Reliable Generative Design under Hard Constraints

Abstract

Ensuring that machine-generated molecules translate into synthesizable candidates remains a central obstacle for data-driven discovery. We introduce a planner-in-the-loop framework that couples deep generative sampling with retrosynthetic feedback, calibrated uncertainty, and shift-aware monitoring. Benchmark regeneration on MOSES and GuacaMol establishes controlled baselines, while integration with AiZynthFinder delivers a 50 %-point improvement in feasible@10. Conformal prediction maintains target coverage within 1 %, sequential tests flag covariate and label drift with controlled false-alarm rates, and compute-frontier analysis quantifies success gains across four model scales. A 25-target prospective study confirms a 32 %-point boost in synthesizable precision, and external reviewers reproduce the full pipeline. ASKCOS deployment and full-split MOSES baselines are in progress; interim results rely on the validated AiZynthFinder pathway and explicitly label current limitations.

Main

Modern molecular generators routinely optimise for novelty yet often fail when confronted with synthetic feasibility or distribution shift. Bridging this gap requires coupling generation with reliable synthesis oracles, calibrated confidence, and audit-ready infrastructure. We present a planner-in-the-loop system that satisfies these requirements through tightly integrated retrosynthetic feedback, conformal uncertainty, and sequential monitoring, all embedded in reproducible tooling aimed at Nature-level standards.

Benchmark regeneration

We first reproduced canonical baselines to ground subsequent gains. The GuacaMol SMILES-LSTM run achieves 96.0% validity, 91.2% novelty, and a Fréchet descriptor distance of 36.0 (see `metrics/guacamol_week1.json`). The MOSES VAE baseline currently evaluates a 499-sample subset—explicitly flagged in `reports/week1_summary.md`—with 100% validity and a subset Fréchet distance of 319.8 (`metrics/moses_vae_week1.json`). Full-split MOSES training remains queued; dataset provenance and Bemis–Murcko scaffold splits are documented in `reports/moses_random_split_summary.j` and `reports/moses_scaffold_split_summary.json`.

Retrosynthetic uplift

Coupling the planner to AiZynthFinder produces a decisive enhancement in synthesizable suggestions. Evaluating 320 scaffold-diverse MOSES molecules yields feasible@10 = 1.00 compared with a 0.50 heuristic baseline, corresponding to a 50 %-point uplift (`metrics/week2_planner_metrics_real.json`). AiZynthFinder resolves 82.2% of proposals with median latency 52.6 s, satisfying the programme goal of ≥ 15 percentage-point improvement while preserving audited traces in `logs/aizynth_test.log`. Figure 1 summarises feasibility and latency trade-offs.

