QPU-PEFT: FEEDBACK-DRIVEN CONFIGURATION SELECTION WITH RELIABILITY-GATED ISING RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

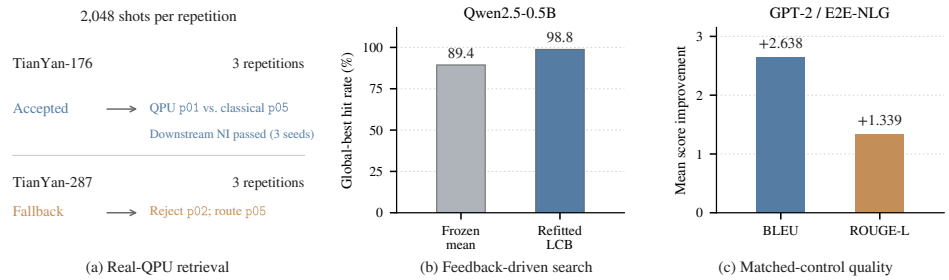
Parameter-efficient fine-tuning enables large language models to adapt to downstream tasks with a small number of task-specific parameters. Quantum-inspired unitary parameterizations offer compact representations of update directions, enabling methods such as Quantum-PEFT (Q-PEFT) to further reduce adapter parameters. However, Q-PEFT performance depends on interacting structural and training choices, yet evaluating each configuration requires costly GPU fine-tuning. This makes effective use of limited validation feedback essential for joint configuration optimization. We propose QPU-PEFT, a feedback-driven framework for budgeted Q-PEFT configuration selection. At each iteration, QPU-PEFT refits a quadratic surrogate from validation feedback, formulating its acquisition objective as a QUBO and an equivalent Ising Hamiltonian, which are used to retrieve the next configuration for GPU evaluation. We also develop a support-and-stability gate to accept QPU proposals after evaluation and otherwise invoke deterministic classical retrieval. We evaluate proposals on real QPUs. On TianYan-176, the gate accepts a hardware proposal, which differs from the deterministic classical route and satisfies a protocol-level 0.5-point BLEU/ROUGE-L non-inferiority criterion over three paired Qwen2.5-1.5B/E2E retraining seeds not used during selection; TianYan-287 separately invokes deterministic fallback. In single-seed offline replay on Qwen2.5-0.5B, Refitted Ridge-LCB reduces five-seed-mean regret by 39.8% relative to structured GP-EI and remains significantly better than RF-EI, TPE, and random search. On GPT-2/E2E-NLG, five paired full-budget retrainings show gains of 2.638 BLEU and 1.339 ROUGE-L over a learning-rate- and parameter-matched control. Finally, our separate resource-constrained selection reduces validation loss with 33.4% fewer trainable parameters and 52.4% less training time.

1 INTRODUCTION

Parameter-efficient fine-tuning (PEFT) adapts pretrained models by optimizing few task-specific parameters with a frozen backbone. Low-Rank Adaptation (LoRA) represents weight updates with low-rank factors (Hu et al., 2022). Quantum-inspired methods such as Quantum-PEFT (Q-PEFT) offer further parameter reduction through compact structured representations of update directions (Koike-Akino et al., 2025). Selecting structural and training settings requires repeated fine-tuning and validation, making exhaustive search costly even for compact adapters.

AutoPEFT applies Bayesian optimization to PEFT configuration search (Zhou et al., 2024), and factorization-machine-based optimization combines surrogates with quantum-assisted candidate selection (Kitai et al., 2020). Building on these foundations, we study Q-PEFT configuration selection under evaluation budgets and structural and parameter-budget constraints. Extending retrieval to real quantum processing units (QPUs) also requires assessing proposal reliability, since low-energy outcomes obtained from finite measurement samples may lack consistent support across runs.

