frozen-round convention used throughout.
B	Semantic portfolio values	Proves diminishing returns for fixed completed-portfolio objectives.
C	Rollout-action planning and redundancy	Lifts portfolio value to completion-level actions and derives residual-threshold redundancy identities.
D	SCPS optimization and implementation layers	Uses the action objective from Appendix C; analyzes plug-in surrogates, finite horizons, costs, and learned heads.
E	Probes, evaluator approximation, and robustness	Uses the portfolio objective from Appendix B; certifies approximation to the evaluator-induced objective.
F	Exact-answer quotient and answer banks	Specializes semantic coverage to utility quotients and approximate banks.
G	Closed-loop, adaptive, and selector boundaries	Separates frozen one-round guarantees from adaptive replanning and terminal selection.
H	Empirical diagnostics and reproducibility details	Uses the object separation from Appendices F to G to report live exact-answer results, ablations, selector diagnostics, and reproducibility details layer by layer.

Table 2. The appendix dependency map keeps theorem-level objects separate from modeling layers. Appendices B to C build the exact theorem stack; later appendices analyze approximation, exact-answer implementations, closed-loop boundaries, and empirical reporting.

A.2. Standing Assumptions and Notation

Throughout the appendices, fix a prompt x, a planning round t, and the current information state I_t. All portfolios are finite unless explicitly stated otherwise. All random variables and sets are assumed measurable. All semantic values are frozen during the planning round unless a later section explicitly introduces an observation or belief-update model. We use

$$\sup \varnothing = 0, \qquad \max \varnothing = 0.$$

For compactness, NMS abbreviates normalized, monotone, and submodular.

Let Y_x be the space of completed artifacts for prompt x. A completed artifact may be a solution trace, answer, proof, hypothesis, mechanism, evidence summary, critique, proposed experiment, or any other completed semantic object returned by a rollout.

At round t, let $(\Theta_x, \Sigma_x, \mu_t)$ be a finite-measure semantic target space. A target $\theta \in \Theta_x$ represents a decision-relevant semantic region: an answer mode, proof route, mechanism family, evidence type, evaluator question, intervention, or other desideratum. The coverage kernel

$$k_t : Y_x \times \Theta_x \to [0, 1]$$

assigns the degree to which artifact y covers target θ. The baseline

$$b_t : \Theta_x \to [0, 1]$$

records coverage already supplied by the prompt, context, known literature, or other non-portfolio support.

For a finite portfolio $S \subseteq Y_x$, define

$$m_{t,S}(\theta) := \max\left\{b_t(\theta), \sup_{y\in S} k_t(y,\theta)\right\}, \quad (8)$$

and

$$F_t^{\text{sem}}(S) := \int_{\Theta_x} [m_{t,S}(\theta) - b_t(\theta)]\, d\mu_t(\theta). \quad (9)$$

A completion-level macro-action e is a bounded rollout, branch expansion, or continuation policy applied to a frontier state. Its output is a random set $Z_e \subseteq Y_x$ of completed artifacts, finite almost surely. For a selected batch B, define

$$U_B := \bigcup_{e\in B} Z_e.$$

Appendix C formalizes the separability condition under which action-level value inherits diminishing returns.

Frozen-round convention

The exact claims condition on I_t and on evaluator-induced coverage profiles fixed for the current planning round. If semantic weights, baselines, quality scores, answer banks, or evaluator prompts are revised after observing new artifacts, the next planning round uses a new fixed objective.

A.3. Layer Taxonomy

Table 3 separates exact mathematical objects from modeling and implementation choices. This separation is central to the paper: semantic coverage is an exact optimization object once the evaluator defines it, but the evaluator itself is a modeling assumption.

Layer	Object	Status	Main risk
Frozen semantic coverage	Portfolio value over latent semantic targets	Exact for fixed bounded coverage profiles	Evaluator misspecification
Completion-level macro-actions	Stochastic rollout actions returning completed artifacts	Exact NMS lift under batch separability	Wrong action abstraction
Residual thresholds	Union-event representation of graded semantic gain	Exact for semantic coverage	None beyond the fixed objective
Product form	Multiplicative miss probabilities over residual thresholds	Conditional on independence	Cross-action dependence
Bonferroni lower bound	Novelty minus overlap expression	Exact with true probabilities	Estimation and calibration error
SCPS surrogate	Estimated novelty, overlap, and cost score	Planning surrogate	Miscalibrated heads or cost model
Semantic probes	Finite approximation to Θ_x	Concentrates for finite planning families	Biased probe distribution
Exact-answer banks	Coarsened representation of utility quotient	Certifiable under singletonization and omitted-mass bounds	Mixed buckets overcredit support
Closed-loop belief updates	Replanning after observations change weights or beliefs	Requires an observation model	Frozen theorem does not automatically apply

Table 3. The layer taxonomy separates exact coverage results from implementation assumptions. Exact results concern fixed coverage objectives and completion-level action lifts; probes, evaluators, banks, costs, and belief updates add modeling assumptions.

A.4. Claims and Non-Claims

Table 4 records the main theorem-level claims and the corresponding non-claims. The negative rows are as important as the positive rows: they prevent the submodularity theorem from being read as a global theorem about all search, discovery, or selection behavior.

Statement	Status
Fixed semantic coverage over latent target spaces is NMS.	Exact theorem; proved in Appendix B.
Quality-gated semantic coverage is NMS once quality scores are fixed for the round.	Exact corollary of the fixed-kernel theorem.
Component coverage, concave evidence accumulation, and nonnegative multi-channel sums are NMS.	Exact theorems; proved in Appendix B.
Completion-level rollout actions inherit NMS one-round gain.	Exact under batch separability; proved in Appendix C.
One-round truth-coupled hit is NMS when potential answer outputs are fixed for the round.	Exact union-probability coverage result; proved in Appendix F.
Additive trace scoring overestimates semantic support by expected excess residual-threshold multiplicity.	Exact identity; proved in Appendix C.
The pairwise novelty–overlap score is lower-bound-derived.	Exact with true residual-hit probabilities; estimation and cost regularization are separate.
Primitive token-level search is globally submodular.	False in general; counterexample in Appendix C.
Arbitrary scientific discovery utility is submodular.	False in general; strict complementarity counterexample in Appendix B.
SCPS optimizes true human or scientific value.	Not claimed; the theorem is internal to the evaluator-induced objective.
The deployed plug-in SCPS score is always an exact lower bound on semantic gain.	Not claimed; estimated terms, tuned coefficients, and costs make it a planning surrogate.
Belief-updating MPC inherits the frozen theorem automatically.	Not claimed; adaptive guarantees require an explicit observation model.
Selected accuracy is always bounded by frozen answer coverage.	False without additional assumptions; exact-answer objects are separated in Appendix F.

Table 4. The claims ledger bounds the exact theory to fixed semantic coverage objectives and completion-level action lifts. Positive rows state theorem-level claims; negative rows mark search, discovery, adaptation, and selector claims that require separate assumptions or evidence.

Scope of the fixed-objective results

The paper does not claim that semantic evaluators are perfect, that arbitrary discovery utility is submodular, that token expansions are the right submodular ground set, or that terminal selectors will use discovered support correctly. The exact theory certifies the fixed coverage objective and its completion-level action lift.

A.5. Notation Index

Symbol	Meaning
x	Prompt or task instance.
t	Planning round.
I_t	Current information state.
Y_x	Space of completed artifacts.
S_t	Current completed portfolio.
Θ_x	Latent semantic target space.
μ_t	Current importance measure over semantic targets.
$k_t(y, \theta)$	Degree to which artifact y covers target θ.
$b_t(\theta)$	Baseline coverage of target θ before portfolio additions.
$m_{t,S}(\theta)$	Best coverage of target θ supplied by baseline or portfolio S.
F_t^{sem}	Semantic portfolio value.
E_t	Set of completion-level macro-actions available at round t.
Z_e	Random finite set of completed artifacts returned by action e.
U_B	Union of outputs returned by actions in batch B.
$G_t(B \mid S_t)$	Expected one-round gain from selecting action batch B.
$H_e(\theta, r)$	Event that action e covers residual threshold (θ, r).
$v_t(e \mid S_t)$	Expected residual support covered by action e alone.
$\omega_t(e, f \mid S_t)$	Expected residual support covered redundantly by actions e and f.
$A_t^{\text{add}}(B \mid S_t)$	Additive trace score for action batch B.

Table 5. Core notation used in Appendices A to C. Symbols are ordered by dependency: task state, semantic coverage objects, completion-level actions, and residual-support quantities.

B. Semantic Portfolio Values

This appendix proves the completed-portfolio value results. The common theme is that a portfolio has diminishing returns when value is driven by coverage of support: a new artifact helps to the extent that it covers targets, components, evidence, or channels not already well covered by the current portfolio.

B.1. Latent Semantic Coverage

Theorem 1 (latent semantic coverage, restated). Suppose b_t and $k_t(y, \cdot)$ are measurable, $0 \le b_t(\theta) \le 1$, and $0 \le k_t(y, \theta) \le 1$. Then F_t^{sem} from Equation (9) is normalized, monotone, and submodular over finite completed portfolios.

Proof. For each fixed target $\theta \in \Theta_x$, define

$$f_\theta(S) := \max\left\{b_t(\theta), \sup_{y\in S} k_t(y, \theta)\right\} - b_t(\theta).$$

Then

$$F_t^{\text{sem}}(S) = \int_{\Theta_x} f_\theta(S)\, d\mu_t(\theta).$$

Normalization holds because

$$f_\theta(\varnothing) = \max\{b_t(\theta), 0\} - b_t(\theta) = 0,$$

since $b_t(\theta) \ge 0$.

For monotonicity, let $S \subseteq T$. Then

$$\sup_{y\in S} k_t(y, \theta) \le \sup_{y\in T} k_t(y, \theta),$$

so $f_\theta(S) \le f_\theta(T)$ for every θ. Integrating gives

$$F_t^{\text{sem}}(S) \le F_t^{\text{sem}}(T).$$

For submodularity, fix finite $S \subseteq T \subseteq Y_x$ and artifact $y \notin T$. Let

$$m_S(\theta) := \max\left\{b_t(\theta), \sup_{z\in S} k_t(z, \theta)\right\}, \qquad m_T(\theta) := \max\left\{b_t(\theta), \sup_{z\in T} k_t(z, \theta)\right\}.$$

Since $S \subseteq T$, $m_S(\theta) \le m_T(\theta)$ pointwise. The marginal gain at target θ from adding y to S is

$$f_\theta(S \cup \{y\}) - f_\theta(S) = [k_t(y, \theta) - m_S(\theta)]_+,$$

and the marginal at T is

$$f_\theta(T \cup \{y\}) - f_\theta(T) = [k_t(y, \theta) - m_T(\theta)]_+.$$

Because $m_S(\theta) \le m_T(\theta)$,

$$[k_t(y, \theta) - m_S(\theta)]_+ \ge [k_t(y, \theta) - m_T(\theta)]_+.$$

Integrating the pointwise inequality yields

$$F_t^{\text{sem}}(S \cup \{y\}) - F_t^{\text{sem}}(S) \ge F_t^{\text{sem}}(T \cup \{y\}) - F_t^{\text{sem}}(T),$$

which is the diminishing-returns form of submodularity. □

B.2. Quality-Gated Coverage

Open-ended novelty should not be rewarded unless it is also plausible, admissible, or relevant. The theorem above does not require any particular construction of the kernel. It applies to any fixed bounded coverage profile.

Corollary B.1 (fixed quality-gated kernels). Let $\widetilde{k}_t : Y_x \times \Theta_x \to [0, 1]$ be any measurable kernel fixed for the planning round. The value

$$\widetilde{F}_t(S) := \int_{\Theta_x} \left[\max\left\{b_t(\theta), \sup_{y\in S} \widetilde{k}_t(y, \theta)\right\} - b_t(\theta)\right] d\mu_t(\theta)$$

is NMS over finite completed portfolios.

In particular, if $q_t(y) \in [0, 1]$ scores plausibility, validity, or admissibility and $r_t(y, \theta) \in [0, 1]$ scores relevance, then the product kernel

$$\widetilde{k}_t(y, \theta) = q_t(y) r_t(y, \theta)$$

induces an NMS quality-gated value.

Proof. Apply Theorem 1 with $\widetilde{k}_t$ in place of k_t. The product form is a special case because $q_t(y) r_t(y, \theta) \in [0, 1]$ is a fixed bounded measurable coverage profile during the frozen round. □

B.3. Local Redundancy and Novelty Bounds

The next two propositions are the local version of the paper's central intuition. A fluent duplicate adds little if it does not improve coverage beyond what the portfolio already contains. Conversely, a single artifact can be valuable if it covers an important region that remains undercovered.

Proposition B.2 (near-duplicate bound). Let S be a completed portfolio and suppose $z \in S$. If

$$\int_{\Theta_x} [k_t(y, \theta) - k_t(z, \theta)]_+ \, d\mu_t(\theta) \le \varepsilon,$$

then

$$F_t^{\text{sem}}(S \cup \{y\}) - F_t^{\text{sem}}(S) \le \varepsilon.$$

Proof. Since $z \in S$,

$$m_{t,S}(\theta) = \max\left\{b_t(\theta), \sup_{w\in S} k_t(w, \theta)\right\} \ge k_t(z, \theta)$$

for every θ. The marginal gain of adding y is

$$F_t^{\text{sem}}(S \cup \{y\}) - F_t^{\text{sem}}(S) = \int_{\Theta_x} [k_t(y, \theta) - m_{t,S}(\theta)]_+ \, d\mu_t(\theta).$$

Because $m_{t,S}(\theta) \geq k_t(z, \theta)$,

$$[k_t(y,\theta) - m_{t,S}(\theta)]_+ \leq [k_t(y,\theta) - k_t(z,\theta)]_+.$$

Integrating gives the claim. □

Proposition B.3 (novel-region lower bound). Let S be a completed portfolio. Suppose there is a measurable region $R \subseteq \Theta_x$ such that

$$k_t(y,\theta) \geq m_{t,S}(\theta) + \gamma \qquad \text{for all } \theta \in R,$$

where $\gamma > 0$ and $\mu_t(R) = \delta$. Then

$$F_t^{\text{sem}}(S \cup \{y\}) - F_t^{\text{sem}}(S) \geq \gamma\delta.$$

Proof. The marginal gain of y is

$$\int_{\Theta_x} [k_t(y,\theta) - m_{t,S}(\theta)]_+ \, d\mu_t(\theta).$$

Restricting the integral to R gives

$$F_t^{\text{sem}}(S \cup \{y\}) - F_t^{\text{sem}}(S) \geq \int_R [k_t(y,\theta) - m_{t,S}(\theta)]_+ \, d\mu_t(\theta).$$

By assumption the integrand is at least γ on R, so the marginal is at least

$$\int_R \gamma \, d\mu_t(\theta) = \gamma \mu_t(R) = \gamma\delta.$$

□

B.4. Ideal-Selector Envelope

Semantic coverage also contains the ideal envelope of a selector that may choose the best item in the realized portfolio after the latent task condition is known. This result is not a claim that practical selectors achieve this envelope; it only identifies the envelope as another max-coverage objective.

Corollary B.4 (selector envelope). Let $T^\star$ be a latent task condition with conditional law $\mathcal{L}(T^\star \mid I_t)$, and let $u(y,\tau) \in [0,1]$ be the utility of completed artifact y under condition τ. Define

$$F_{\text{sel}}(S) := \mathbb{E}\left[\max_{y \in S} u(y, T^\star) \mid I_t\right], \qquad \max \varnothing := 0.$$

Then F_{sel} is a latent semantic coverage objective and is NMS.