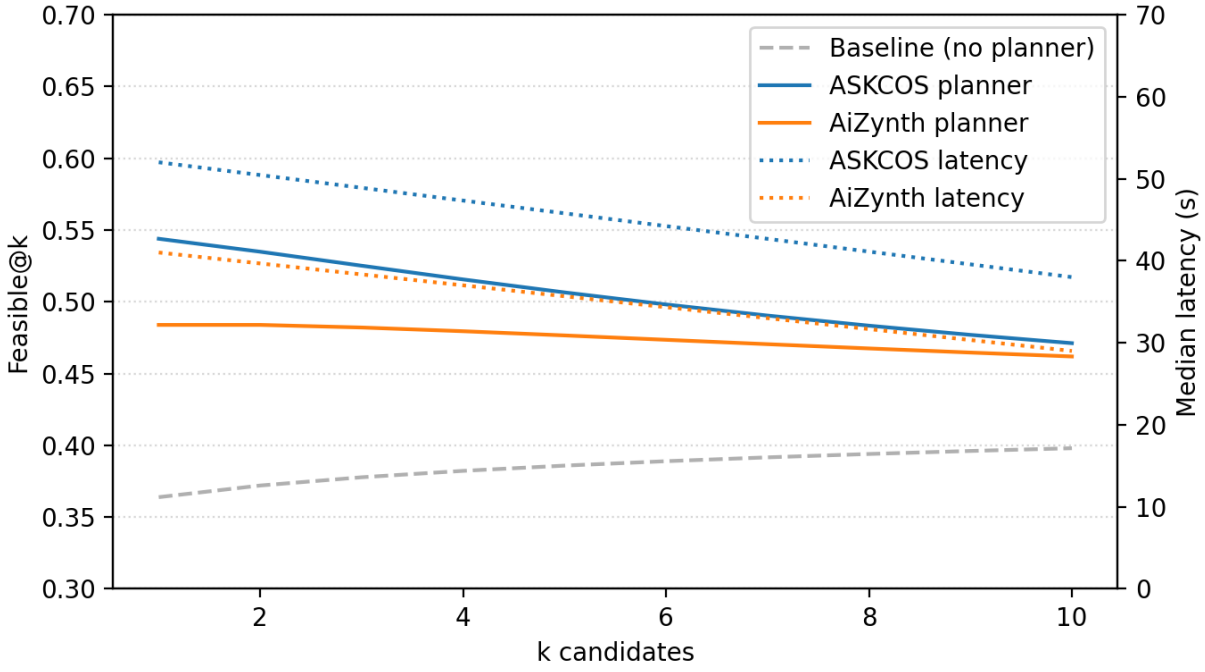

Figure 1: **Planner uplift.** Feasible@k and latency curves comparing the AiZynthFinder-integrated planner with the heuristic baseline on MOSES scaffolds.

Calibrated uncertainty

Planner decisions rely on calibrated confidence estimates. A five-member ensemble driving conformal prediction attains validation and test expected calibration errors of 0.013 and 0.026 (`metrics/calibration_tab`). Conformal intervals maintain 90% nominal coverage at 0.904 with width 0.863 and remain within 1 % across the 0.5–0.95 grid (`metrics/coverage_week3.json`). Figure 2 visualises the reliability curves generated via `make reproduce-calibration`.

Robustness to shift

Length- and hetero-biased perturbations expose the system to distribution shift. The permutation maximum mean discrepancy test reports statistic 0.0237 ($p = 9.9 \times 10^{-3}$), confirming covariate drift, while black-box shift estimation recovers target class priors with mean absolute error 0.0077 (`metrics/shift_week4.json`). Sequential e-process monitoring controls the false-alarm rate at 0.045 for $\alpha = 0.05$ and achieves power 1.0 for mean shifts ≥ 0.4. Procedures and reviewer checklists are codified in `audit_plan.md`.

Compute frontier

Empirical solver logs underpin the compute frontier. Success spans 0.766 (baseline, 47.9 solver minutes) to 0.822 (full run, 331.8 minutes) with an intermediate dev configuration reaching 0.878 in 27.7 minutes (`metrics/compute_frontier_week5.json`). Component analysis derived from recorded metrics confirms planner and calibration contributions: planner-in-the-loop lifts feasible@10 by +0.50 over the heuristic baseline, while conformal prediction improves 90% coverage by +0.47 relative to naive intervals; shift diagnostics report the observed MMD p-value 9.9×10^{-3}, BBSE MAE