We propose **QPU-PEFT**, a feedback-driven framework coupling configuration selection with constrained Ising retrieval. After each evaluation, we refit a regularized quadratic surrogate to validation outcomes, modeling configuration choices and pairwise interactions. We formulate the acquisition objective and configuration constraints as a quadratic unconstrained binary optimization (QUBO) problem and map it to an equivalent Ising Hamiltonian (Lucas, 2014). QPU proposals undergo



(a) Real-QPU retrieval (b) Feedback-driven search (c) Matched-control quality

Figure 1: QPU-PEFT results. (a) The accepted TianYan-176 proposal differs from the classical choice and passes downstream non-inferiority; TianYan-287 triggers deterministic fallback. (b) Complete-policy comparison of Frozen Ridge (mean) and Refitted Ridge-LCB on the fixed-learning-rate Qwen2.5-0.5B pool. (c) BLEU/ROUGE-L gains over a learning-rate- and parameter-matched control across five paired full-budget retrainings.

prespecified legality, repeat-support, and distribution-stability checks before acceptance, with deterministic classical fallback upon rejection.

We evaluate retrieval on two real QPUs. TianYan-176 routes `p01` instead of classical `p05`; after the selected configuration is fixed, `p01` meets prespecified 0.5-point BLEU/ROUGE-L non-inferiority margins across three paired full-budget seeds. TianYan-287 rejects `p02` and falls back to `p05`. On fixed-learning-rate Qwen2.5-0.5B, Refitted Ridge-LCB raises the global-best hit rate from 89.4% to 98.8% in paired replay. On GPT-2/E2E-NLG, five-seed retraining improves a learning-rate- and parameter-matched control by 2.638 BLEU and 1.339 ROUGE-L (Figure 1).

Our contributions are as follows.

- We propose **QPU-PEFT** for budgeted Q-PEFT configuration selection, combining feedback updates with constrained QUBO/Ising objectives; under single-seed feedback, it reduces five-seed-mean regret by 39.8% relative to structured GP-EI.
- We develop a support-and-stability gate for accepting QPU proposals after evaluation, and our design can integrate well with deterministic classical retrieval.
- We validate search quality and downstream adaptation of our method on two real QPUs.

2 Related Work

2.1 Parameter-Efficient and Quantum-Inspired Fine-Tuning

Parameter-efficient fine-tuning (PEFT) adapts pretrained models with few trainable parameters. LoRA represents updates with low-rank factors (Hu et al., 2022), AdaLoRA dynamically allocates rank across modules (Zhang et al., 2023), DyLoRA supports nested ranks (Valipour et al., 2023), DoRA separates weight magnitude and direction (Liu et al., 2024), and PiSSA initializes adapters from principal singular components (Meng et al., 2024).

Quantum-inspired methods provide alternative compact parameterizations. QuanTA constructs high-rank updates from local tensor factors (Chen et al., 2024), QuIC Adapters use Hamming-weight-preserving circuit-inspired transformations (Raj & Coyle, 2025), and Q-PEFT represents low-rank frames using compact coordinates and truncated matrix expansions (Koike-Akino et al., 2025). QPU-PEFT complements these parameterization designs by selecting interacting structural and training choices under evaluation and resource constraints.

2.2 Budgeted Configuration Selection

Hyperparameter optimization reduces the number of expensive evaluations through model-based search or adaptive resource allocation, as in TPE (Bergstra et al., 2011), Hyperband (Li et al., 2018), and BOHB (Falkner et al., 2018). BOCS fits interaction-aware surrogates over binary structures and

optimizes their acquisition objectives (Baptista & Poloczek, 2018). For PEFT, AutoPEFT uses multi-objective Bayesian optimization and studies configuration transfer across tasks (Zhou et al., 2024), while BIPEFT iteratively selects modules and ranks under a budget (Chang et al., 2024). Building on these foundations, QPU-PEFT refits its acquisition model with sequential validation feedback and compiles retrieval into QUBO/Ising objectives with explicit structural and trainable-parameter constraints, supporting reliability-gated hardware routing.

2.3 QUBO/Ising Retrieval and Quantum Hardware

Pairwise binary objectives admit equivalent QUBO and Ising representations (Lucas, 2014). QAOA samples low-energy states (Farhi et al., 2014), while surrogate-based methods combine learned models with quantum-assisted candidate selection (Kitai et al., 2020). In QPU-PEFT, QUBO encodes the feedback-derived acquisition score with feasibility and resource constraints, rather than serving as a separate quality predictor. This common objective supports exact, classical, Ising, and QPU retrieval. For finite-shot hardware, a prespecified reliability gate separates proposal generation from acceptance, invoking deterministic classical retrieval when hardware evidence is insufficient.