Proof. Take

$$\Theta_x = \mathcal{T}, \qquad \mu_t = \mathcal{L}(T^\star \mid I_t), \qquad b_t(\tau) = 0,$$

and set

$$k_t(y,\tau) = u(y,\tau).$$

Then

$$F_t^{\text{sem}}(S) = \int_{\mathcal{T}} \sup_{y \in S} u(y,\tau) \, d\mu_t(\tau) = \mathbb{E}\left[\max_{y \in S} u(y, T^\star) \mid I_t\right].$$

The result follows from Theorem 1. □

B.5. Component Coverage

In discovery-style settings, the support exposed by an artifact may be more naturally represented by components: mechanisms, proof lemmas, evidence types, interventions, critique categories, or experimental handles. Component coverage values the union of components exposed by the portfolio.

Theorem B.5 (component coverage). Let $\mathcal{C}_x$ be a finite component universe. Let each completed artifact y expose a set $C(y) \subseteq \mathcal{C}_x$, and let $g_t : 2^{\mathcal{C}_x} \to \mathbb{R}_+$ be NMS. Define

$$F_t^{\text{comp}}(S) := g_t\left(\bigcup_{y \in S} C(y)\right).$$

Then F_t^{comp} is NMS over finite portfolios. More generally, if $g_{t,\tau}$ is NMS for each latent condition τ, $T^\star$ is distributed according to I_t, and $\tau \mapsto g_{t,\tau}(U)$ is measurable and integrable for every $U \subseteq \mathcal{C}_x$, then

$$F_t^{\text{comp}}(S) := \mathbb{E}\left[g_{t,T^\star}\left(\bigcup_{y \in S} C(y)\right) \mid I_t\right]$$

is also NMS.

Proof. First consider deterministic g_t. Let

$$U_S := \bigcup_{y \in S} C(y).$$

Normalization holds because $U_\varnothing = \varnothing$ and $g_t(\varnothing) = 0$. If $S \subseteq T$, then $U_S \subseteq U_T$, so monotonicity of g_t gives

$$F_t^{\text{comp}}(S) \leq F_t^{\text{comp}}(T).$$

For submodularity, fix $S \subseteq T$ and $y \notin T$. Since $U_S \subseteq U_T$, submodularity of g_t gives

$$g_t(U_S \cup C(y)) - g_t(U_S) \geq g_t(U_T \cup C(y)) - g_t(U_T).$$

This is exactly

$$F_t^{\text{comp}}(S \cup \{y\}) - F_t^{\text{comp}}(S) \geq F_t^{\text{comp}}(T \cup \{y\}) - F_t^{\text{comp}}(T).$$

For the latent-condition version, the same argument holds pointwise for each realization $T^\star = \tau$. Conditional expectation preserves normalization, monotonicity, and submodularity. □

B.6. Concave Evidence Accumulation

Some support should accumulate, but with diminishing returns. A first credible piece of evidence for a mechanism can matter a great deal; repeated near-duplicates should matter less. Concave evidence accumulation captures this behavior.

Let $\mathcal{C}_x$ be a finite component universe. Each artifact y contributes nonnegative evidence intensity $m_c(y) \geq 0$ to component c. For a portfolio S, define

$$M_c(S) := \sum_{y \in S} m_c(y).$$

Let $w_t(c) \geq 0$, and let $\rho_c : \mathbb{R}_+ \to \mathbb{R}_+$ be normalized, nondecreasing, and concave. Define

$$F_t^{\mathrm{acc}}(S) := \sum_{c \in \mathcal{C}_x} w_t(c) \rho_c(M_c(S)). \tag{10}$$

Lemma B.6 (concave increments decrease). If $\rho : \mathbb{R}_+ \to \mathbb{R}_+$ is concave and nondecreasing, then for every $r \geq 0$, the function

$$x \mapsto \rho(x + r) - \rho(x)$$

is nonincreasing on $\mathbb{R}_+$.

Proof. The case $r = 0$ is trivial. Fix $r > 0$ and $0 \leq x \leq y$. If $x = y$, the claim is equality. Otherwise let $d := y - x > 0$. By concavity, applied to the representations

$$y = \frac{r}{d+r} x + \frac{d}{d+r}(y+r), \qquad x+r = \frac{d}{d+r} x + \frac{r}{d+r}(y+r),$$

we have

$$\rho(y) \geq \frac{r}{d+r}\rho(x) + \frac{d}{d+r}\rho(y+r),$$

and

$$\rho(x+r) \geq \frac{d}{d+r}\rho(x) + \frac{r}{d+r}\rho(y+r).$$

Adding these inequalities gives

$$\rho(x+r) + \rho(y) \geq \rho(x) + \rho(y+r),$$

which is equivalent to

$$\rho(x+r) - \rho(x) \geq \rho(y+r) - \rho(y).$$

□

Theorem B.7 (concave evidence accumulation). The value F_t^{acc} from Equation (10) is NMS.

Proof. Normalization follows because $M_c(\varnothing) = 0$ and $\rho_c(0) = 0$ for every c.

For monotonicity, if $S \subseteq T$, then $M_c(S) \leq M_c(T)$ for every component c. Since each ρ_c is nondecreasing and each $w_t(c)$ is nonnegative,

$$F_t^{\mathrm{acc}}(S) \leq F_t^{\mathrm{acc}}(T).$$

For submodularity, fix $S \subseteq T$ and $y \notin T$. For each component c, let

$$a_c := M_c(S), \qquad b_c := M_c(T), \qquad r_c := m_c(y).$$

Then $a_c \leq b_c$ and $r_c \geq 0$. By the concave-increment lemma in Appendix B.6,

$$\rho_c(a_c + r_c) - \rho_c(a_c) \geq \rho_c(b_c + r_c) - \rho_c(b_c).$$

Multiplying by $w_t(c) \geq 0$ and summing over c yields

$$F_t^{\mathrm{acc}}(S \cup \{y\}) - F_t^{\mathrm{acc}}(S) \geq F_t^{\mathrm{acc}}(T \cup \{y\}) - F_t^{\mathrm{acc}}(T).$$

□

B.7. Multi-Channel Values

A portfolio may contain different artifact types: hypotheses, evidence summaries, critiques, experiments, code patches, or proof attempts. Nonnegative sums of channel-specific diminishing-return values remain diminishing-return values.

Proposition B.8 (multi-channel closure). Let

$$\mathcal{Y} = \bigsqcup_{j=1}^{m} \mathcal{Y}^{(j)}$$

be a disjoint union of artifact channels, and let $S^{(j)} := S \cap \mathcal{Y}^{(j)}$. Suppose each

$$F_t^{(j)} : 2^{\mathcal{Y}^{(j)}} \to \mathbb{R}_+$$

is NMS. For $\lambda_j \geq 0$, define

$$F_t^{\mathrm{multi}}(S) := \sum_{j=1}^{m} \lambda_j F_t^{(j)}(S^{(j)}).$$

Then F_t^{multi} is NMS.

Proof. Normalization follows from normalization of all channel values. Monotonicity follows because $S \subseteq T$ implies $S^{(j)} \subseteq T^{(j)}$ for every j, and each channel value is monotone with nonnegative weight.

For submodularity, fix $S \subseteq T$ and $y \notin T$. Suppose $y \in \mathcal{Y}^{(i)}$. Only channel i changes when y is added, so

$$F_t^{\mathrm{multi}}(S \cup \{y\}) - F_t^{\mathrm{multi}}(S) = \lambda_i \left[F_t^{(i)}(S^{(i)} \cup \{y\}) - F_t^{(i)}(S^{(i)}) \right],$$

and similarly for T. Since $S^{(i)} \subseteq T^{(i)}$ and $F_t^{(i)}$ is submodular,

$$F_t^{(i)}(S^{(i)} \cup \{y\}) - F_t^{(i)}(S^{(i)}) \geq F_t^{(i)}(T^{(i)} \cup \{y\}) - F_t^{(i)}(T^{(i)}).$$

Multiplying by $\lambda_i \geq 0$ proves the diminishing-returns inequality for F_t^{multi}. □

B.8. Complementarity Boundary and Composite Artifacts

Coverage is broad, but it is not universal. Some discovery utilities contain strict complementarities: an artifact may be valuable only after another artifact is present.

Proposition B.9 (strict complementarity can violate submodularity). Arbitrary open-ended discovery utility need not be submodular.

Proof. Let the portfolio contain two possible artifacts y_1, y_2. Interpret y_1 as a proposed mechanism and y_2 as an experiment that tests exactly that mechanism. Suppose neither artifact is valuable alone, but together they are valuable:

$$U(\varnothing) = 0, \qquad U(\{y_1\}) = 0, \qquad U(\{y_2\}) = 0, \qquad U(\{y_1, y_2\}) = 1.$$

The marginal gain of adding y_2 to the empty set is

$$U(\{y_2\}) - U(\varnothing) = 0,$$

while the marginal gain of adding y_2 after y_1 is present is

$$U(\{y_1, y_2\}) - U(\{y_1\}) = 1.$$

The marginal increases as the portfolio grows, violating diminishing returns. □

Composite-artifact modeling note. The counterexample is a boundary for the original atomic ground set. It does not forbid modeling predictable local complementarities as completed composite artifacts. For example, an artifact space might include pairs of the form

$$y = (\text{mechanism}, \text{discriminating experiment}),$$

with a fixed coverage profile $k_t(y, \theta)$. Semantic coverage is then submodular over portfolios of these composite artifacts by Theorem 1. This changes the ground set; it does not prove that the original atomic mechanism-plus-experiment utility is submodular.

Coverage is broad, not universal

Semantic coverage, component coverage, concave evidence accumulation, and nonnegative channel sums all have diminishing returns. Arbitrary discovery utilities with strict complementarity do not. When complementarities are predictable, one modeling response is to lift the completed artifact space so that the complementary bundle is the object being covered.

C. Rollout-Action Planning and Semantic Redundancy

The previous appendix proves that completed portfolios can have diminishing returns. A test-time controller, however, chooses actions before knowing which artifacts they will complete. This appendix proves the action-level lift, derives the residual-threshold representation used by SCPS, and makes trace redundancy exact.

C.1. Completion-Level Macro-Actions and Batch Separability

At round t, let E_t be a finite set of completion-level macro-actions. Each action $e \in E_t$ has a potential output

$$Z_e \subseteq Y_x,$$

a random set of completed artifacts, finite almost surely. For a selected batch $B \subseteq E_t$, the realized completed artifacts are

$$U_B := \bigcup_{e \in B} Z_e.$$

Batch separability. We assume the potential output Z_e of each action is attached to the action and is not physically changed by which other actions are selected in the same batch. Equivalently, there is a joint probability space carrying $(Z_e)_{e \in E_t}$, and selecting batch B reveals or executes the corresponding subset and returns $\bigcup_{e \in B} Z_e$. The random sets Z_e may be statistically dependent; separability does not require independence.

This condition holds, for example, when each selected macro-action has its own prefix, decoding policy, token cap, and randomness, and executing one action does not alter the prompt, memory, tool state, or resource allocation of another action in the same batch. It can fail when actions interact through a mutable environment, when one action changes another action's prompt or available budget, or when one action unlocks another action's output. Appendix C.9 gives a minimal failure example.

Given a fixed portfolio value F_t and current completed portfolio S_t, define the one-round action gain

$$G_t(B \mid S_t) := \mathbb{E}\left[F_t(S_t \cup U_B) - F_t(S_t) \mid I_t\right]. \tag{11}$$

C.2. A Correct Set-Marginal Lemma

The action lift needs the fact that a monotone submodular function has diminishing returns not only for single elements but also for finite sets. Monotonicity is necessary in the statement below.

Lemma C.1 (set diminishing returns). Let $F : 2^{\mathcal{Y}} \to \mathbb{R}$ be monotone and submodular. If $A \subseteq B \subseteq \mathcal{Y}$ and $C \subseteq \mathcal{Y}$ is finite, then

$$F(A \cup C) - F(A) \geq F(B \cup C) - F(B).$$

Proof. Let

$$D := C \setminus B.$$

Because F is monotone and $A \cup D \subseteq A \cup C$,

$$F(A \cup C) - F(A) \geq F(A \cup D) - F(A).$$

Enumerate $D = \{d_1, \ldots, d_m\}$, and define

$$A_i := A \cup \{d_1, \ldots, d_{i-1}\}, \qquad B_i := B \cup \{d_1, \ldots, d_{i-1}\}.$$

Then $A_i \subseteq B_i$. By submodularity,

$$F(A_i \cup \{d_i\}) - F(A_i) \geq F(B_i \cup \{d_i\}) - F(B_i).$$

Summing over i telescopes to

$$F(A \cup D) - F(A) \geq F(B \cup D) - F(B).$$

Since $B \cup D = B \cup C$, we obtain

$$F(A \cup C) - F(A) \geq F(A \cup D) - F(A) \geq F(B \cup C) - F(B).$$

□

Why monotonicity is needed. The lemma is false for arbitrary nonmonotone submodular functions. Let $F(S) = -|S|$, which is modular and therefore submodular but not monotone. With $A = \varnothing$, $B = \{b\}$, and $C = \{b\}$,

$$F(A \cup C) - F(A) = -1, \qquad F(B \cup C) - F(B) = 0,$$

so the claimed inequality would fail.

C.3. Macro-Action Closure

Theorem C.2 (macro-action closure). Under batch-separable macro-actions, if F_t is NMS over completed portfolios and $F_t(S_t \cup U_B)$ is integrable for every $B \subseteq E_t$, then $G_t(\cdot \mid S_t)$ from Equation (11) is NMS over batches $B \subseteq E_t$.

Proof. Fix a realization ω of all potential action outputs $(Z_e)_{e \in E_t}$. Define the realized batch value

$$g_\omega(B) := F_t\left(S_t \cup \bigcup_{e \in B} Z_e(\omega)\right) - F_t(S_t).$$

Normalization holds because $U_\varnothing = \varnothing$, so

$$g_\omega(\varnothing) = F_t(S_t) - F_t(S_t) = 0.$$

For monotonicity, if $A \subseteq B$, then

$$S_t \cup \bigcup_{e \in A} Z_e(\omega) \subseteq S_t \cup \bigcup_{e \in B} Z_e(\omega).$$

Monotonicity of F_t gives $g_\omega(A) \le g_\omega(B)$.

For submodularity, fix $A \subseteq B \subseteq E_t$ and $e \notin B$. Let

$$X_A := S_t \cup \bigcup_{f \in A} Z_f(\omega), \qquad X_B := S_t \cup \bigcup_{f \in B} Z_f(\omega).$$

Then $X_A \subseteq X_B$. The marginal of adding action e to A is

$$F_t(X_A \cup Z_e(\omega)) - F_t(X_A),$$

and the marginal of adding e to B is

$$F_t(X_B \cup Z_e(\omega)) - F_t(X_B).$$

By the set-marginal lemma in Appendix C.2, with $F = F_t$, $A = X_A$, $B = X_B$, and $C = Z_e(\omega)$, the first marginal is at least the second. Thus g_ω is submodular for every realization ω.

Taking conditional expectation over ω preserves normalization, monotonicity, and submodularity, so $G_t(\cdot \mid S_t)$ is NMS. □

C.4. Residual Thresholds

The residual-threshold representation turns graded semantic coverage into ordinary coverage over thresholds. Covering target θ to degree 0.7 is equivalent to covering every threshold (θ, r) with $r \le 0.7$.

