Thermostability-First QD + BO Active-Learning Pipeline for Engineering Functional PETase Variants

(Quick Draft)

Abstract

Poly(ethylene terephthalate) (PET) accumulation demands biocatalysts that are robust at elevated temperatures while retaining catalytic function. We present a thermostability-first computational framework that couples active-learning–guided directed evolution with quality-diversity (QD) maintenance and Bayesian optimization (BO) to design PET-degrading enzyme variants. Thermostability is treated as the primary objective, with maintained (or minimally reduced) activity as a secondary constraint. A calibrated surrogate predicts stability and activity proxies from sequence; uncertainty-aware acquisition selects diverse, high-value candidates across a QD archive. We begin from the FAST-PETase scaffold and, contingent on success, extend the same workflow to the industrially promising cutinase scaffold LCC-ICCG. Each round promotes a small batch (≈8–16) to high-fidelity checks—AlphaFold2 (AF2) confidence and active-site geometry, ΔΔG calculations, and docking/short MD—after which the model and archive are updated. Structural and biochemical hard constraints (e.g., catalytic triad preservation) are enforced throughout. The pipeline is designed to identify thermostable, functional variants while remaining experimentally tractable.

1. Introduction

Plastic waste mitigation motivates the development of enzymes capable of rapid PET depolymerization under industrially relevant conditions. While random mutagenesis and plain genetic algorithms can yield local improvements, rugged, epistatic fitness landscapes often induce premature convergence. We describe a principled, sample-efficient design-build-test-learn (DBTL) loop that integrates surrogate-model BO with QD to illuminate multiple optima while satisfying stringent structural constraints. In this work, we prioritize **thermostability as the majority objective** while maintaining sufficient catalytic function to remain practically useful. We **start with FAST-PETase** as our initial scaffold and, upon successful validation of the workflow, **port the pipeline to LCC-ICCG**.

2. Methods

2.1 Overview of the DBTL cycle

1. **Initialization.** Assemble an initial training set: wild-type PETase and FAST-PETase, known engineered PETase/LCC variants from literature, and a small, diverse library of single/double mutants near the active site and stability-relevant regions. Evaluate with fast proxies and train the initial surrogate.

2. **Surrogate modeling.** A deep-ensemble regressor (or GP with sequence kernels) predicts objectives —PET activity proxy and stability (ΔΔG / T_m)—and outputs calibrated uncertainty. Inputs combine residue-level encodings with protein-LM/structure features.
3. **Candidate generation with diversity maintenance.** Maintain a QD archive over behavior axes such as mutation count and predicted stability bins. Propose candidates via: (i) **local** trust-region mutations around archive elites; (ii) **LM/MPNN-guided** proposals that are plausible yet diverse; (iii) **novelty injections** subject to constraints.
4. **Uncertainty-aware scoring & acquisition.** Rank candidates using a composite objective (or Pareto ranking) and uncertainty-aware acquisition (e.g., UCB and Thompson sampling), then select a batch stratified across QD niches with explicit sequence-distance diversity controls.
5. **High-fidelity evaluation.** Promote batch members to more accurate checks: **AlphaFold2 (AF2)** predictions (pLDDT/pAE; active-site RMSD and pocket integrity), ΔΔG (Rosetta/FoldX), activity proxies (docking or short MD with PET oligomers), and solubility/expression heuristics.
6. **Update & iterate.** Add results to training data; retrain/calibrate the surrogate; refresh the archive; repeat for $\leq$3 rounds or until convergence.

2.2 Objectives and constraints

We optimize a constrained multi-objective: maximize predicted stability (ΔΔG or T_m) and maintain sufficient activity (e.g., k_cat/K_M proxy), with penalties for infeasibility. Hard constraints include: preservation of the catalytic Ser-His-Asp triad and pocket geometry; mutation-count limits per round; avoidance of problematic motifs (e.g., N-X-S/T); and structural-compatibility filters (LM likelihood / MPNN fit on the PETase backbone).

Composite scalar objective
$F(\mathrm{seq}) = w_{stab}\,\tilde{S} + w_{act}\,\tilde{A} - w_{sol}\,\tilde{X} - w_{pen}\,P(\mathrm{seq})$, with normalized objectives and large penalties for constraint violations. Alternatively, select from a surrogate-predicted Pareto front subject to $\tilde{S} \geq 0$ ($\geq$WT stability) with ϵ -constraints.