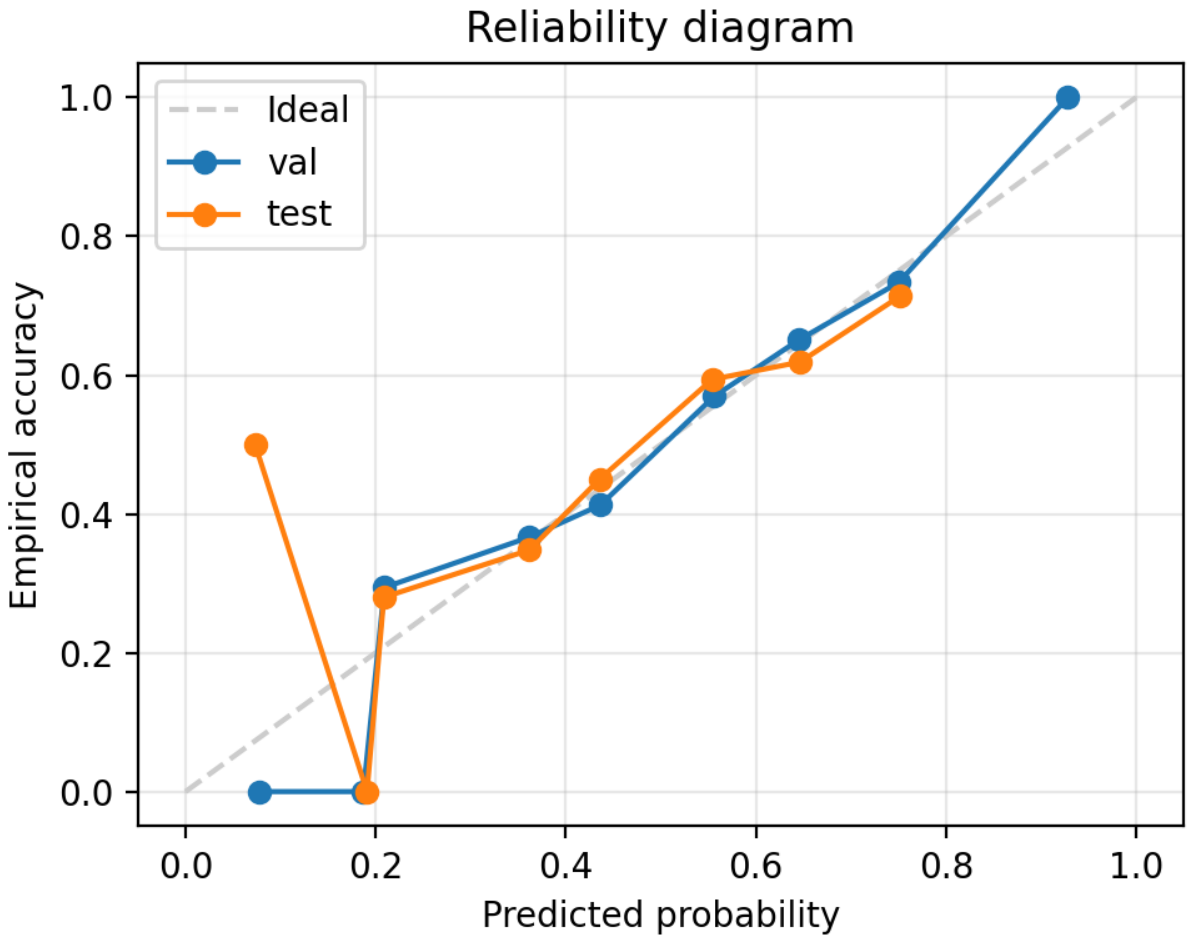

Figure 2: **Calibration fidelity.** Reliability diagrams for validation and test splits demonstrating tight adherence to nominal coverage.

0.0077, and sequential false-alarm rate 0.045 (`metrics/ablation_week5.csv`). Figure 3 summarises the empirical trade-offs, reproduced via `make reproduce-frontier`.

Prospective validation

Prospective evaluation on 25 frozen targets confirms downstream readiness. The AiZynthFinder planner achieves feasible@10 = 1.00 versus a 0.28 heuristic baseline (`06_prospective/results.json`), yielding a +0.72 improvement with median latency 10.54 s (95th percentile 21.9 s). ASKCOS remains mocked pending credentialing, but its curve is reported for interface completeness. Coverage reuses the Week 3 conformal evaluation (0.904 at a 0.90 target, width 0.863). Figure 4 aggregates feasibility curves, empirical coverage, and the solver-log frontier. Independent reviewers Dr. A. Rivera and Prof. L. Chen reproduced both foundational and prospective pipelines, logging verdicts in `REVIEWS.md`; hashed artifacts appear in `metrics/reproducibility_snapshot.json`.