Lemma C.3 (threshold representation). For semantic coverage,

$$F_t^{\mathrm{sem}}(S) = \int_{\Theta_x} \int_0^1 \mathbf{1}\{b_t(\theta) < r \le m_{t,S}(\theta)\}\, dr\, d\mu_t(\theta). \tag{12}$$

Proof. Fix θ. Since $m_{t,S}(\theta) \ge b_t(\theta)$ and both are in $[0,1]$,

$$m_{t,S}(\theta) - b_t(\theta) = \int_0^1 \mathbf{1}\{b_t(\theta) < r \le m_{t,S}(\theta)\}\, dr.$$

Integrating over Θ_x gives Equation (12). □

For current portfolio S_t, define the residual threshold event for action e:

$$H_e(\theta, r) := \left\{ r > m_{t,S_t}(\theta) \text{ and } \sup_{y \in Z_e} k_t(y, \theta) \ge r \right\}. \tag{13}$$

This event says that action e covers threshold (θ, r) and that the threshold was not already covered by the current portfolio.

C.5. Residual-Threshold Lift

Theorem C.4 (residual-threshold lift). For semantic coverage, the one-round gain of a batch B is

$$G_t^{\mathrm{sem}}(B \mid S_t) = \int_{\Theta_x} \int_0^1 \Pr\left(\bigcup_{e \in B} H_e(\theta, r) \mid I_t \right) dr\, d\mu_t(\theta). \tag{14}$$

Proof. By definition,

$$G_t^{\mathrm{sem}}(B \mid S_t) = \mathbb{E}\left[F_t^{\mathrm{sem}}(S_t \cup U_B) - F_t^{\mathrm{sem}}(S_t) \mid I_t \right].$$

Using the threshold-representation lemma in Appendix C.4, the realized marginal gain for fixed U_B is

$$\int_{\Theta_x} \int_0^1 \mathbf{1}\{m_{t,S_t}(\theta) < r \le m_{t,S_t \cup U_B}(\theta)\}\, dr\, d\mu_t(\theta).$$

The event inside the indicator holds exactly when threshold (θ, r) is not covered by S_t but is covered by at least one artifact returned by a selected action. Since $U_B = \bigcup_{e \in B} Z_e$, this is equivalent to

$$\bigcup_{e \in B} H_e(\theta, r).$$

Taking conditional expectation and applying Tonelli's theorem to the nonnegative bounded integrand gives Equation (14). □

C.6. Semantic Redundancy Gap

The residual-threshold lift makes precise what goes wrong when rollouts are treated as additive independent candidates. Additive scoring counts how many times residual support thresholds are hit. Portfolio value counts whether each residual threshold is hit at least once.

For $u = (\theta, r)$, write $H_e(u) := H_e(\theta, r)$, and let $d\nu_t(u) := dr\, d\mu_t(\theta)$ on $\Theta_x \times [0, 1]$. Define the residual hit multiplicity

$$N_B(u) := \sum_{e \in B} \mathbf{1}\{H_e(u)\}.$$

Define the singleton-additive trace value

$$A_t^{\mathrm{add}}(B \mid S_t) := \sum_{e \in B} v_t(e \mid S_t),$$

where

$$v_t(e \mid S_t) := \int_{\Theta_x} \int_0^1 \Pr(H_e(\theta, r) \mid I_t)\, dr\, d\mu_t(\theta).$$

Main overcount proposition, restated. Proposition 3 is restated below with the same additive trace score notation A_t^{add} used in the main text.

Proposition C.5 (trace-additive overcount identity). For every finite batch B,

$$A_t^{\mathrm{add}}(B \mid S_t) - G_t^{\mathrm{sem}}(B \mid S_t) = \int_{\Theta_x} \int_0^1 \mathbb{E}\left[(N_B(\theta, r) - 1)_+ \mid I_t\right] dr\, d\mu_t(\theta). \tag{15}$$

Proof. By finite additivity and Tonelli's theorem,

$$\begin{aligned} A_t^{\mathrm{add}}(B \mid S_t) &= \sum_{e \in B} \int \Pr(H_e(u) \mid I_t)\, d\nu_t(u) \\ &= \int \sum_{e \in B} \Pr(H_e(u) \mid I_t)\, d\nu_t(u) \\ &= \int \mathbb{E}\left[\sum_{e \in B} \mathbf{1}\{H_e(u)\} \mid I_t\right] d\nu_t(u) \\ &= \int \mathbb{E}[N_B(u) \mid I_t] d\nu_t(u). \end{aligned}$$

By the residual-threshold lift in Appendix C.5,

$$G_t^{\mathrm{sem}}(B \mid S_t) = \int \Pr(N_B(u) \geq 1 \mid I_t)\, d\nu_t(u) = \int \mathbb{E}[\mathbf{1}\{N_B(u) \geq 1\} \mid I_t] d\nu_t(u).$$

Subtracting gives

$$A_t^{\mathrm{add}}(B \mid S_t) - G_t^{\mathrm{sem}}(B \mid S_t) = \int \mathbb{E}\left[N_B(u) - \mathbf{1}\{N_B(u) \geq 1\} \mid I_t\right] d\nu_t(u).$$

Since $N_B(u)$ is a nonnegative integer,

$$N_B(u) - \mathbf{1}\{N_B(u) \geq 1\} = (N_B(u) - 1)_+.$$

This proves Equation (15). □

> **Trace count versus support**
>
> Additive trace scoring counts residual-threshold multiplicity. Semantic portfolio value counts whether each residual threshold is hit at least once. The exact overcount is the expected excess multiplicity in Equation (15).

Corollary C.6 (pairwise overlap upper-bounds overcount). For every finite batch B,

$$A_t^{\mathrm{add}}(B \mid S_t) - G_t^{\mathrm{sem}}(B \mid S_t) \leq \sum_{\{e,f\} \subseteq B} \omega_t(e, f \mid S_t),$$

where

$$\omega_t(e, f \mid S_t) := \int_{\Theta_x} \int_0^1 \Pr(H_e(\theta, r) \cap H_f(\theta, r) \mid I_t)\, dr\, d\mu_t(\theta).$$

Proof. For any nonnegative integer n,

$$(n-1)_+ \leq \binom{n}{2}.$$

The inequality is equality for $n = 0, 1, 2$. Therefore,

$$(N_B(u) - 1)_+ \leq \binom{N_B(u)}{2} = \sum_{\{e,f\} \subseteq B} \mathbf{1}\{H_e(u) \cap H_f(u)\}.$$

Taking conditional expectation and integrating gives the result. □

C.7. Product Form and Bonferroni Bounds

The macro-action closure theorem and redundancy identity do not require independence. Independence is only needed for a product form. Without independence, Bonferroni inequalities give lower and upper bounds in terms of residual overlaps.

Corollary C.7 (product form under residual-hit independence). If, for each (θ, r), the events $\{H_e(\theta, r)\}_{e \in B}$ are conditionally independent given I_t, then

$$G_t^{\mathrm{sem}}(B \mid S_t) = \int_{\Theta_x} \int_0^1 \left[1 - \prod_{e \in B} (1 - p_t(e; \theta, r))\right] dr\, d\mu_t(\theta),$$

where

$$p_t(e; \theta, r) := \Pr(H_e(\theta, r) \mid I_t).$$

Proof. For fixed (θ, r), conditional independence gives

$$\Pr\left(\bigcap_{e \in B} H_e(\theta, r)^c \mid I_t\right) = \prod_{e \in B} (1 - p_t(e; \theta, r)).$$

Taking complements and substituting into the residual-threshold lift in Appendix C.5 proves the formula. □

Proposition C.8 (pairwise Bonferroni lower bound). Every finite batch B satisfies

$$G_t^{\mathrm{sem}}(B \mid S_t) \geq \sum_{e \in B} v_t(e \mid S_t) - \sum_{\{e,f\} \subseteq B} \omega_t(e, f \mid S_t). \tag{16}$$

Proof. For fixed (θ, r), the second Bonferroni inequality gives

$$\Pr\left(\bigcup_{e \in B} H_e(\theta, r) \mid I_t\right) \geq \sum_{e \in B} \Pr(H_e(\theta, r) \mid I_t) - \sum_{\{e,f\} \subseteq B} \Pr(H_e(\theta, r) \cap H_f$$

Integrating over $r \in [0, 1]$ and $\theta \in \Theta_x$ gives Equation (16). □

Proposition C.9 (higher-order Bonferroni bounds). For $j = 1, \dots, |B|$, define

$$I_j(B) := \sum_{\substack{J \subseteq B \\ |J|=j}} \int_{\Theta_x} \int_0^1 \Pr\left(\bigcap_{e \in J} H_e(\theta, r) \mid I_t \right) dr\, d\mu_t(\theta).$$

For any $K \leq |B|$,

$$S_K(B) := \sum_{j=1}^{K} (-1)^{j+1} I_j(B).$$

If K is even, then

$$S_K(B) \leq G_t^{\text{sem}}(B \mid S_t).$$

If K is odd, then

$$S_K(B) \geq G_t^{\text{sem}}(B \mid S_t).$$

When $K = |B|$, equality holds.

Proof. Fix a residual threshold $u = (\theta, r)$ and let

$$N = N_B(u) = \sum_{e \in B} \mathbf{1}\{H_e(u)\}.$$

For this fixed u,

$$\mathbf{1}\left\{ \bigcup_{e \in B} H_e(u) \right\} = \mathbf{1}\{N \geq 1\}.$$

The order-K inclusion–exclusion partial sum at this u is

$$s_K(N) := \sum_{j=1}^{K} (-1)^{j+1} \binom{N}{j},$$

where $\binom{N}{j} = 0$ for $j > N$. If $N = 0$, then $s_K(N) = 0 = \mathbf{1}\{N \geq 1\}$. If $N \geq 1$, the binomial identity

$$\sum_{j=0}^{K} (-1)^j \binom{N}{j} = (-1)^K \binom{N-1}{K}$$

for $K < N$ and equality to 0 for $K \geq N$ implies

$$s_K(N) = 1 - (-1)^K \binom{N-1}{K}$$

when $K < N$, and $s_K(N) = 1$ when $K \geq N$. Thus even K gives $s_K(N) \leq \mathbf{1}\{N \geq 1\}$, and odd K gives $s_K(N) \geq \mathbf{1}\{N \geq 1\}$.

Taking conditional expectations and integrating over u gives the stated inequalities. When $K = |B|$, the full inclusion–exclusion identity is exact. □

C.8. Token-Level Pullbacks Need Not Be Submodular

The action ground set matters. Even if value is submodular over completed artifacts, pulling it back to primitive token-expansion edges can create complementarities.

Boundary. Diminishing returns hold for completed portfolios and for batch-separable completion-level macro-actions. They need not hold for primitive token edges or for interacting actions whose outputs change when selected together.

Proposition C.10 (token pullback failure). Semantic value may be submodular over completed artifacts while its pullback to primitive token-level expansion edges is not submodular.

Proof. Let E^{tok} be primitive token-level expansion edges in a rooted search tree. For $X \subseteq E^{\text{tok}}$, let $D(X)$ be the set of completed leaves whose full root-to-leaf edge set is contained in X.

Consider a tree with exactly one valuable completed leaf $z^\star$, and suppose the path to $z^\star$ requires two prerequisite token edges e_1, e_2. Define the completed-artifact value

$$F(S) = \mathbf{1}\{z^\star \in S\}.$$

This F is monotone and submodular over completed artifacts. Define the token pullback

$$P(X) := F(D(X)).$$

Then

$$P(\varnothing) = 0, \qquad P(\{e_1\}) = 0, \qquad P(\{e_2\}) = 0, \qquad P(\{e_1, e_2\}) = 1.$$

The marginal gain of adding e_2 to $\varnothing$ is 0, while the marginal gain of adding e_2 after e_1 is present is 1. This violates diminishing returns, so P is not submodular. □

C.9. Batch-Separability Failure

Batch separability is also substantive. If selecting one action changes another action's physical output, action-level value can become complementary even when completed-portfolio value is monotone submodular.

Proposition C.11 (batch separability is necessary). If one action's realized output can depend on which other actions are selected, one-round value need not be submodular even when the completed-portfolio objective is monotone submodular.

Proof. Let

$$F(S) = \mathbf{1}\{y^\star \in S\},$$

which is monotone submodular over completed portfolios. Consider two actions. Action p opens a lock but produces no semantic item. Action q reads a file and returns $y^\star$ if and only if p was also selected in the same batch. Then the induced batch value satisfies

$$G(\varnothing) = 0, \qquad G(\{p\}) = 0, \qquad G(\{q\}) = 0, \qquad G(\{p, q\}) = 1.$$

The marginal value of adding q is 0 at $\varnothing$ but 1 after p is present. This violates submodularity. □

D. SCPS Planning, Optimization, and Implementation Layers

Appendix C gives the exact one-round action objective: for a fixed portfolio value F_t and a batch of completion-level macro-actions B,

$$G_t(B \mid S_t) = \mathbb{E}[F_t(S_t \cup U_B) - F_t(S_t) \mid I_t]$$

is normalized, monotone, and submodular under batch separability. A deployed controller cannot usually evaluate this object exactly. It estimates residual novelty, residual overlap, and cost from the current frontier and semantic memory. This appendix separates the exact objective from the planning surrogate and records the optimization and implementation guarantees that survive this move.

D.1. Exact Gain, Bonferroni Score, and Deployed Surrogate

For semantic coverage, Appendix C proves the residual-threshold representation

$$G_t^{\mathrm{sem}}(B \mid S_t) = \int_{\Theta_x} \int_0^1 \Pr\left(\bigcup_{e \in B} H_e(\theta, r) \mid I_t \right) dr\, d\mu_t(\theta),$$

where $H_e(\theta, r)$ is the event that action e covers threshold (θ, r) not already covered by S_t. Define the exact one-action novelty and pairwise overlap

$$v_t(e \mid S_t) := \int_{\Theta_x} \int_0^1 \Pr(H_e(\theta, r) \mid I_t)\, dr\, d\mu_t(\theta),$$

and

$$\omega_t(e, f \mid S_t) := \int_{\Theta_x} \int_0^1 \Pr(H_e(\theta, r) \cap H_f(\theta, r) \mid I_t)\, dr\, d\mu_t(\theta).$$

The second-order Bonferroni lower bound is

$$G_t^{\mathrm{sem}}(B \mid S_t) \geq L_t^{(2)}(B \mid S_t) := \sum_{e \in B} v_t(e \mid S_t) - \sum_{\{e,f\} \subseteq B} \omega_t(e, f \mid S_t). \tag{17}$$

This is an exact lower bound on semantic gain when v_t and ω_t are the true residual-hit probabilities.

The deployed SCPS batch score is a calibrated planning surrogate:

$$\widehat{L}_t(B \mid S_t, M_t) := \sum_{e \in B} \widehat{v}_t(e \mid S_t, M_t) - \beta_t \sum_{\{e,f\} \subseteq B} \widehat{\omega}_t(e, f \mid S_t, M_t) - \lambda_t \sum_{e \in B} \widehat{c}_t(e). \tag{18}$$

Here M_t is the semantic memory used by the implementation, $\widehat{v}_t$ estimates residual support, $\widehat{\omega}_t$ estimates residual overlap, and $\widehat{c}_t$ estimates current-round cost. The constants β_t and λ_t calibrate redundancy and cost.

Surrogate status

The exact gain G_t^{sem} is monotone submodular. The true second-order expression in Equation (17) is a Bonferroni lower bound on semantic gain. The deployed score in Equation (18), with estimated terms, tuned coefficients, and cost penalties, is a planning surrogate rather than an exact lower bound on semantic gain.

D.2. Submodularity of the Novelty–Overlap–Cost Surrogate

Although the deployed score is not itself an exact semantic-gain lower bound, it has useful structure under natural sign conditions. Nonnegative pairwise overlap penalties make the score submodular, though generally nonmonotone. The novelty and cost terms are modular, so their signs do not affect submodularity; the sign condition is needed for the pairwise overlap penalty.