2.3 Surrogate details and uncertainty

We use deep ensembles (5× networks) for mean/variance estimates, validated via cross-validation and error–uncertainty correlation; or a GP with Hamming/subsequence kernels on a reduced site set. **Training-set curation** draws from multiple scaffolds and proven variants: WT PETase, FAST-PETase, and published high-stability/high-activity variants across PETase and LCC families (including LCC-ICCG). Features include one-hot or mutation strings, LM embeddings (e.g., ESM-2), **AF2-derived structural context** (residue pLDDT/pAE, local environment descriptors), ΔΔG, and docking/MD summaries where available. We emphasize thermostability as the primary target in loss weighting and model selection while constraining activity above a minimum threshold.

2.4 Acquisition and batch design

We combine: (i) **UCB** with an annealed exploration coefficient (high early, lower later); (ii) **Thompson sampling** to diversify; (iii) **diversity constraints** (minimum Hamming distance); and (iv) **QD-stratified picks** (top per niche). Optional: diversity-guided batch-BO objectives.

2.5 High-fidelity scoring stack

- **Structure (AlphaFold2-centric):** AF2 confidence (pLDDT), pAE, backbone/active-site RMSD to FAST-PETase; MPNN sequence-on-backbone likelihood; pocket volume and catalytic-geometry checks.
- **Stability:** Rosetta/FoldX ΔΔG; ML stability predictors; aggregation/solubility flags.
- **Activity proxies:** docking energies/contacts for PET fragments; brief MD residence-time/contact metrics.
- **Expression/production:** simple ML classifier on composition/charge; codon-optimization reserved for final constructs.

2.6 Scaffold strategy and transferability

We treat **FAST-PETase** as the launch scaffold for the full pipeline (features, constraints, and QD bins tuned to this backbone). After demonstrating success, we **transfer the trained components and design rules to LCC-ICCG** by: (i) re-embedding sequences on the LCC backbone with AF2/MPNN features; (ii) freezing generic modules (LM/structural encoders) while re-calibrating the final surrogate head on LCC literature variants; and (iii) re-initializing the QD archive with LCC-specific stability bins.

2.7 Expanded training set for the surrogate

Construct a labeled corpus including: (a) WT PETase and FAST-PETase; (b) accessible **published PETase/LCC variants with reported thermostability or activity**; (c) in-silico singles/doubles near stability hot spots with ΔΔG labels; and (d) any internal assay data as it becomes available. This broader set improves generalization and enables cross-scaffold transfer.

2.8 Model outputs (what the pipeline returns)

Artifact	Format	Generated at	Purpose	Example filename/schema
Selected sequences	FASTA / CSV	After acquisition each round	Records the round's chosen variants with IDs, mutation lists, diversity distances, constraint flags	`round_{r}_top_{k}.fasta` ; `round_{r}_selected.csv`
Structure files	PDB (AF2/ ColabFold) + JSON sidecar	After high-fidelity checks	Inspect backbone/ active-site geometry; store pLDDT/pAE; downstream scoring	`af2_{seqid}.pdb` ; `af2_{seqid}.json`

Artifact	Format	Generated at	Purpose	Example filename/schema
Stability tables	CSV / Parquet	After ΔΔG evaluation	Per-mutation and cumulative ΔΔG; learned stability score; aggregation/ solubility	`stability_round_{r}.csv`
Activity-proxy reports	CSV + optional MD log	After docking/ short MD	Capture docking scores, contacts, residence-time metrics for PET fragments	`activity_proxy_round_{r}.csv`; `md_{seqid}.log`
Acquisition logs	CSV / JSONL	During selection each round	Trace surrogate means/SDs, acquisition values (UCB/TS), decisions, trust-region state	`acq_round_{r}.jsonl`
QD archive snapshots	JSON (+ PNG heatmap)	End of each iteration and end of round	Persist the archive of elites across mutation-count × stability bins; visualize coverage	`qd_archive_round_{r}.json`; `qd_heatmap_round_{r}.png`
Pareto fronts & metrics	CSV + PNG	End of round	Show stability vs. activity trade-off; report hit rate / best-of-N / realized EI	`pareto_round_{r}.csv`; `pareto_round_{r}.png`
Uncertainty calibration	PNG + CSV	End of round	Reliability diagram and error–uncertainty correlation for the surrogate	`calibration_round_{r}.png`; `calibration_round_{r}.csv`

Artifact	Format	Generated at	Purpose	Example filename/schema
Design rationales	Markdown / PDF	With handoff bundle	Human-readable per-candidate notes (structural context, ΔΔG decomposition, pocket checks)	`rationale_{seqid}.md`
Per-round summary report (with Activity Floor)	Markdown / PDF / HTML	End of round	Executive summary including **Activity Floor** parameter (e.g., **≥70% WT at 50–60 °C**), winners, regressions, QA	`report_round_{r}_afloor-70pct_50-60C.md`

3. Materials (software & hardware)

Software stack (recommended)