3 QPU-PEFT

3.1 Problem Formulation and Framework Overview

Let $\mathcal{X}$ denote a finite set of Q-PEFT configurations satisfying structural constraints and a trainable-parameter budget $P(x) \leq P_{\max}$, where $P(x)$ counts trainable parameters. Each configuration specifies structural and training choices, such as adapter rank, Taylor order, and learning rate.

For a frozen backbone W_0 and training data $\mathcal{D}_{\mathrm{tr}}$, a prespecified training procedure produces adapter parameters $\theta_x = \mathcal{A}(W_0, \mathcal{D}_{\mathrm{tr}}; x)$. We seek

$$x^\star \in \arg\min_{x \in \mathcal{X}} f(x), \qquad f(x) = \mathcal{L}_{\mathrm{val}}(W_0, \theta_x). \tag{1}$$

Each evaluation requires fine-tuning and validation, so the search is restricted to B configuration evaluations, including the initial observations.

QPU-PEFT performs sequential configuration selection from an initial observation set $\mathcal{D}_0$ (Fig. 2). At each iteration, it refits a surrogate to observed validation outcomes, compiles the updated retrieval score into a constrained QUBO/Ising objective, and retrieves an unevaluated configuration. QPU proposals undergo prespecified reliability checks, with deterministic classical fallback upon rejection. The resulting configuration x_{t+1} is evaluated and added to the observation set,

$$\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(x_{t+1}, f(x_{t+1}))\}. \tag{2}$$

After the budget is exhausted, the configuration with the lowest observed validation loss is selected. This selection and the final evaluation protocol are fixed before full-budget retraining and held-out testing.

3.2 Feedback-Driven Configuration Selection

At iteration t, QPU-PEFT uses the observations in $\mathcal{D}_t$ to guide the selection of the next configuration. The eligible candidate set contains feasible configurations not yet evaluated,

$$\mathcal{U}_t = \mathcal{X} \setminus \{x \mid (x, y) \in \mathcal{D}_t\}. \tag{3}$$

Each configuration is represented by one-hot indicators for its categorical fields. The full encoding $z(x) \in \{0,1\}^n$ includes any constraint auxiliaries, while surrogate features depend only on configuration indicators. Using the observations available in $\mathcal{D}_t$, we fit a ridge surrogate,

$$\widehat{f}_t(x) = w_t^\top \phi(z(x)), \tag{4}$$

where the prespecified feature map ϕ contains main effects and selected pairwise interactions. For mean acquisition, the quadratic retrieval score equals the surrogate prediction. The sequential QPU replay encodes LCB scores over an eight-candidate subset using a separate domain-wall representation.

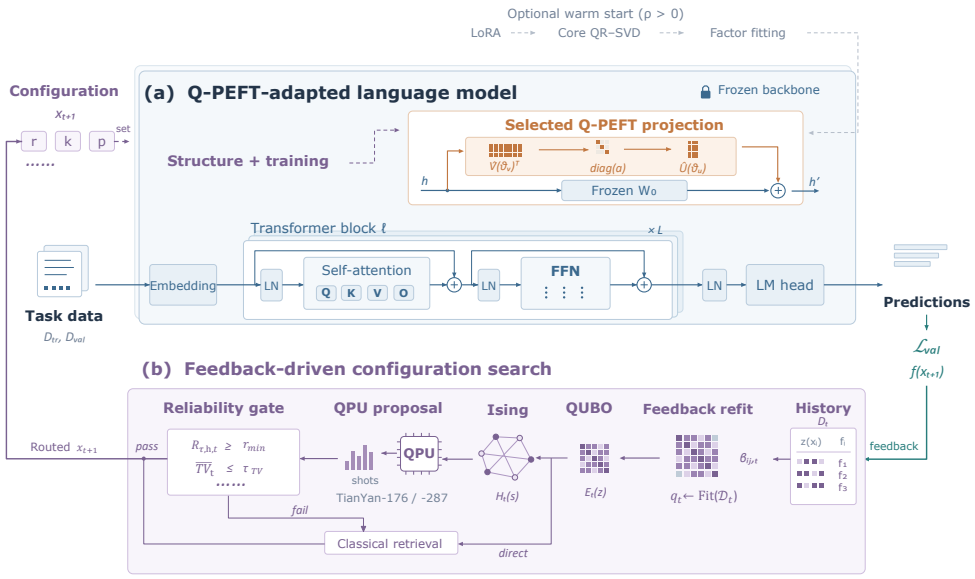


Figure 2: QPU-PEFT overview. (a) Q-PEFT with optional LoRA–QR–SVD initialization. (b) Validation feedback updates the surrogate and constrained QUBO/Ising objective. Retrieval combines reliability-gated real-QPU proposals with deterministic classical selection and fallback.