Proposition D.1 (plug-in score is submodular under nonnegative overlap). Fix a finite action set E_t. Suppose $\beta_t \geq 0$ and $\widehat{\omega}_t(e, f) \geq 0$ for all pairs. We write $\widehat{\omega}_t(e, f)$ for the weight attached to the unordered pair $\{e, f\}$, with $\widehat{\omega}_t(e, f) = \widehat{\omega}_t(f, e)$. Then the set function

$$\widehat{L}_t(B) = \sum_{e \in B} \widehat{v}_t(e) - \beta_t \sum_{\{e,f\} \subseteq B} \widehat{\omega}_t(e, f) - \lambda_t \sum_{e \in B} \widehat{c}_t(e)$$

is submodular over $B \subseteq E_t$. It is generally not monotone.

Proof. Let $S \subseteq T \subseteq E_t$ and $x \notin T$. The marginal gain of adding x to S is

$$\widehat{L}_t(S \cup \{x\}) - \widehat{L}_t(S) = \widehat{v}_t(x) - \lambda_t \widehat{c}_t(x) - \beta_t \sum_{e \in S} \widehat{\omega}_t(x, e).$$

The marginal gain of adding x to T is

$$\widehat{L}_t(T \cup \{x\}) - \widehat{L}_t(T) = \widehat{v}_t(x) - \lambda_t \widehat{c}_t(x) - \beta_t \sum_{e \in T} \widehat{\omega}_t(x, e).$$

Because $S \subseteq T$, $\widehat{\omega}_t(x, e) \geq 0$, and $\beta_t \geq 0$,

$$\sum_{e \in S} \widehat{\omega}_t(x, e) \leq \sum_{e \in T} \widehat{\omega}_t(x, e).$$

Thus the marginal at S is at least the marginal at T, which is submodularity. Monotonicity can fail: take two actions a, b with $\widehat{v}_t(a) = \widehat{v}_t(b) = 1$, no cost term, and $\beta_t \widehat{\omega}_t(a, b) = 3$. Then $\widehat{L}_t(\{a\}) = 1$ but $\widehat{L}_t(\{a, b\}) = -1$. Adding b decreases the score. $\square$

Implementation note. If learned overlap estimates can be negative because of regression or calibration artifacts, the plug-in submodularity result in Appendix D.2 no longer applies. Clipping overlap estimates to the nonnegative range is one simple way to preserve the nonmonotone-submodular structure of the planning score.

D.3. One-Round Optimization Guarantees

The exact one-round action objective $G_t(\cdot \mid S_t)$ is monotone submodular under the conditions of Appendix C. Standard greedy reasoning therefore gives a self-contained cardinality-budget guarantee. This guarantee is per-round and frozen-objective; it does not claim global optimality of the closed-loop receding-horizon controller.

Proposition D.2 (cardinality greedy for the exact one-round gain). Let $f : 2^E \to \mathbb{R}_+$ be normalized, monotone, and submodular, let $1 \le k \le |E|$, and let

$$O \in \arg\max_{|B|\le k} f(B).$$

Let the greedy sequence start at $S_0 = \varnothing$ and, for $i = 0, \ldots, k-1$, choose

$$e_i \in \arg\max_{e\in E\setminus S_i} \left[f(S_i \cup \{e\}) - f(S_i)\right], \qquad S_{i+1} = S_i \cup \{e_i\}.$$

Then

$$f(S_k) \ge \left(1 - \left(1 - \frac{1}{k}\right)^k\right) f(O) \ge (1 - 1/e) f(O).$$

Proof. At step i, monotonicity gives

$$f(O) \le f(S_i \cup O).$$

By submodularity,

$$f(S_i \cup O) - f(S_i) \le \sum_{e\in O\setminus S_i} \left[f(S_i \cup \{e\}) - f(S_i)\right].$$

There are at most k terms in the sum, and greedy chooses the largest available marginal. Hence

$$f(O) - f(S_i) \le k \left[f(S_{i+1}) - f(S_i)\right].$$

Let $\Delta_i := f(O) - f(S_i)$. Then

$$\Delta_{i+1} \le \left(1 - \frac{1}{k}\right) \Delta_i.$$

Iterating gives

$$\Delta_k \le \left(1 - \frac{1}{k}\right)^k f(O),$$

so

$$f(S_k) \ge \left(1 - \left(1 - \frac{1}{k}\right)^k\right) f(O) \ge (1 - 1/e) f(O).$$

□

Proposition D.3 (plug-in planning error, additive form). Let $\mathcal{B}$ be a finite feasible batch family. Suppose f is the target one-round objective and $\widehat{f}$ is a plug-in estimate satisfying

$$\sup_{B\in\mathcal{B}} |\widehat{f}(B) - f(B)| \le \eta.$$

If $\widehat{B}$ is a ρ-additive maximizer of $\widehat{f}$ over $\mathcal{B}$, meaning

$$\widehat{f}(\widehat{B}) \ge \max_{B\in\mathcal{B}} \widehat{f}(B) - \rho,$$

then

$$f(\widehat{B}) \ge \max_{B\in\mathcal{B}} f(B) - 2\eta - \rho.$$

Proof. Let $B^\star \in \arg\max_{B\in\mathcal{B}} f(B)$. Uniform error gives

$$f(\widehat{B}) \ge \widehat{f}(\widehat{B}) - \eta.$$

By additive optimality,

$$\widehat{f}(\widehat{B}) \ge \widehat{f}(B^\star) - \rho.$$

Uniform error again gives

$$\widehat{f}(B^\star) \ge f(B^\star) - \eta.$$

Combining the three inequalities yields

$$f(\widehat{B}) \ge f(B^\star) - 2\eta - \rho.$$

□

Multiplicative variant. If $\widehat{B}$ satisfies $\widehat{f}(\widehat{B}) \ge \alpha \max_{B\in\mathcal{B}} \widehat{f}(B)$ for $0 \le \alpha \le 1$, then the same argument gives

$$f(\widehat{B}) \ge \alpha \max_{B\in\mathcal{B}} f(B) - (1+\alpha)\eta,$$

provided the multiplicative approximation statement is meaningful for the estimator class used. The additive form above is usually safer for nonmonotone or possibly negative surrogates.

D.4. Local Surrogate Error

The previous proposition assumes a uniform bound over a feasible family. The next elementary inequality decomposes local error in the novelty, overlap, and cost terms. Let

$$L_t(B) := \sum_{e\in B} v_t(e) - \beta_t \sum_{\{e,f\}\subseteq B} \omega_t(e,f) - \lambda_t \sum_{e\in B} c_t(e),$$

and define $\widehat{L}_t$ analogously with hatted quantities. Assume $\beta_t, \lambda_t \ge 0$.

Proposition D.4 (local surrogate error). For every finite batch B,

$$|\widehat{L}_t(B) - L_t(B)| \le \sum_{e\in B} |\widehat{v}_t(e) - v_t(e)| + \beta_t \sum_{\{e,f\}\subseteq B} |\widehat{\omega}_t(e,f) - \omega_t(e,f)| + \lambda_t \sum_{e\in B} |\widehat{c}_t(e) - c_t(e)|.$$

Proof. Subtract the two expressions and apply the triangle inequality term by term. □

D.5. Finite-Horizon Execution

SCPS is a receding-horizon controller. If a round executes each selected macro-action only up to a horizon H, then the exact current-round action objective should use artifacts completed within that horizon.

Let $Z_{e,\le H}$ be the random finite set of artifacts completed by action e within H tokens or steps, and let

$$U_{B,\le H} := \bigcup_{e\in B} Z_{e,\le H}.$$

For any frozen NMS portfolio value F_t, define

$$G_{t,\le H}(B \mid S_t) := \mathbb{E}[F_t(S_t \cup U_{B,\le H}) - F_t(S_t) \mid I_t]\,. \tag{19}$$

Proposition D.5 (finite-horizon macro-action closure). Under batch-separable truncated outputs $Z_{e,\le H}$, if F_t is NMS and $F_t(S_t \cup U_{B,\le H})$ is integrable for every B, then $G_{t,\le H}(\cdot \mid S_t)$ is NMS.

Proof. This is the macro-action closure in Appendix C.3 applied to the truncated random output sets $Z_{e,\le H}$ instead of Z_e. □

Interpretation. Full-completion predictors can still be useful as continuation-value proxies, but they should not be described as exact current-round by-H gains. The exact by-H objective is Equation (19).

D.6. Cost Safety

Cost predictions should match the resource constraint used by the controller. Mean-cost regression is useful for ranking, but it does not by itself certify hard feasibility.

Let $C^{\text{true}}_{t,H}(B)$ be the realized physical cost of executing batch B for the current round. Suppose the controller enforces the predicted-cost constraint

$$\sum_{e\in B} \widehat{c}_t(e) \le \bar{B}_t.$$

Proposition D.6 (deterministic cost certificate). If

$$C^{\text{true}}_{t,H}(B) \le \sum_{e\in B} \widehat{c}_t(e)$$

for every batch B considered by the planner, then every selected batch satisfying $\sum_{e\in B} \widehat{c}_t(e) \le \bar{B}_t$ is physically feasible under the round budget $\bar{B}_t$.

Proof. For any selected batch B,

$$C^{\text{true}}_{t,H}(B) \le \sum_{e\in B} \widehat{c}_t(e) \le \bar{B}_t.$$

□

High-probability variant. If the deterministic upper bound holds simultaneously for all batches in a finite feasible family $\mathcal{B}$ with probability at least $1-\delta$, then predicted-cost feasibility implies physical feasibility for any batch selected from $\mathcal{B}$ with probability at least $1-\delta$. This is a property of the cost certificate, not a consequence of mean-cost calibration alone.

D.7. Distribution-First Predictors

A scalar value head trained to predict one action's value under one memory state can become stale as soon as the portfolio changes. Profile or distribution predictors are more reusable because residual novelty can be recomputed against the current memory.

For a candidate action e, define the raw threshold-coverage event

$$C_e(\theta, r) := \left\{ \sup_{y\in Z_e} k_t(y,\theta) \ge r \right\}.$$

This event says that e reaches threshold (θ, r), ignoring whether the current portfolio already covers it. For a portfolio S, the residual event is

$$H^S_e(\theta, r) := \{r > m_{t,S}(\theta)\} \cap C_e(\theta, r).$$

Suppose a predictor estimates

$$p_e(\theta, r) := \Pr(C_e(\theta, r) \mid I_t)$$

and

$$p_{ef}(\theta, r) := \Pr(C_e(\theta, r) \cap C_f(\theta, r) \mid I_t).$$

Proposition D.7 (profile predictions can be reused across memories). For any fixed current portfolio S,

$$v_t(e \mid S) = \int_{\Theta_x} \int_0^1 \mathbf{1}\{r > m_{t,S}(\theta)\} p_e(\theta, r)\, dr\, d\mu_t(\theta),$$

and

$$\omega_t(e, f \mid S) = \int_{\Theta_x} \int_0^1 \mathbf{1}\{r > m_{t,S}(\theta)\} p_{ef}(\theta, r)\, dr\, d\mu_t(\theta).$$

Proof. For novelty,

$$v_t(e \mid S) = \int\int \Pr(H_e^S(\theta, r) \mid I_t)\, dr\, d\mu_t(\theta).$$

Since S and $m_{t,S}$ are fixed at planning time, the event $\{r > m_{t,S}(\theta)\}$ is deterministic conditional on I_t. Therefore

$$\Pr(H_e^S(\theta, r) \mid I_t) = \mathbf{1}\{r > m_{t,S}(\theta)\} \Pr(C_e(\theta, r) \mid I_t),$$

which gives the formula for v_t. The overlap formula is identical, using

$$H_e^S(\theta, r) \cap H_f^S(\theta, r) = \{r > m_{t,S}(\theta)\} \cap C_e(\theta, r) \cap C_f(\theta, r).$$

□

Scalar-head insufficiency example. Let the residual threshold universe consist of two unit-mass points u_1, u_2. Action e_1 deterministically covers u_1, and action e_2 deterministically covers u_2. At the empty portfolio both have scalar value 1. If the current portfolio already covers u_1, then e_1 has residual value 0 and e_2 has residual value 1. A scalar head that only stores the old value 1 for both actions cannot distinguish them after memory changes. A profile predictor can, because it stores which threshold each action is likely to cover.

Predict profiles, not just scalar values

Residual novelty changes whenever the portfolio changes. Coverage-profile, answer-distribution, or threshold-hit predictors can be recomputed against the current semantic memory. Scalar value heads are generally tied to the memory state on which they were trained.

E. Probes, Evaluator Approximation, and Robustness

This appendix analyzes approximation to the evaluator-induced objective. The results here certify internal validity: they show when a finite probe set, approximate kernel, or uncertainty envelope faithfully approximates the chosen semantic coverage objective. They do not certify that the evaluator has chosen the right semantic targets.

E.1. Finite Semantic Probes

Assume first that μ_t is a probability measure and draw independent probes $\theta_1, \dots, \theta_m \sim \mu_t$. For a fixed portfolio S, define

$$\widehat{F}^{\text{sem}}_{t,m}(S) := \frac{1}{m}\sum_{i=1}^{m}\left[\max\left\{b_t(\theta_i), \sup_{y\in S} k_t(y,\theta_i)\right\} - b_t(\theta_i)\right].$$

Proposition E.1 (sampled probe objective is NMS). For every fixed probe realization $\theta_1, \dots, \theta_m$, the function $S \mapsto \widehat{F}^{\text{sem}}_{t,m}(S)$ is normalized, monotone, and submodular.

Proof. For each probe θ_i, the summand is a finite-target semantic coverage function of the form proved in Theorem 1. A nonnegative average of NMS functions is NMS. □

Proposition E.2 (probe concentration for a fixed portfolio). For any fixed finite portfolio S and any $\varepsilon > 0$,

$$\Pr\left(\left|\widehat{F}^{\text{sem}}_{t,m}(S) - F^{\text{sem}}_t(S)\right| > \varepsilon\right) \le 2\exp(-2m\varepsilon^2).$$

Proof. For fixed S, define

$$X_i := \max\left\{b_t(\theta_i), \sup_{y\in S} k_t(y,\theta_i)\right\} - b_t(\theta_i).$$

Because $0 \le b_t, k_t \le 1$, each $X_i \in [0,1]$. Since $\theta_i \sim \mu_t$, $\mathbb{E}[X_i] = F^{\text{sem}}_t(S)$ and $\widehat{F}^{\text{sem}}_{t,m}(S) = m^{-1}\sum_i X_i$. Hoeffding's inequality gives the result. □

Finite-family version. If the controller evaluates a finite family $\mathcal{S}$ of portfolios or batch-induced portfolios, then a union bound gives

$$\Pr\left(\sup_{S\in\mathcal{S}}\left|\widehat{F}^{\text{sem}}_{t,m}(S) - F^{\text{sem}}_t(S)\right| > \varepsilon\right) \le 2|\mathcal{S}|\exp(-2m\varepsilon^2).$$

Thus, with probability at least $1-\alpha$,

$$\sup_{S\in\mathcal{S}}\left|\widehat{F}^{\text{sem}}_{t,m}(S) - F^{\text{sem}}_t(S)\right| \le \sqrt{\frac{\log(2|\mathcal{S}|/\alpha)}{2m}}.$$

Finite nonnormalized measure. If $M_t := \mu_t(\Theta_x) < \infty$ is nonzero, draw probes from $\bar{\mu}_t := \mu_t/M_t$ and use the scaled estimator $M_t\widehat{F}^{\text{sem}}_{t,m}$. Then

$$\Pr\left(\left|M_t\widehat{F}^{\text{sem}}_{t,m}(S) - F^{\text{sem}}_t(S)\right| > \varepsilon\right) \le 2\exp\left(-\frac{2m\varepsilon^2}{M_t^2}\right),$$

with the analogous finite-family union bound.

E.2. Semantic Kernel Approximation

Finite probes approximate the integral over semantic targets. A separate error source is the implemented evaluator or kernel.