- **Python 3.10+** with **PyTorch** (CUDA build) and **scikit-learn**; optional **GPyTorch + BoTorch** for BO; **Optuna** for tuning.
- **Protein LMs / sequence models: ESM-2** (embeddings), **ProteinMPNN** (sequence-on-backbone scoring/proposals).
- **Structure prediction: AlphaFold2** or **ColabFold** (for speed) to obtain pLDDT/pAE and PDBs.
- **Stability tools: Rosetta cartesian_ddg** or **FoldX** for ΔΔG; optional ML stability predictors.
- **Docking/MD (activity proxies): AutoDock Vina** or **GNINA** for PET-fragment docking; **OpenMM** or **GROMACS** for short MD.
- **QD/EA utilities:** a light **MAP-Elites/QD** implementation (e.g., **QDpy** or custom) for archive management and stratified selection.
- **Data plumbing & viz: pandas**, **numpy**, **matplotlib/plotly**, **seaborn**, **Biopython**.
- **Reproducibility: conda/mamba** envs, **Docker** (CUDA base image) optional, **git** + **DVC** (or MLflow) for artifacts.

Hardware (minimum → recommended)

- **GPU:** ≥12 GB VRAM (min) → **24 GB+** (e.g., RTX 3090/4090) for AF2/MPNN and batched inference; multi-GPU optional for AF2 jobs.
- **CPU/RAM:** 8–12 cores, **32 GB RAM (min)** → **64–128 GB** for AF2 relaxation and Rosetta batches.
- **Storage: 1 TB NVMe SSD** (datasets, PDBs, AF2 intermediates).
- **OS:** Linux (Ubuntu 22.04/24.04), recent **NVIDIA driver** + **CUDA toolkit** matching PyTorch build.
- **Nice to have:** second GPU for parallel AF2; HPC/SLURM for bulk AF2/MD; a **conda-locked** `environment.yml` for reproducibility.

4. Risk Analysis and Mitigations

- **Model misspecification / epistasis:** Uncertainty-driven exploration, multi-site proposals, and ablations to detect overfitting.
- **Proxy gaps:** Orthogonal scorers; early injection of any experimental readouts; anchors to WT/known baselines.
- **Local optima:** QD archive across mutation-count × stability bins; LM-guided jumps; trust-region resets.
- **Distribution shift:** Mutation-radius throttling per round; explicit high-distance uncertainty priors.
- **Operational complexity:** Stepwise enablement (QD-lite stratification; manual LM proposals) with fallbacks.

5. Expected Outcomes and Evaluation

We target thermostability-first outcomes: (i) a **high-stability mutant** with activity above a pre-set functional threshold; (ii) a **balanced lead** with improved stability and near-WT activity; and (iii) a **transfer test** demonstrating that design rules learned on FAST-PETase aid discovery on LCC-ICCG. We report per-round performance, uncertainty calibration curves, diversity coverage, and ablation comparisons (hit rate, best-of-N, and realized expected improvement).

6. Reproducibility and Reporting

We provide code to: (i) train the surrogate with fixed seeds; (ii) generate/score candidates; (iii) run acquisition; (iv) export archives and evaluation logs. All figures (fitness vs. round, Pareto fronts, niche heatmaps) are auto-generated from logs. Final sequences include per-mutation rationales.

7. Discussion and Future Work

Extensions include insertion/deletion moves, explicit multi-temperature scoring, diffusion-model–based conditional proposals on the PETase backbone, and multi-scaffold co-optimization.

7.1 Appendix — QD archive and BO mechanics

Quality–Diversity archive. We discretize two behavior axes: mutation count (x) and predicted stability bin (y, e.g., binned ΔΔG or a stability score). Each cell (niche) holds an elite: the best candidate assigned to that niche under hard constraints. Update rule: map candidate to a niche; insert if empty; otherwise replace only if its composite score is higher. Keeping a small top-k per niche helps recombination and calibration.

Why QD helps. It preserves stepping-stones at different mutation radii, prevents mode collapse when the optimizer exploits one basin, and enables stratified batch picks across niches to guarantee sequence diversity.

Bayesian optimization details. The surrogate provides a mean and standard deviation for each candidate. We use: (A) Upper Confidence Bound, $\mathrm{UCB}_t(x) = \hat{f}(x) + \beta_t \, \hat{\sigma}(x)$, with β_t annealed from roughly 2.0 (exploration) to 0.5 (exploitation). (B) Thompson Sampling: draw a sample score from the predictive

distribution per candidate and rank by that sample, which naturally explores high-uncertainty regions. Batching enforces a minimum Hamming distance between chosen sequences and allocates picks across QD niches. Trust regions around current elites expand on success and shrink on stagnation.