We also evaluate an optional lower-confidence-bound acquisition rule,

$$a_t(x) = \widehat{f}_t(x) - \kappa_t u_t(x), \qquad x \in \mathcal{U}_t, \tag{5}$$

where $u_t(x)$ estimates predictive uncertainty through bootstrap refits and $\kappa_t \geq 0$ controls exploration. Direct ranking minimizes the selected score over $\mathcal{U}_t$, with $\kappa_t = 0$ recovering mean-based selection. After each evaluation, Eq. (2) updates $\mathcal{D}_t$, after which the surrogate, acquisition score, and retrieval score are recomputed; the open-loop control retains its initial fit.

3.3 QUBO Formulation and Ising Mapping

At iteration t, the retrieval score is expressed as a quadratic function of the binary encoding,

$$q_t(z) = \gamma_{0,t} + \sum_{i=1}^{n} \gamma_{i,t} z_i + \sum_{i<j} \gamma_{ij,t} z_i z_j. \tag{6}$$

The linear terms capture main effects, while the pairwise terms represent interactions between encoded configuration choices. The coefficients are determined from the current feedback state and held fixed throughout the retrieval step.

Let $C(z)$ be a nonnegative quadratic penalty encoding categorical consistency, structural compatibility, and the parameter-budget constraint. It satisfies $C(z) = 0$ for legal encodings and $C(z) > 0$ otherwise. The resulting QUBO objective is

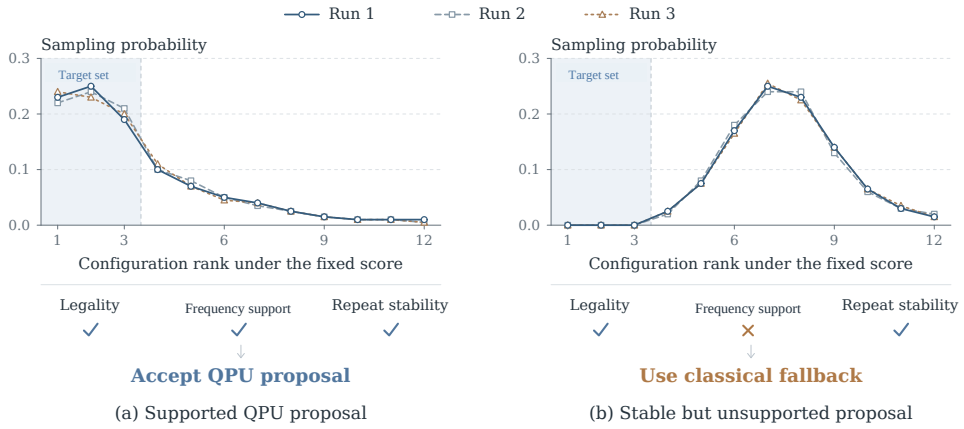
$$E_t(z) = q_t(z) + \lambda C(z), \qquad z \in \{0,1\}^n, \tag{7}$$

where $\lambda > 0$ controls the penalty strength.

The substitution $z_i = (1 - s_i)/2$, with $s_i \in \{-1, +1\}$, yields the equivalent Ising objective,

$$H_t(s) = E_t\left(\frac{1-s}{2}\right) = c_t + \sum_i h_{i,t} s_i + \sum_{i<j} J_{ij,t} s_i s_j. \tag{8}$$

For each fixed instance, exact enumeration, classical QUBO solvers, idealized Ising sampling, and real-QPU retrieval share the same objective and configuration decoder. This provides a controlled comparison of retrieval backends on a common discrete optimization problem.