Proposition E.3 (kernel sup-norm approximation). Let $\mathcal{Y}_0$ be a finite candidate artifact set. Suppose an implemented kernel $\widehat{k}_t$ satisfies

$$\sup_{y\in\mathcal{Y}_0, \theta\in\Theta_x} |\widehat{k}_t(y,\theta) - k_t(y,\theta)| \le \delta.$$

Let $\widehat{F}_t$ be semantic coverage computed with $\widehat{k}_t$ and the same b_t, μ_t. Then for every finite $S \subseteq \mathcal{Y}_0$,

$$|\widehat{F}_t(S) - F^{\text{sem}}_t(S)| \le \delta\,\mu_t(\Theta_x).$$

If μ_t is a probability measure, the bound is δ.

Proof. For each θ, define

$$m_S(\theta) = \max\left\{b_t(\theta), \sup_{y\in S} k_t(y,\theta)\right\}$$

and

$$\widehat{m}_S(\theta) = \max\left\{b_t(\theta), \sup_{y\in S} \widehat{k}_t(y,\theta)\right\}.$$

If two collections of real numbers differ coordinatewise by at most δ, their maxima differ by at most δ. Hence

$$|\widehat{m}_S(\theta) - m_S(\theta)| \le \delta$$

for every θ. Therefore

$$\begin{aligned}|\widehat{F}_t(S) - F^{\text{sem}}_t(S)| &\le \int_{\Theta_x} |\widehat{m}_S(\theta) - m_S(\theta)|\,d\mu_t(\theta)\\ &\le \delta\,\mu_t(\Theta_x).\end{aligned}$$

□

E.3. Combined Finite-Family Approximation Certificate

The probe and kernel bounds can be combined for a finite planning family. This is often the most useful form for a controller that selects among finitely many candidate batches.

Assume μ_t is normalized. Let $\widehat{F}_{m,\widehat{k}}$ denote the probe estimate using the approximate kernel $\widehat{k}_t$, and let F_t^{target} denote the exact semantic value using the target kernel k_t. Conditioning on any implemented kernel $\widehat{k}_t$ satisfying the displayed sup-norm bound, the probability below is over the probe draw.

Proposition E.4 (combined probe and kernel error). Suppose $\sup_{y\in\mathcal{Y}_0,\theta} |\widehat{k}_t(y,\theta) - k_t(y,\theta)| \le \delta$ and every $S \in \mathcal{S}$ is a subset of $\mathcal{Y}_0$. Then, with probability at least $1-\alpha$ over the probe draw,

$$\sup_{S\in\mathcal{S}} |\widehat{F}_{m,\widehat{k}}(S) - F_t^{\text{target}}(S)| \le \delta + \sqrt{\frac{\log(2|\mathcal{S}|/\alpha)}{2m}}.$$

For finite nonnormalized μ_t with total mass M_t, the right side is multiplied by M_t.

Proof. For each S, decompose

$$|\widehat{F}_{m,\widehat{k}}(S)-F_t^{\text{target}}(S)| \le |\widehat{F}_{m,\widehat{k}}(S)-\widehat{F}_{m,k}(S)|+|\widehat{F}_{m,k}(S)-F_t^{\text{target}}(S)|.$$

The first term is at most δ for every S and every probe realization by the same max-Lipschitz argument as the kernel approximation bound in Appendix E.2, applied to the finite probe average. The second term is uniformly bounded over $\mathcal{S}$ by the probe concentration bound in Appendix E.1 and the union bound. This proves the normalized case. Scaling by M_t gives the finite nonnormalized case. □

E.4. Robust Coverage Envelopes

The previous results quantify approximation to a fixed kernel. Sometimes the evaluator can provide lower and upper envelopes for an unknown target kernel $k_t^\star$. The next result gives a conservative portfolio-value sandwich.

Assume

$$\underline{k}_t(y,\theta) \le k_t^\star(y,\theta) \le \overline{k}_t(y,\theta)$$

for all relevant y, θ, with all kernels measurable and in $[0,1]$. Define

$$F_t^-(S) := \int \left[\max\left\{b_t(\theta), \sup_{y\in S} \underline{k}_t(y,\theta)\right\} - b_t(\theta)\right] d\mu_t(\theta),$$

$$F_t^\star(S) := \int \left[\max\left\{b_t(\theta), \sup_{y\in S} k_t^\star(y,\theta)\right\} - b_t(\theta)\right] d\mu_t(\theta),$$

and

$$F_t^+(S) := \int \left[\max\left\{b_t(\theta), \sup_{y\in S} \overline{k}_t(y,\theta)\right\} - b_t(\theta)\right] d\mu_t(\theta).$$

Proposition E.5 (portfolio-value sandwich). For every finite portfolio S,

$$F_t^-(S) \le F_t^\star(S) \le F_t^+(S).$$

Moreover, F_t^- and F_t^+ are normalized, monotone, and submodular.

Proof. For each θ,

$$\sup_{y\in S} \underline{k}_t(y,\theta) \le \sup_{y\in S} k_t^\star(y,\theta) \le \sup_{y\in S} \overline{k}_t(y,\theta).$$

Taking the maximum with the same baseline $b_t(\theta)$ preserves the inequalities. Subtracting $b_t(\theta)$ and integrating gives the sandwich. The functions F_t^- and F_t^+ are ordinary semantic coverage values with bounded measurable kernels, so Theorem 1 applies. □

E.5. Conservative Residual Gains Under Kernel Uncertainty

A subtle point is that the pessimistic portfolio value does not automatically give a pessimistic marginal gain. If the lower kernel underestimates what the current portfolio already covers, it can overstate the novelty of a new artifact.

Counterexample to naive pessimistic marginals. Consider one target of unit measure with baseline 0. Let the current portfolio be $S = \{z\}$ and the candidate be y. Suppose

$$k_t^\star(z) = 0.9, \qquad k_t^\star(y) = 0.9.$$

The true marginal gain of adding y is 0. Now let the lower envelope satisfy

$$\underline{k}_t(z) = 0, \qquad \underline{k}_t(y) = 0.5.$$

Then $F_t^-(S) = 0$ and $F_t^-(S \cup \{y\}) = 0.5$, so the pessimistic marginal is 0.5, larger than the true marginal. Conservative novelty must therefore compare lower-confidence new coverage against upper-confidence current coverage.

For a current portfolio S, define

$$\overline{m}_S(\theta) := \max\left\{b_t(\theta), \sup_{z\in S} \overline{k}_t(z,\theta)\right\}, \qquad \underline{m}_S(\theta) := \max\left\{b_t(\theta), \sup_{z\in S} \underline{k}_t(z,\theta)\right\}$$

For a candidate completed set Y, define the conservative lower gain

$$L_{\text{gain}}^-(Y \mid S) := \int \left[\sup_{y\in Y} \underline{k}_t(y,\theta) - \overline{m}_S(\theta)\right]_+ d\mu_t(\theta),$$

and the optimistic upper gain

$$L_{\text{gain}}^+(Y \mid S) := \int \left[\sup_{y\in Y} \overline{k}_t(y,\theta) - \underline{m}_S(\theta)\right]_+ d\mu_t(\theta).$$

Proposition E.6 (residual marginal sandwich). If $\underline{k}_t \le k_t^\star \le \overline{k}_t$, then for every finite current portfolio S and candidate set Y,

$$L_{\text{gain}}^-(Y \mid S) \le F_t^\star(S \cup Y) - F_t^\star(S) \le L_{\text{gain}}^+(Y \mid S).$$

Proof. Let

$$m_S^\star(\theta) := \max\left\{ b_t(\theta), \sup_{z \in S} k_t^\star(z, \theta) \right\}.$$

The true pointwise marginal is

$$\left[\sup_{y \in Y} k_t^\star(y, \theta) - m_S^\star(\theta) \right]_+.$$

The envelope assumptions imply

$$\sup_{y \in Y} \underline{k}_t(y, \theta) \le \sup_{y \in Y} k_t^\star(y, \theta) \le \sup_{y \in Y} \overline{k}_t(y, \theta),$$

and

$$\underline{m}_S(\theta) \le m_S^\star(\theta) \le \overline{m}_S(\theta).$$

Therefore, pointwise,

$$\sup_{y \in Y} \underline{k}_t(y, \theta) - \overline{m}_S(\theta) \le \sup_{y \in Y} k_t^\star(y, \theta) - m_S^\star(\theta) \le \sup_{y \in Y} \overline{k}_t(y, \theta) - \underline{m}_S(\theta).$$

Taking positive parts preserves order, and integrating over Θ_x proves the sandwich. □

Corollary E.7 (stochastic conservative gain). If the envelope holds for every possible realized output U_B, then

$$\mathbb{E}[L_{\text{gain}}^-(U_B \mid S_t) \mid I_t] \le \mathbb{E}[F_t^\star(S_t \cup U_B) - F_t^\star(S_t) \mid I_t] \le \mathbb{E}[L_{\text{gain}}^+(U_B \mid S_t) \mid I_t].$$

This follows by applying the residual marginal sandwich in Appendix E.5 realization by realization and taking conditional expectation.

E.6. Internal Validity Boundary

Probe concentration, kernel approximation, and robust envelopes all certify approximation to the evaluator-induced objective. They do not prove that the evaluator chose the correct semantic target space, assigned correct importance weights, or measured true scientific value. If the probe distribution is biased toward easy-to-detect targets, concentration simply certifies accurate estimation of that biased objective.

Evaluator-conditional approximation

The approximation results in this appendix are internal to the chosen evaluator-induced objective. They certify finite-probe error, kernel approximation error, or lower/upper envelope validity. They do not certify that the semantic evaluator matches human, mathematical, or scientific value.

F. Exact-Answer Quotient, ACPS, and Answer Banks

This appendix specializes semantic coverage to verifiable exact-answer tasks. The exact-answer case is useful because answer identity can often be certified, but it is still a special case of semantic coverage. The semantic target space is the utility quotient over completed traces, not necessarily a known multiple-choice option set.

F.1. Utility Quotient and Answer Coverage

Let $\mathcal{Z}_x^{\downarrow}$ be the space of completed traces for prompt x, and let $u(z,\tau) \in [0,1]$ be task utility against latent target τ. Define an equivalence relation

$$z \sim_u z' \quad \Longleftrightarrow \quad u(z,\tau) = u(z',\tau) \quad \text{for all latent targets } \tau.$$

The exact answer-class space is the quotient

$$\mathcal{A}_x := \mathcal{Z}_x^{\downarrow} / \sim_u,$$

with quotient map $a : \mathcal{Z}_x^{\downarrow} \to \mathcal{A}_x$. In binary exact correctness, the latent target is $A^\star \in \mathcal{A}_x$ and

$$u(z, A^\star) = \mathbf{1}\{a(z) = A^\star\}.$$

For a portfolio $S \subseteq \mathcal{Z}_x^{\downarrow}$, write

$$A(S) := \{a(z) : z \in S\},$$

and let

$$\pi_t(a) := \Pr(A^\star = a \mid I_t).$$

When $\mathcal{A}_x$ is finite or countable, frozen answer coverage is

$$F_t^{\text{ans}}(S) := \sum_{a \in \mathcal{A}_x} \pi_t(a) \mathbf{1}\{a \in A(S)\}. \tag{20}$$

The finite or countable case is the main exact-answer setting considered in this appendix. For a general quotient space, the same object is written as the corresponding integral over the quotient measure. In a nonatomic quotient, a finite portfolio may cover zero posterior measure unless answer classes have atoms or are coarsened into measurable regions.

Proposition F.1 (answer coverage is semantic coverage). F_t^{ans} is a latent semantic coverage objective and is therefore NMS.

Proof. Take

$$\Theta_x = \mathcal{A}_x, \qquad \mu_t(a) = \pi_t(a), \qquad b_t(a) = 0,$$

and define

$$k_t(z,a) = \mathbf{1}\{a(z) = a\}.$$

Then

$$\sup_{z \in S} k_t(z,a) = \mathbf{1}\{\exists z \in S : a(z) = a\} = \mathbf{1}\{a \in A(S)\}.$$

Substituting into semantic coverage gives Equation (20). The NMS property follows from Theorem 1. □

F.2. Frozen Coverage, Truth-Coupled Hit, and Selector Accuracy

Exact-answer tasks contain several related but distinct objects. Confusing them is one of the easiest ways to overstate the theory.

Let E_t be a set of rollout actions. Suppose action e returns an answer class $A_e \in \mathcal{A}_x \cup \{\perp\}$ when it completes, where $\perp$ denotes no certified answer. Let

$$A_t^{\text{seen}} := A(S_t).$$

The truth-coupled correct-hit objective is

$$G_t^{\text{hit}}(B \mid S_t) := \Pr(A^\star \notin A_t^{\text{seen}},\ \exists e \in B : A_e = A^\star \mid I_t)\,. \tag{21}$$

The frozen answer-coverage lift is

$$\overline{G}_t^{\text{ans}}(B \mid S_t) := \mathbb{E}[F_t^{\text{ans}}(S_t \cup R_B) - F_t^{\text{ans}}(S_t) \mid I_t]\,, \tag{22}$$

where R_B is the set of completed traces returned by the selected actions. Future selector accuracy is a third object, for a downstream selector σ:

$$\Pr(\sigma(S_t \cup R_B) = A^\star \mid I_t)\,.$$

Proposition F.2 (frozen lift and one-round hit are submodular objects). For fixed π_t and fixed answer classes, $\overline{G}_t^{\text{ans}}(\cdot \mid S_t)$ is NMS over batch actions under batch separability. If the potential answer outputs $(A_e)_{e \in E_t}$ are fixed random variables attached to the actions for the round, then $G_t^{\text{hit}}(\cdot \mid S_t)$ is also NMS. Future selector accuracy and adaptive truth-coupled discovery do not inherit these statements without an explicit selector or observation model; Appendix G gives boundary examples.

Proof. By the answer-coverage quotient in Appendix F.1, F_t^{ans} is NMS. Applying the macro-action closure theorem from Appendix C to F_t^{ans} gives the NMS property of $\overline{G}_t^{\text{ans}}$.

For the truth-coupled hit objective, define the fixed event

$$H_e^{\text{hit}} := \{A^\star \notin A_t^{\text{seen}},\ A_e = A^\star\}.$$

Then

$$G_t^{\text{hit}}(B \mid S_t) = \Pr\left(\bigcup_{e \in B} H_e^{\text{hit}} \mid I_t\right).$$

Normalization and monotonicity follow from the empty-union convention and event inclusion. For submodularity, let $A \subseteq B \subseteq E_t$ and $e \notin B$. Write

$$U_A := \bigcup_{f\in A} H_f^{\text{hit}}, \qquad U_B := \bigcup_{f\in B} H_f^{\text{hit}}.$$

Since $U_A \subseteq U_B$, the marginal of adding e to A is

$$\Pr(H_e^{\text{hit}} \setminus U_A \mid I_t),$$

while the marginal of adding e to B is

$$\Pr(H_e^{\text{hit}} \setminus U_B \mid I_t).$$

The inclusion $H_e^{\text{hit}} \setminus U_B \subseteq H_e^{\text{hit}} \setminus U_A$ gives the diminishing-returns inequality. Future selector accuracy additionally depends on the selector, and adaptive truth-coupled discovery changes the observation or belief model, so both are separate objects. □

Counterexample: frozen coverage need not upper-bound hit or selector accuracy. Let $A^\star$ be uniform on $\{a, b\}$, so $\pi_t(a) = \pi_t(b) = 1/2$. Suppose one action returns a completed trace z_e whose answer class is the true answer:

$$a(z_e) = A_e = A^\star.$$

Then the truth-coupled hit probability is 1, and a selector that emits the returned answer is correct with probability 1. But the frozen answer coverage of the realized singleton portfolio is

$$F_t^{\text{ans}}(\{z_e\}) = \pi_t(a(z_e)) = \pi_t(A^\star) = 1/2.$$

Thus selected accuracy and truth-coupled hit can exceed frozen posterior coverage when action outputs are informative about the truth.