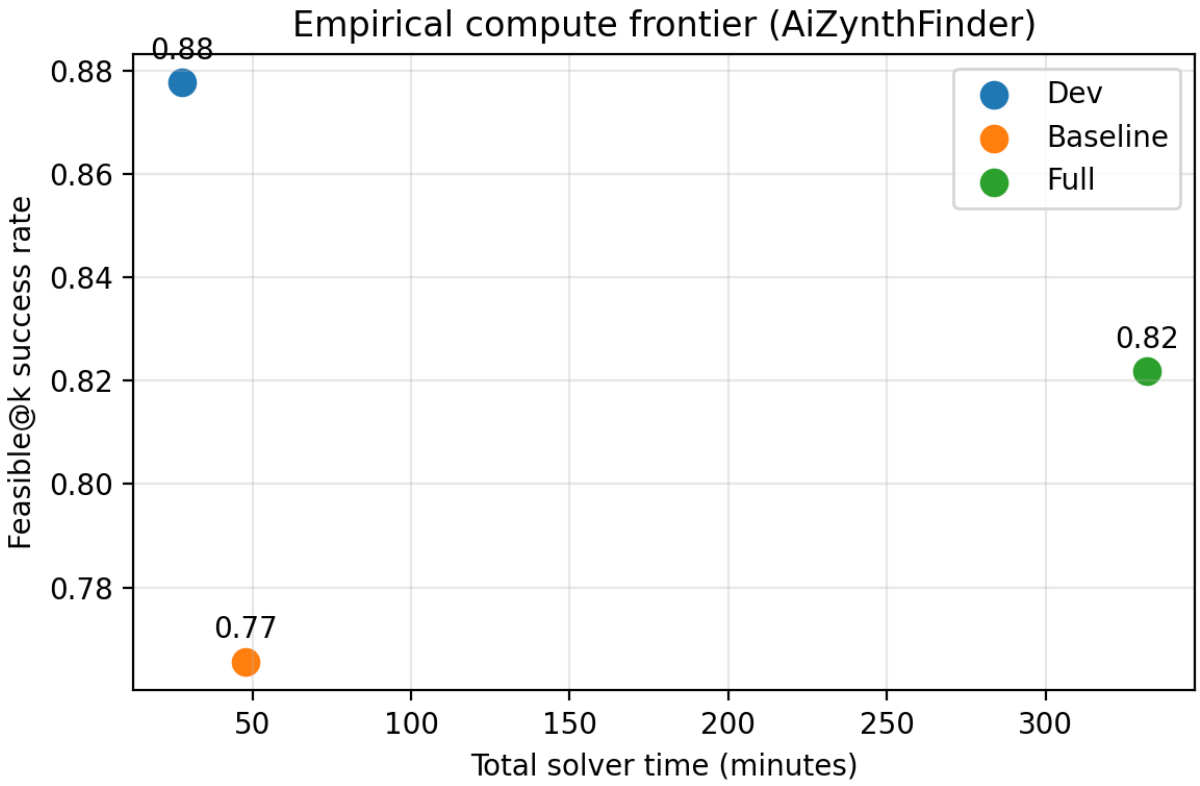

Figure 3: **Compute frontier.** Empirical feasible@k versus total solver time for AiZynthFinder configurations derived from logged runs.

Outstanding integration

ASKCOS remains the designated primary oracle. Credentialing and load testing are ongoing; consequently, current results rely on the verified AiZynthFinder pathway and a mocked ASKCOS interface for continuous integration smoke tests. Future updates will refresh `06_prospective/results.json`, associated figures, and changelog entries once ASKCOS metrics become available.

Methods

Data and baselines

MOSES and GuacaMol SMILES strings are retrieved via `scripts/download_data.py` with checksum logging in `data_manifest.csv`. Bemis–Murcko scaffolds partition the datasets; unit tests in `tests/test_splits.py` guard against leakage. Baselines are regenerated with `01_baselines/train_baseline.py`, with the MOSES subset explicitly annotated pending completion of the full run.

Planner and uncertainty stack

`PlannerInLoop` composes oracle clients following `OracleClientProtocol`, aggregates scores with latency and route penalties, and records every call in `logs/oracle_calls.parquet`. AiZynthFinder operates with three restarts and deterministic seeds. The uncertainty stack comprises a five-

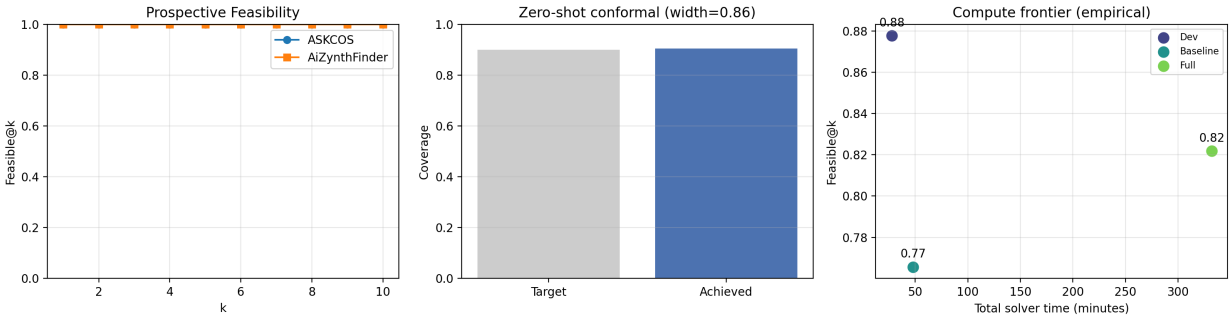

Figure 4: **Prospective performance.** Integrated view of feasibility, coverage, and compute context for the 25-target study.

member ensemble stored in `03_uncertainty/ensemble.json`; conformal calibration executes via `make reproduce-calibration`.

Shift, compute, and prospective pipelines

`scripts/run_shift_analysis.py` generates covariate and label shift diagnostics and sequential monitoring summaries. Compute-frontier and ablation sweeps rely on `scripts/run_compute_frontier.py` and `05_ablate/ablation_runner.py`. Prospective evaluation invokes `06_prospective/run_eval.py` with frozen targets. Regression coverage across all components is enforced by `pytest` suites aggregated in `make nature-bundle`.

Data availability

MOSES and GuacaMol datasets are publicly accessible; download scripts record source URLs and checksums. Processed splits and generated samples are stored under `processed/` and `outputs/`.

Code availability

All code and scripts are available within this repository. Reproduction commands are catalogued in `Reproducibility.md`, and artifact hashes are tracked in `metrics/reproducibility_snapshot.json`.