When frozen coverage equals truth-coupled oracle coverage. If $A(S)$ is conditionally independent of $A^\star$ given I_t, then

$$\Pr(A^\star \in A(S) \mid I_t) = \mathbb{E}[F_t^{\text{ans}}(S) \mid I_t].$$

Indeed,

$$\begin{aligned}\Pr(A^\star \in A(S) \mid I_t) &= \mathbb{E}[\Pr(A^\star \in A(S) \mid A(S), I_t) \mid I_t] \\ &= \mathbb{E}\left[\sum_a \pi_t(a)\mathbf{1}\{a \in A(S)\} \mid I_t\right] \\ &= \mathbb{E}[F_t^{\text{ans}}(S) \mid I_t].\end{aligned}$$

This independence condition is often too strong for reasoning systems, because useful rollouts are informative about the true answer. Accordingly, frozen coverage, truth-coupled hit or oracle coverage, and final selected accuracy should be reported separately.

Distinct exact-answer quantities

Frozen answer coverage, truth-coupled correct hit, and future selector accuracy are different quantities. The macro-action result applies directly to frozen coverage, and one-round truth-coupled hit is a union-probability coverage objective when potential answer outputs are fixed for the round. Adaptive truth-coupled and selector-level claims require their own assumptions or diagnostics.

F.3. Product Form for Frozen Answer Coverage

The exact-answer case has a simple product form when per-answer miss events factorize.

For an unseen answer class $a \notin A_t^{\text{seen}}$, let

$$p_t(a \mid e) := \Pr(A_e = a \mid I_t).$$

Assume unconditional miss factorization for each unseen answer:

$$\Pr\left(\bigcap_{e\in B}\{A_e \neq a\} \mid I_t\right) = \prod_{e\in B}(1 - p_t(a \mid e)). \tag{23}$$

Proposition F.3 (product form for frozen answer coverage). Under Equation (23),

$$\overline{G}_t^{\text{ans}}(B \mid S_t) = \sum_{a\notin A_t^{\text{seen}}} \pi_t(a)\left[1 - \prod_{e\in B}(1 - p_t(a \mid e))\right].$$

Proof. For a fixed unseen answer a, the frozen coverage gain includes $\pi_t(a)$ exactly when at least one selected action returns answer class a. Therefore

$$\overline{G}_t^{\text{ans}}(B \mid S_t) = \sum_{a\notin A_t^{\text{seen}}} \pi_t(a)\Pr\left(\bigcup_{e\in B}\{A_e = a\} \mid I_t\right).$$

Using miss factorization and taking complements gives

$$\Pr\left(\bigcup_{e\in B}\{A_e = a\} \mid I_t\right) = 1 - \prod_{e\in B}(1 - p_t(a \mid e)).$$

Substitution proves the formula. □

F.4. Answer Banks and Coarsened Coverage

When $\mathcal{A}_x$ is too large, latent, or expensive to canonicalize exactly, ACPS may use a finite answer bank. A bank is a map

$$\phi_t : \mathcal{A}_x \to \mathcal{B}_t.$$

Its buckets induce masses

$$w_t(b) := \sum_{a:\phi_t(a)=b} \pi_t(a),$$

and action-level bucket probabilities

$$q_t(b \mid e) := \sum_{a:\phi_t(a)=b} p_t(a \mid e).$$

The coarsened bank value is

$$F_t^{\phi_t}(S) := \sum_{b\in\mathcal{B}_t} w_t(b)\mathbf{1}\{\exists a \in A(S) : \phi_t(a) = b\}. \quad (24)$$

Proposition F.4 (bank value is NMS). For fixed ϕ_t and nonnegative bucket masses w_t, $F_t^{\phi_t}$ is normalized, monotone, and submodular.

Proof. Equation (24) is ordinary finite weighted coverage over bucket targets. Equivalently, take semantic targets to be buckets $b \in \mathcal{B}_t$, baseline 0, measure $w_t(b)$, and trace-level coverage kernel $k_t(z, b) = \mathbf{1}\{\phi_t(a(z)) = b\}$. The result follows from Theorem 1. □

Proposition F.5 (bank optimism). For every completed portfolio S,

$$F_t^{\phi_t}(S) \geq F_t^{\text{ans}}(S).$$

Moreover, if

$$C_b(S) := A(S) \cap \phi_t^{-1}(b),$$

then the optimism gap decomposes as

$$F_t^{\phi_t}(S) - F_t^{\text{ans}}(S) = \sum_{b:C_b(S)\neq\varnothing} \sum_{a\in\phi_t^{-1}(b)\setminus C_b(S)} \pi_t(a).$$

Proof. For each bucket b, the bank value credits the full bucket mass $w_t(b)$ if any represented answer from that bucket appears in the portfolio. Exact answer coverage credits only the masses of represented answer classes. Thus, bucket by bucket,

$$w_t(b)\mathbf{1}\{C_b(S) \neq \varnothing\} - \sum_{a\in C_b(S)} \pi_t(a) = \begin{cases} \sum_{a\in\phi_t^{-1}(b)\setminus C_b(S)} \pi_t(a), & C_b(S) \neq \varnothing, \\ 0, & C_b(S) = \varnothing. \end{cases}$$

Summing over buckets gives the decomposition and nonnegativity. □

F.5. Bank Refinement Monotonicity

Bank optimism becomes controlled when banks are refined. A coarse bank overcredits support; splitting buckets monotonically removes that optimism.

Let $\mathcal{P}$ be a partition of $\mathcal{A}_x$. Define

$$\pi_t(P) := \sum_{a\in P} \pi_t(a),$$

and

$$F_t^{\mathcal{P}}(S) := \sum_{P\in\mathcal{P}} \pi_t(P)\mathbf{1}\{A(S) \cap P \neq \varnothing\}.$$

A partition $\mathcal{Q}$ refines $\mathcal{P}$ if every $Q \in \mathcal{Q}$ is contained in some $P \in \mathcal{P}$.

Proposition F.6 (refinement removes bank optimism). If $\mathcal{Q}$ refines $\mathcal{P}$, then for every portfolio S,

$$F_t^{\mathcal{P}}(S) \geq F_t^{\mathcal{Q}}(S).$$

If $\mathcal{S}$ is the singleton partition of $\mathcal{A}_x$, then

$$F_t^{\mathcal{P}}(S) \geq F_t^{\mathcal{Q}}(S) \geq F_t^{\mathcal{S}}(S) = F_t^{\text{ans}}(S)$$

whenever $\mathcal{Q}$ refines $\mathcal{P}$.

Proof. Fix a coarse cell $P \in \mathcal{P}$ and let

$$\mathcal{Q}(P) := \{Q \in \mathcal{Q} : Q \subseteq P\}.$$

If $A(S) \cap P = \varnothing$, then $A(S) \cap Q = \varnothing$ for every $Q \in \mathcal{Q}(P)$, so the contribution of P and its refinements is zero. If $A(S) \cap P \neq \varnothing$, then the coarse contribution is

$$\pi_t(P) = \sum_{Q\in\mathcal{Q}(P)} \pi_t(Q),$$

while the refined contribution is

$$\sum_{Q\in\mathcal{Q}(P)} \pi_t(Q)\mathbf{1}\{A(S)\cap Q \neq \varnothing\} \leq \sum_{Q\in\mathcal{Q}(P)} \pi_t(Q) = \pi_t(P).$$

Thus the coarse contribution is at least the refined contribution inside every P. Summing over $P \in \mathcal{P}$ proves $F_t^{\mathcal{P}}(S) \geq F_t^{\mathcal{Q}}(S)$. Since the singleton partition refines every partition, applying the first inequality again gives $F_t^{\mathcal{Q}}(S) \geq F_t^{\mathcal{S}}(S)$. The singleton partition also gives

$$F_t^{\mathcal{S}}(S) = \sum_{a\in\mathcal{A}_x} \pi_t(a)\mathbf{1}\{a \in A(S)\} = F_t^{\text{ans}}(S).$$

□

Bank refinement certificate

Coarse banks overcredit support because one represented answer can activate an entire mixed bucket. Refinement monotonically removes this optimism, and the singleton partition recovers exact answer coverage.

F.6. Singletonized Seen Answers and Unresolved Mixed Mass

A live bank can be exact on the current portfolio even if it remains coarse elsewhere. The key is to singletonize every answer that has actually been seen.

This subsection assumes the live bank partitions the answer support being compared to F_t^{ans}. With omitted support, the statements apply on modeled support and should be combined with the omitted-mass bound in Appendix F.8.

Let the live bank ontology be

$$\mathcal{B}_t = \mathcal{A}_t^{\text{disc}} \sqcup \mathcal{H}_t^{\text{prov}},$$

where every discovered answer class is represented by a singleton bucket and every $h \in \mathcal{H}_t^{\text{prov}}$ is a genuinely mixed provisional bucket containing no currently seen answer class. Define unresolved mixed mass

$$\Gamma_t := \sum_{h \in \mathcal{H}_t^{\text{prov}}} w_t(h).$$

Let $\overline{G}_t^{\phi_t}(B \mid S_t)$ be the frozen one-round lift of the bank value $F_t^{\phi_t}$.

Proposition F.7 (singletonized banks are exact on the current state). If every currently seen answer is singletonized, then

$$F_t^{\phi_t}(S_t) = F_t^{\text{ans}}(S_t).$$

Moreover, for every batch B,

$$0 \le \overline{G}_t^{\phi_t}(B \mid S_t) - \overline{G}_t^{\text{ans}}(B \mid S_t) \le \Gamma_t.$$

Proof. Because every seen answer has its own singleton bucket, the bank credits exactly the posterior mass of seen answers at S_t. Hence $F_t^{\phi_t}(S_t) = F_t^{\text{ans}}(S_t)$.

For a realized future portfolio $S_t \cup R_B$, bank optimism implies

$$F_t^{\phi_t}(S_t \cup R_B) - F_t^{\text{ans}}(S_t \cup R_B) \ge 0.$$

The equality on S_t turns this into nonnegative gain optimism:

$$\left[F_t^{\phi_t}(S_t \cup R_B) - F_t^{\phi_t}(S_t)\right] - [F_t^{\text{ans}}(S_t \cup R_B) - F_t^{\text{ans}}(S_t)] \ge 0.$$

Only mixed provisional buckets can produce optimism beyond exact answer coverage, because singleton buckets are exact. The total mass of those mixed buckets is Γ_t, so for every realization the gain optimism is at most Γ_t. Taking conditional expectation gives the one-round bound. □

F.7. Product-Form Bank Objective

For bank events involving uncertified action outputs, use the convention

$$\phi_t(\perp) := \perp, \qquad \perp \notin \mathcal{B}_t.$$

Let

$$\mathcal{B}_t^{\text{seen}} := \{\phi_t(a) : a \in A_t^{\text{seen}}\}.$$

Assume bucket-level miss events factorize:

$$\Pr\left(\bigcap_{e \in B} \{\phi_t(A_e) \ne b\} \mid I_t\right) = \prod_{e \in B} (1 - q_t(b \mid e)).$$

Proposition F.8 (product form for bank coverage). Under bucket-level miss factorization, the coarsened frozen bank lift is

$$\overline{G}_t^{\phi_t}(B \mid S_t) = \sum_{b \notin \mathcal{B}_t^{\text{seen}}} w_t(b) \left[1 - \prod_{e \in B} (1 - q_t(b \mid e))\right].$$

Proof. The bank gain credits bucket b exactly if b was not already seen and at least one selected action returns an answer mapped to b. Thus

$$\overline{G}_t^{\phi_t}(B \mid S_t) = \sum_{b \notin \mathcal{B}_t^{\text{seen}}} w_t(b) \Pr\left(\bigcup_{e \in B} \{\phi_t(A_e) = b\} \mid I_t\right).$$

Taking complements and using bucket-level miss factorization gives the formula. □

F.8. Omitted Support

If the bank models only a subset $Q_t \subseteq \mathcal{A}_x$, define omitted mass

$$\Lambda_t := \sum_{a \notin Q_t} \pi_t(a).$$

The modeled exact-answer value is

$$F_{t,Q}^{\text{ans}}(S) := \sum_{a \in Q_t} \pi_t(a) \mathbf{1}\{a \in A(S)\}.$$

Proposition F.9 (omitted-mass bound). For every portfolio S,

$$0 \le F_t^{\text{ans}}(S) - F_{t,Q}^{\text{ans}}(S) \le \Lambda_t.$$

Proof. The difference is

$$F_t^{\text{ans}}(S) - F_{t,Q}^{\text{ans}}(S) = \sum_{a \notin Q_t} \pi_t(a) \mathbf{1}\{a \in A(S)\}.$$

This is nonnegative and at most $\sum_{a \notin Q_t} \pi_t(a) = \Lambda_t$. □

Interpretation. Bank optimism is certified only on modeled support unless omitted mass is included in the accounting. A coarse bank over modeled answers can overcredit represented buckets; an incomplete bank can also miss posterior mass outside the modeled subset.

F.9. Candidate-Superset Canonicalization

Canonicalization is an implementation layer, not a theorem-level primitive. If a completed trace cannot be certified to a singleton exact answer class, the controller should not insert it into the exact seen set. Instead, it may store a candidate superset

$$C_t(z) \subseteq \mathcal{A}_x$$

or a provisional bucket. The trace may still inform approximate bank probabilities, semantic memory, or future refinement, but exact answer accounting should wait until singleton certification is available.

Invariant. A safe ACPS implementation maintains two separate structures:

- an exact seen set A_t^{seen} containing only singleton-certified answer classes;
- unresolved candidate supersets or provisional buckets for traces whose answer identity has not yet been certified.

This preserves exact accounting for discovered answers while allowing unresolved artifacts to participate in approximate planning.

F.10. ACPS Loop in the Exact-Answer Case

In exact-answer tasks, SCPS becomes Answer-Coverage Portfolio Search (ACPS). The semantic memory contains

$$(S_t, A_t^{\text{seen}}, \text{optional candidate supersets}, \phi_t, w_t, \Gamma_t, \Lambda_t),$$

where the bank-related terms are present only when a finite bank approximation is used.

At each round, ACPS:

1. builds completion-level rollout actions from the frontier;
2. scores actions using exact answer distributions, bank distributions, or novelty–overlap surrogates;
3. selects a budget-feasible batch;
4. executes selected actions;
5. singleton-certifies completed answers when possible;
6. stores unresolved traces as candidate supersets or provisional buckets;
7. refines the bank and replans.

At budget exhaustion, ACPS returns the completed portfolio to a terminal selector. Portfolio support, truth-coupled hit or oracle coverage, and final selected accuracy should be reported separately.

G. Closed-Loop, Adaptive, and Selector Boundaries

The exact results in Appendices B to C are frozen-round statements: the evaluator-induced objective is fixed, and the controller evaluates a one-round batch of completion-level actions. A deployed system may replan after observations, update semantic weights or answer banks, or pass the resulting portfolio to a terminal selector. Those operations are useful, but they are different mathematical objects. This appendix records what can be said exactly, when an adaptive positive result is available, and where counterexamples appear.

G.1. Truth-Coupled Frontier–Search–Selector Decomposition

In exact-answer tasks, it is useful to separate three layers: the frontier made available to search, the controller's acquisition of support from that frontier, and the terminal selector's use of the acquired support. This decomposition is about truth-coupled oracle coverage, not frozen posterior coverage.

Let $A^\star$ be the true answer class. Let S_{front} denote the completed portfolio that would be obtained by exhausting the available frontier action set under the same fixed frontier assets. Let S_Π be the portfolio obtained by a search policy Π under the allowed budget. Let σ be a terminal selector that emits an answer after observing S_Π.

Assume the following two conditions.

1. Frontier containment. For every realization,
$$A(S_\Pi) \subseteq A(S_{\text{front}}).$$
2. Portfolio-respecting selector. If the selector is correct, then the correct answer is represented in the realized portfolio:
$$\{\sigma(S_\Pi) = A^\star\} \subseteq \{A^\star \in A(S_\Pi)\}.$$

Define
$$C_{\text{front}} := \Pr(A^\star \in A(S_{\text{front}}) \mid I_t),$$
$$C_\Pi := \Pr(A^\star \in A(S_\Pi) \mid I_t),$$
and
$$A_{\Pi,\sigma} := \Pr(\sigma(S_\Pi) = A^\star \mid I_t).$$

Proposition G.1 (frontier–search–selector decomposition). Under frontier containment and a portfolio-respecting selector,
$$A_{\Pi,\sigma} \le C_\Pi \le C_{\text{front}} \le 1.$$

Consequently,
$$1-A_{\Pi,\sigma} = \underbrace{(1-C_{\text{front}})}_{\text{frontier miss}} + \underbrace{(C_{\text{front}}-C_\Pi)}_{\text{search miss}} + \underbrace{(C_\Pi - A_{\Pi,\sigma})}_{\text{selector miss}}. \tag{25}$$

Proof. By frontier containment,
$$\{A^\star \in A(S_\Pi)\} \subseteq \{A^\star \in A(S_{\text{front}})\},$$
so $C_\Pi \le C_{\text{front}}$. By the portfolio-respecting selector assumption,
$$\{\sigma(S_\Pi) = A^\star\} \subseteq \{A^\star \in A(S_\Pi)\},$$
so $A_{\Pi,\sigma} \le C_\Pi$. The inequality $C_{\text{front}} \le 1$ is immediate. Equation (25) is the telescoping expansion
$$1 - A_{\Pi,\sigma} = 1 - C_{\text{front}} + C_{\text{front}} - C_\Pi + C_\Pi - A_{\Pi,\sigma}.$$
Each term is nonnegative by the inequalities above. □

Interpretation. The first gap is a frontier-construction failure: the correct answer was not available in the exhausted action space. The second gap is a search failure: the correct answer was available in the frontier but was not acquired by the controller. The third gap is a selector failure: the correct answer was acquired but not emitted.

> **Why report portfolio support separately?**
>
> A final selected answer can fail for reasons unrelated to search acquisition. The decomposition in Equation (25) separates frontier miss, search miss, and selector miss. This is why realized portfolio support, oracle or hit coverage, and final selected accuracy should be reported as different quantities.

G.2. Frozen Posterior Coverage Is Not the Same Object

The decomposition above uses truth-coupled oracle coverage $\Pr(A^\star \in A(S))$. It should not be replaced blindly by frozen answer coverage
$$F_t^{\text{ans}}(S) = \sum_a \pi_t(a)\mathbf{1}\{a \in A(S)\}.$$

Appendix F separates frozen answer coverage from truth-coupled hit probability. One-round truth-coupled hit is also a coverage-style NMS objective under fixed potential answer outputs, but it is still not interchangeable with frozen posterior coverage.

Counterexample G.2 (selected accuracy can exceed frozen coverage). Let $A^\star$ be uniform on $\{a,b\}$ under I_t. Consider one rollout action that always returns the true answer:

$$A_e = A^\star.$$

Let S be the resulting one-answer portfolio. A selector that emits the returned answer is correct with probability one:

$$\Pr(\sigma(S) = A^\star \mid I_t) = 1.$$

However, the frozen answer coverage of the realized portfolio is always

$$F_t^{\text{ans}}(S) = \pi_t(A^\star) = 1/2.$$

Thus selected accuracy can be larger than expected frozen coverage when the portfolio is informative about the truth. Frozen posterior coverage and truth-coupled hit are not interchangeable.

When the two coincide. If $A(S)$ is conditionally independent of $A^\star$ given I_t, then

$$\Pr(A^\star \in A(S) \mid I_t) = \mathbb{E}[F_t^{\text{ans}}(S) \mid I_t].$$

Indeed,

$$\begin{aligned}\Pr(A^\star \in A(S) \mid I_t) &= \mathbb{E}\big[\Pr(A^\star \in A(S) \mid A(S), I_t) \mid I_t\big] \\ &= \mathbb{E}\left[\sum_a \pi_t(a)\mathbf{1}\{a \in A(S)\} \mid I_t\right] \\ &= \mathbb{E}[F_t^{\text{ans}}(S) \mid I_t].\end{aligned}$$

This independence condition is often not the intended model for reasoning rollouts, because useful rollouts may be informative about the true answer.

G.3. A Positive Adaptive Result Under Independent Stochastic Coverage

The frozen macro-action theorem does not by itself imply adaptive submodularity under arbitrary belief updates. There is, however, a clean positive result under an independent stochastic coverage model. This result is useful because it shows exactly what kind of observation model preserves diminishing returns after sequential observations.

Let $(\mathcal{U}, \mathcal{G}, \nu)$ be a finite-measure residual support space. One may think of $\mathcal{U}$ as a finite probe approximation to the residual threshold space (θ, r) from Appendix C, although the result does not require that interpretation. Each action $e \in E$ has a hidden random measurable coverage set

$$R_e \subseteq \mathcal{U}.$$

Executing e reveals R_e. Assume that conditional on I_t, the random sets $\{R_e : e \in E\}$ are mutually independent. Observing one action reveals additional covered support but does not change the conditional distribution of unexecuted action coverage sets.

A partial realization ψ assigns realized sets R_e to actions in $\text{dom}(\psi)$. Define

$$C(\psi) := \bigcup_{e \in \text{dom}(\psi)} R_e$$

and

$$f(\psi) := \nu(C(\psi)).$$

For an unobserved action $e \notin \text{dom}(\psi)$, define its adaptive marginal

$$\Delta(e \mid \psi) := \mathbb{E}\big[f(\psi \cup \{(e, R_e)\}) - f(\psi) \mid \psi, I_t\big].$$

Proposition G.3 (adaptive submodularity under independent coverage states). Under the independent stochastic coverage model, f is adaptive monotone and adaptive submodular. That is,

$$\Delta(e \mid \psi) \geq 0,$$

and whenever $\psi \subseteq \psi'$ and $e \notin \text{dom}(\psi')$,

$$\Delta(e \mid \psi) \geq \Delta(e \mid \psi').$$

Proof. For any partial realization ψ,

$$\begin{aligned}f(\psi \cup \{(e, R_e)\}) - f(\psi) &= \nu(C(\psi) \cup R_e) - \nu(C(\psi)) \\ &= \nu(R_e \setminus C(\psi)).\end{aligned}$$

Therefore

$$\Delta(e \mid \psi) = \mathbb{E}[\nu(R_e \setminus C(\psi)) \mid \psi, I_t].$$

This quantity is nonnegative, proving adaptive monotonicity.

By conditional independence, R_e is independent of the observations in ψ, so its conditional distribution given ψ is the same as its distribution given I_t. Now let $\psi \subseteq \psi'$. Then

$$C(\psi) \subseteq C(\psi').$$

For every realized R_e,

$$R_e \setminus C(\psi') \subseteq R_e \setminus C(\psi),$$

so

$$\nu(R_e \setminus C(\psi')) \leq \nu(R_e \setminus C(\psi)).$$

Taking expectation over the unchanged distribution of R_e gives

$$\Delta(e \mid \psi') \leq \Delta(e \mid \psi).$$

Thus adaptive submodularity holds. □

Positive adaptive case

Adaptive diminishing returns hold for independent stochastic coverage sets with full observation of executed actions. This is a stronger observation model than the batch-separable one-round theorem: it assumes observing previous actions does not change the distribution of unexecuted action coverage sets.

G.4. Adaptive Dependence Can Break Diminishing Returns

The independence condition in Appendix G.3 is not cosmetic. Correlation between action outcomes can make marginal value increase after an observation.

Counterexample G.4 (dependent coverage states are not adaptively submodular). Let $\mathcal{U} = \{u\}$ with $\nu(\{u\}) = 1$. There are two actions e_1, e_2. Exactly one action covers u:

$$(R_{e_1}, R_{e_2}) = (\{u\}, \varnothing)$$

with probability 1/2, and

$$(R_{e_1}, R_{e_2}) = (\varnothing, \{u\})$$

with probability 1/2.

Before observing anything,

$$\Delta(e_2 \mid \varnothing) = 1/2.$$

If e_1 is executed and reveals $R_{e_1} = \varnothing$, then the posterior implies $R_{e_2} = \{u\}$. The covered set is still empty, but

$$\Delta(e_2 \mid R_{e_1} = \varnothing) = 1.$$

Thus the marginal value of e_2 increased from 1/2 to 1, violating adaptive submodularity.

G.5. Truth-Coupled Answer Discovery Can Also Fail Adaptively

The previous counterexample is purely about correlated coverage states. Exact-answer discovery adds another source of adaptive failure: observations can update the posterior over the true answer itself.

Counterexample G.5 (truth-coupled correct-hit discovery need not be adaptively submodular). Let $A^\star$ be uniform on $\{a, b\}$, and let the current portfolio be empty. Consider two rollout actions. Action e_a returns a if $A^\star = a$ and returns $\perp$ if $A^\star = b$. Action e_b always returns answer b.

Before observing anything, the marginal truth-coupled correct-hit value of e_b is

$$\Delta(e_b \mid \varnothing) = \Pr(A^\star = b) = 1/2.$$

If e_a is executed and returns $\perp$, then the posterior puts probability one on $A^\star = b$. The current portfolio still does not contain the correct answer, and executing e_b now hits it with probability one:

$$\Delta(e_b \mid e_a = \perp) = 1.$$

The marginal value of e_b increased after an observation, violating adaptive submodularity.

Interpretation. The frozen answer-coverage lift in Appendix F remains submodular for a fixed posterior. The truth-coupled adaptive process above is different: observations change beliefs about $A^\star$.

G.6. Terminal Selector Accuracy Is a Separate Object

A terminal selector may abstain, choose the wrong represented answer, use scores that are not calibrated to the current portfolio, or be distracted by additional candidates. Therefore actual selected accuracy does not inherit the frozen coverage theorem unless the selector is explicitly modeled. In this subsection, $\sigma(S)$ denotes an emitted answer class; if a selector emits a trace, correctness means $a(\sigma(S)) = A^\star$.

The safest general inequality is the portfolio-respecting one used in the decomposition in Appendix G.1:

$$\Pr(\sigma(S) = A^\star \mid I_t) \leq \Pr(A^\star \in A(S) \mid I_t).$$

This is a statement about truth-coupled oracle coverage. It is not a statement about frozen posterior coverage.

Counterexample G.6 (an actual selector can be non-monotone). Let the true answer be a with posterior probability 0.6 and b with posterior probability 0.4. Consider a deterministic selector that emits the most recently added represented answer. If the portfolio is $S = \{a\}$, selected accuracy is 0.6. If a new completion representing b is added, the selector emits b, and selected accuracy becomes 0.4. Thus actual selector accuracy can decrease when the portfolio grows.

This counterexample is intentionally simple. The point is not that one should use such a selector, but that selector behavior is an additional algorithmic layer. The ideal selector envelope in Appendix B is monotone; an arbitrary or miscalibrated terminal selector need not be.

G.7. Stopping Certificates

Stopping rules are not needed for the core SCPS theorem, but they are useful when a controller wants to certify that little value remains under a terminal objective. The following generic statement is included only as a template: it is meaningful when the implementation can actually upper-bound remaining gain.

Let T_t be any terminal value on realized portfolios, and let

$$V_t^\star := \sup_\Pi \mathbb{E}\big[T_t(S_{\text{final}}^\Pi) \mid I_t\big]$$

be the best expected terminal value achievable from the current state under the remaining budget.

Proposition G.7 (generic stopping certificate). Suppose R_t is an upper bound on the best additional gain still achievable, meaning that for every continuation policy Π,

$$\mathbb{E}\big[T_t(S_{\text{final}}^\Pi) - T_t(S_t) \mid I_t\big] \le R_t.$$

Then

$$V_t^\star \le T_t(S_t) + R_t.$$

In particular, if $R_t \le \varepsilon$, stopping now is ε-optimal for terminal value T_t.

Proof. For every continuation policy Π,

$$\begin{aligned}\mathbb{E}[T_t(S_{\text{final}}^\Pi) \mid I_t] &= T_t(S_t) + \mathbb{E}[T_t(S_{\text{final}}^\Pi) - T_t(S_t) \mid I_t] \\ &\le T_t(S_t) + R_t.\end{aligned}$$

Taking the supremum over Π gives the result. □

G.8. Closed-Loop Reporting Checklist

When SCPS is deployed as a controller rather than evaluated as a frozen one-round objective, reports should keep the following quantities separate:

1. Frontier capacity: what support was available if the frontier were exhausted?
2. Search acquisition: what support did the controller actually acquire under budget?
3. Frozen coverage: what posterior or evaluator-induced mass did the portfolio cover under the frozen objective?
4. Truth-coupled hit/oracle coverage: did the portfolio contain the correct answer or target in hindsight?
5. Terminal selection: did the final selector emit the correct or preferred item?
6. Abstention and wrong-preference modes: when selection fails, did it abstain, choose an unsupported item, or prefer the wrong represented item?

Closed-loop boundary

Frozen semantic coverage and its batch-separable one-round lift have exact diminishing-returns guarantees. Arbitrary belief updates, correlated observations, mutable action outcomes, and terminal selectors require additional modeling assumptions and should be evaluated as separate layers.

H. Empirical Diagnostics and Reproducibility Details

The empirical results are reported with the same layer separation used in the theory. The live MMLU-Pro experiment evaluates portfolio construction: whether search discovers answer classes that were not already represented. Selector accuracy is a separate object. This appendix records the protocol and the evidence-selection rule used in the main text.

> **Empirical claim boundary**
>
> The main result is a live exact-answer MMLU-Pro portfolio-construction result. It is not a claim that terminal selection is solved or that open-ended answer-bank construction is ready.

H.1. Evidence Selection

Table 6 states which empirical sources are promoted to paper claims and which are used only as scope or diagnostic evidence.

Evidence source	Paper status	Use in the manuscript
Final live MMLU-Pro evaluation	Main evidence	Supports the claim that exact-answer SCPS constructs higher Pass@16 portfolios than live tree-prefix and semantic-pruning baselines on a disjoint 256-question evaluation set.
Live ablations and cost readout	Supporting context	Interpret the selected configuration and report realized resource use without changing the promoted claim.
Selector diagnostics	Limitation	Records the terminal-selection bottleneck separately from portfolio construction.

Table 6. The experimental narrative promotes only the evidence that matches the claim layer. Diagnostic quantities are reported separately from the live portfolio-construction claim.

H.2. Live MMLU-Pro Exact-Bank Protocol and Surrogate Training

The promoted live evidence uses a 256-question held-out MMLU-Pro exact-bank slice selected with seed 47 from the MMLU-Pro test split. Each question has a finite multiple-choice answer bank, so answer coverage can be evaluated exactly after canonicalizing model outputs to option labels. The selection rule excludes the 512-question controller-training split as well as all questions used for pilot analyses, tuning, or auxiliary diagnostics, so the held-out questions are disjoint from the data used to train or tune the learned SCPS surrogate bundle. The exact question identifiers, split construction, training-data provenance, and exclusion summary are included in the anonymized audit artifact described in Appendix H.6.

Model and decoding. All live generations use Qwen3.5-9B (Team, 2026) served through a native vLLM backend (Kwon et al., 2023). The prompt format is an answer-only multiple-choice prompt, with canonicalization under the MCQ option-bank evaluator. The compared methods share the same live-controller family, questions, prompt style, answer canonicalizer, hard token limits, and terminal-option scoring protocol. Realized decode, auxiliary, total token cost, rollout counts, and budget violations are reported for each method.

Methods. The selected SCPS configuration uses exact option-bank memory, answer-diverse action generation, and a learned surrogate bundle trained from the disjoint controller-training split. The bundle supplies answer-distribution, length, and collision/overlap signals; it is a deployed surrogate, not a theorem-level lower bound and not a final-selector model. The controller uses only pre-execution decision-local signals: answer support, proposal target or predicted answer features, novelty/nonredundancy features, terminal-option log-probability scores computed before terminal execution, seen-answer state, and cost/budget estimates. Tree-prefix expansion is a ToT-style frontier-expansion baseline that extends active prefixes without residual-support or overlap scoring. Semantic pruning is a redundancy-control baseline that removes or collapses duplicate semantic states without optimizing the Bonferroni residual-support objective. The baselines share the same split and hard limits; their realized token counts differ because the policies allocate and terminate rollouts differently.

Surrogate training interface. The deployed estimates in Equation (7) are derived from the trained heads in Table 9: answer-distribution predictions define residual novelty against the current option bank, pairwise collision predictions define overlap, and length predictions define the cost term. The controller-training split supplies labels by canonicalizing deployment-kernel completions to option or null outcomes, recording pairwise same-answer collisions, and logging completion lengths.

Method family	Shared access	Distinguishing rule
SCPS	Split, prompt, canonicalizer, option bank, hard budgets, terminal-option scores, and learned surrogate bundle.	Scores actions by estimated residual support, estimated overlap, and cost; the selected root16 variant uses answer-diverse root proposals with target 12 and answer-family quota 8.
Tree-prefix	Same split, prompt, canonicalizer, hard budgets, terminal-option scores, and learned surrogate bundle.	Expands ranked frontier prefixes without the residual-support/overlap objective and without semantic pruning.
Semantic pruning	Same split, prompt, canonicalizer, hard budgets, terminal-option scores, and learned surrogate bundle.	Skips prefixes whose predicted answer is already seen or whose pairwise overlap with selected prefixes exceeds 0.75, replacing them by continuing down the ranked feasible list when possible.
Self-consistency and semantic diversity	Same split, prompt, canonicalizer, model, temperature, rollout cap, and hard budgets.	Sample root completions directly; semantic diversity selects unique answer classes first, then fills remaining slots, without the SCPS receding-horizon allocation score.

Table 7. The baseline comparison separates shared evaluator access from the policy difference under test. The tree baselines have the same predictor bundle and evaluation conventions as SCPS, but not the residual-support allocation objective.

Setting	Value
Held-out split	MMLU-Pro test split, 256 questions, held-out seed 47, zero overlap with controller-training, validation, pilot, tuning, and auxiliary diagnostic question IDs.
Run configuration	Seed 13, Qwen3.5-9B, vLLM native, bfloat16, temperature 0.6, answer-only MCQ prompt, MCQ canonicalizer.
Budgets	Maximum 16 rollouts, max output 192 tokens per completion, decode-token limit 3072, total-token limit 30000, no budget violations in the reported rows.
Selected SCPS policy	Exact option-bank memory, batch size 1, 10 planning rounds, root proposal batch 16, distinct-source-prefix target 12, terminal answer-class quota 8.
SCPS score coefficients	Discovery weight 1.0, overlap penalty 0.5, certification weight 0.0, repeat penalty 0.25, seen-answer penalty 1.0.

Table 8. Deployed controller settings are fixed before the promoted hold-out comparison. These settings identify the exact root16 SCPS variant used in the main table.

Metrics. Pass@N is 1 if any candidate in the generated portfolio canonicalizes to the gold option. This is the exact-answer portfolio-coverage metric; Appendix G also calls the same quantity truth-coupled hit/oracle coverage when distinguishing it from terminal selection. Common selector accuracy is the accuracy of the shared practical verifier/selector. Majority accuracy chooses the most frequent canonicalized answer class. Unique answers counts distinct canonicalized answer classes per portfolio. Duplicate-token fraction measures the fraction of generated token mass assigned to duplicate answer classes. Mean token counts are realized totals. Selector metrics are reported to diagnose the selection bottleneck.

H.3. Live MMLU-Pro Results

Table 10 reports the final live comparison on the 256-question held-out evaluation set. Auxiliary checks are used only as diagnostics and are not pooled into this final live claim.

Table 11 reports paired Pass@16 deltas for the final live comparison against tree-prefix and semantic pruning.

The confidence intervals in Table 11 are paired intervals over the 256 held-out questions. They do not include variation from generation randomness, controller-training randomness, surrogate-training seeds, or model-serving nondeterminism. All reported rows use the same held-out split seed 47 and

Component	Target / model	Rows	Validation readout
Value head	Bounded-value MLP over pooled Qwen features; target is realized candidate value.	1636 train, 204 validation	Brier 0.256, ECE 0.223, Spearman 0.256, lower-bound coverage 0.887.
Answer distribution	MLP over pooled features plus decision-local metadata; target is answer histogram plus null mass.	1636 train, 204 validation	Mean $L_1 = 1.691$, nonnull-mass RMSE 0.098, top-1 agreement 0.147.
Length/cost	Survival-length MLP over pooled features; target is first valid-answer length.	1636 train, 204 validation	RMSE 23.017, Spearman 0.597, cap-hit ECE 0.006.
Collision/overlap	Pairwise MLP over paired pooled features; target is answer collision.	2454 train, 306 validation	Brier 0.102, ECE 0.048, RMSE 0.319.
Calibration	Quantile-regression conformal lower-bound calibration.	Disjoint calibration rows	Miscoverage level 0.1; deployed value target is lower bound.

Table 9. The learned surrogate bundle is specified as a deployed planning aid, not a theorem-level oracle. All components are trained away from the 256-question evaluation split.

Method	Pass@16 ↑	Mean tokens ↓	Unique answers ↑	Dup. token frac. ↓
Exact-answer SCPS (selected configuration)	233/256 (91.0%)	1120.25	3.469	55.4%
Tree-prefix	192/256 (75.0%)	3546.10	1.766	81.5%
Semantic pruning	190/256 (74.2%)	3548.08	1.770	81.3%

Table 10. The selected SCPS configuration is the promoted live portfolio-construction result. Pass@16 is portfolio coverage, not final selected accuracy. Counts are retained here for auditability. Bold marks the promoted portfolio/resource metrics.

Comparison	Delta (pp)	95% CI (pp)	SCPS-only wins	Baseline-only wins
SCPS minus tree-prefix	+16.0	[10.6, 21.5]	49	8
SCPS minus semantic pruning	+16.8	[11.3, 22.7]	52	9

Table 11. The promoted paired comparisons are live comparisons against the two tree-search baselines. Intervals are approximate paired 95% intervals for Pass@16 deltas on the same questions; raw paired counts are shown in the wins columns.

run seed 13; completed promoted baseline runs were not discarded from the headline comparison.

H.4. Live MMLU-Pro Ablations and Cost Readout

The ablation evidence is used to interpret the selected live configuration, not to construct a post-hoc tuning hierarchy. Auxiliary live checks remain diagnostic context; the main paper promotes only the final comparison in Table 10.

The ablations test whether the answer-diverse targeted action space and the deployed novelty–overlap–cost score matter. The non-targeted root-sampling variant removes targeted answer-diverse action generation and reaches 82.4% Pass@16. Targeted or answer-diverse variants remain between 87.5% and 91.0% Pass@16. Removing the explicit overlap penalty or the score cost term from the promoted root16 configuration reaches 90.2% and 89.1%, respectively. Several paired intervals overlap, so we treat smaller between-variant differences as diagnostic rather than decisive tuning evidence.

The answer-diverse action generator is an important part of the empirical system. A same-held-out diagnostic packet that removes targeted answer-diverse action generation reaches 211/256 (82.4%) Pass@16, while the targeted or answer-diverse SCPS variants in Tables 13 and 14 remain between 87.5% and 91.0%. We therefore interpret the experiment as evidence for the deployed SCPS system and its answer-diverse action space together, not as proof that the scoring objective alone explains the full gain.

The result is not an equal-realized-token claim. The methods share matched live protocols and hard limits, but realized costs differ. On the 256-question evaluation set, SCPS uses 1120.25 mean total tokens versus 3546.10 for tree-prefix and 3548.08 for semantic pruning. We therefore phrase the evidence as a live portfolio-construction result with reported realized costs, not as a claim that all methods spent exactly the same number of tokens.

Variant	Residual support	Overlap term	Score cost	Pass@16 ↑	Mean tokens ↓	What it tests
Full SCPS	yes	yes	yes	233/256 (91.0%)	1120	Full controller.
No targeted action generation	weak	yes	yes	211/256 (82.4%)	–	Whether answer-diverse actions matter.
No explicit overlap penalty	yes	no	yes	231/256 (90.2%)	1123	Whether duplicate-risk scoring matters.
No score cost term	yes	yes	no	228/256 (89.1%)	1125	Whether the score cost term matters.
Semantic pruning baseline	partial	heuristic	budget only	190/256 (74.2%)	3548	Pruning without the portfolio objective.

Table 12. Ablations diagnose the deployed SCPS design; they are not a decisive variant hierarchy. All rows use the same 256-question held-out split and Pass@16 portfolio-coverage metric. The no-overlap row removes the explicit pairwise overlap penalty from the score while leaving other redundancy controls in place; the no-cost row removes the score cost term while retaining hard rollout and token budgets. Semantic pruning uses heuristic redundancy filters rather than the residual-support portfolio objective, and the non-targeted diagnostic did not preserve a comparable realized-token readout.

Method	Pass@16 ↑	Mean tokens ↓	Unique answers ↑	Dup. token frac. ↓	Role
SCPS root16 / target12 / quota8	233/256 (91.0%)	1120.25	3.469	55.4%	Promoted SCPS configuration.
SCPS root16 no explicit overlap penalty	231/256 (90.2%)	1122.96	3.395	56.6%	Score ablation; other redundancy controls retained.
SCPS root16 no score cost term	228/256 (89.1%)	1124.76	3.488	55.0%	Score ablation; hard budgets retained.
SCPS root12 / target10 / quota6	226/256 (88.3%)	1113.88	3.402	54.3%	Proposal-cap ablation.
SCPS anchor target-faithful calguard	228/256 (89.1%)	1110.22	3.367	54.5%	Prior targeted-action anchor.
SCPS answer-diverse no calguard	224/256 (87.5%)	1113.80	3.352	55.3%	Guard ablation.
Self-consistency r16	200/256 (78.1%)	316.13	2.555	84.0%	Independent root sampling.
Semantic diversity r16	202/256 (78.9%)	316.13	2.570	83.9%	Root answer-diversity baseline.
Tree-prefix	192/256 (75.0%)	3546.10	1.766	81.5%	Tree-search baseline.
Semantic pruning	190/256 (74.2%)	3548.08	1.770	81.3%	Redundancy-control tree baseline.

Table 13. Expanded baselines and ablations show that SCPS is not only compared against duplicate-heavy tree policies. The independent root-sampling baselines improve answer diversity over the tree baselines but remain below the selected SCPS portfolio coverage on this held-out slice. The no-overlap and no-cost rows ablate score terms while retaining the same hard budgets.

Variant	Pass@16	Delta vs. selected (pp)	95% CI (pp)	Interpretation
Selected answer-diverse configuration	91.0%	–	–	Promoted live variant.
No explicit overlap penalty	90.2%	−0.8	$[-3.1, 1.2]$	Removes the explicit pairwise overlap term; other redundancy controls remain.
No score cost term	89.1%	−2.0	$[-4.7, 0.8]$	Removes the score cost term; hard budgets remain.
Lower proposal-cap variant	88.3%	−2.7	$[-5.9, 0.4]$	Proposal-cap ablation; interval overlaps zero.
Guarded targeted-action variant	89.1%	−2.0	$[-5.1, 1.2]$	Targeted-action ablation; interval overlaps zero.
Answer-diverse no-guard variant	87.5%	−3.5	$[-6.6, -0.4]$	Guard diagnostic; lower in this paired readout.

Table 14. SCPS ablations diagnose the deployed answer-diverse system design, not a decisive variant hierarchy. Deltas are paired differences relative to the selected SCPS configuration. Confidence intervals are paired bootstrap intervals over the same held-out questions.

H.5. Selector Diagnostics

The main limitation is selector conversion. On the 256-question evaluation set, SCPS reaches 91.0% Pass@16, while common verifier accuracy is 41.8% and majority accuracy is 65.2%. The common verifier and majority rule do not convert all discovered support into selected answers. This pattern supports the paper's main limitation: improving the search portfolio does not by itself solve terminal selection.

Method	Common selector ↑	Majority ↑
Exact-answer SCPS (selected configuration)	41.8%	65.2%
Tree-prefix	43.0%	60.2%
Semantic pruning	43.4%	62.5%

Table 15. Selector metrics diagnose conversion, not portfolio construction. Results use the same live MMLU-Pro portfolios as Table 10. These columns are not promoted as terminal answer-selection claims.

H.6. Reproducibility Details

The submission includes a single anonymized supplement ZIP with two directories: scps_anonymous_artifact/ and scps_code_artifact/. The data artifact contains the held-out question identifiers and gold option labels, sanitized method settings, the prompt template, a sanitized surrogate-bundle summary, per-question selector outcomes, compact canonicalized answer-class portfolios, paired delta tables, and a script that recomputes the reported rates from the CSV files. The code artifact contains the core runnable portfolio_search package, sanitized paper-specific configs, a table-recomputation utility that reads the sibling data artifact, and smoke tests for the released configuration surface. Together, these artifacts support auditing the reported tables and exercising the core implementation, but they do not provide full fresh live-rerun reproducibility. Fresh live reruns additionally require the public benchmark, Qwen3.5-9B access, a vLLM-compatible serving stack, and a compatible trained predictor bundle supplied by the user. The submission package does not include raw generation text, full frontier diagnostics, trained predictor weights, non-anonymous run manifests, machine identifiers, user paths, or model-cache paths.

The main live run used Qwen/Qwen3.5-9B served with vLLM native in bfloat16 on a single CUDA device (tensor_parallel_size=1, max model length 16384). Mean question wall-clock was 56.05 seconds, mean total tokens 1120.25, mean decode tokens 836.12, and mean generation speed 145.84 tokens/s. The available logs record only a generic CUDA device identifier, not a portable GPU model name; we therefore do not name a GPU model. The learned SCPS surrogate bundle was trained from the 512-question controller-training split, and the evaluation-set manifest excludes that split as well as all questions used for pilot analyses, tuning, or auxiliary diagnostics.

A full post-review archive should additionally include:

1. exact package lockfiles and hardware logs for the live serving environment;
2. raw per-question generations and full frontier diagnostics;
3. model, tokenizer, serving backend, package versions, and hardware records;
4. execution logs sufficient to reproduce realized token counts and wall-clock summaries.

Assets and licenses. The main experiment uses MMLU-Pro and Qwen3.5-9B. The MMLU-Pro dataset card lists an MIT license, and the Qwen3.5-9B model card lists Apache-2.0. The implementation uses vLLM for serving. The supplement includes license and third-party notice files and does not redistribute benchmark question text, model weights, tokenizer files, or serving binaries.

Anonymity. The paper text intentionally avoids local filesystem paths, personal repositories, acknowledgments, and non-anonymous project URLs. Any supplementary code or data package must be built as a fresh anonymous archive, excluding logs or metadata that reveal usernames, machine identifiers, repository owners, or cluster paths.

Empirical reporting principle

A portfolio-search experiment is strongest when it reports support found by the search procedure separately from the accuracy of the terminal selector that consumes that support.