(a) Supported QPU proposal

(b) Stable but unsupported proposal

Figure 3: Schematic of preregistered-target QPU gating. Curves show repeated sampling distributions over eligible configurations. Acceptance requires sufficient frequency-shortlist target support and repeat stability; otherwise, deterministic classical fallback applies. Both panels assume legal decoding.

Constraint preservation. Assume that the set of legal encodings $\mathcal{Z}_{\text{legal}}$ is nonempty. When illegal encodings exist, define

$$v_{\min} = \min_{z \notin \mathcal{Z}_{\text{legal}}} C(z) > 0, \qquad \Delta_t = \max_{z \in \{0,1\}^n} q_t(z) - \min_{z \in \{0,1\}^n} q_t(z).$$

If $\lambda v_{\min} > \Delta_t$, then

$$\arg\min_{z \in \{0,1\}^n} E_t(z) = \arg\min_{z \in \mathcal{Z}_{\text{legal}}} q_t(z). \tag{9}$$

Thus, the QUBO preserves the minimizers of the frozen retrieval score q_t over the legal set.

3.4 Reliability-Gated QPU Retrieval

QPU-PEFT treats hardware output as a proposal and applies prespecified checks on decoding validity, repeat support, and sampling stability before model evaluation. Repetition r returns a shortlist $\mathcal{S}_{h,r,t}$ containing its h most frequent legal configurations. For any nonempty support set $\mathcal{A} \subseteq \mathcal{U}_t$, define

$$R_t(\mathcal{A}) = \sum_{r=1}^{R} \mathbb{I}[\mathcal{S}_{h,r,t} \cap \mathcal{A} \neq \varnothing]\,. \tag{10}$$

The retained states and QPU proposal are

$$\mathcal{S}_t^{\text{ret}} = \bigcup_{r=1}^{R} \mathcal{S}_{h,r,t}, \qquad x_{t+1}^{\text{QPU}} \in \arg\min_{x \in \mathcal{S}_t^{\text{ret}}} q_t(x), \tag{11}$$

with deterministic tie breaking. The support set is $\mathcal{A}_t = \mathcal{T}_t$ for preregistered-target audits, where $\mathcal{T}_t \subseteq \mathcal{U}_t$ is frozen before hardware execution, and $\mathcal{A}_t = \{x_{t+1}^{\text{QPU}}\}$ for target-free sequential replay. Let L_t indicate legal and complete decoding, and let $\overline{\text{TV}}_t$ be the mean pairwise total variation distance between repeated output distributions. The gate is

$$G_t = \mathbb{I}\big[L_t = 1 \wedge R_t(\mathcal{A}_t) \geq r_{\min} \wedge \overline{\text{TV}}_t \leq \tau_{\text{TV}}\big]\,. \tag{12}$$

The next configuration is

$$x_{t+1} = \begin{cases} x_{t+1}^{\text{QPU}}, & G_t = 1, \\ x_{t+1}^{\text{classical}}, & G_t = 0, \end{cases} \tag{13}$$

Here $x_{t+1}^{\text{classical}} \in \mathcal{U}_t$ is supplied by a deterministic classical rule. The objective, decoder, shortlist size, thresholds, support rule, and classical candidate are fixed before inspecting the corresponding hardware outcomes.

Gate-conditioned selection bound. Let $x_t^\star \in \arg\min_{x\in\mathcal{U}_t} f(x)$ and suppose $\sup_{x\in\mathcal{U}_t} |f(x) - q_t(x)| \le \epsilon_t$. Define

$$\Delta_{\mathcal{A},t} = \max_{x\in\mathcal{A}_t} q_t(x) - \min_{x\in\mathcal{U}_t} q_t(x), \qquad \delta_t^{\mathrm{C}} = q_t(x_{t+1}^{\mathrm{classical}}) - \min_{x\in\mathcal{U}_t} q_t(x).$$

Then

$$f(x_{t+1}) - f(x_t^\star) \le 2\epsilon_t + G_t\Delta_{\mathcal{A},t} + (1-G_t)\delta_t^{\mathrm{C}}. \tag{14}$$

For an exact-score Top-K target, $\Delta_{\mathcal{A},t} = q_t^{(K)} - q_t^{(1)}$; for target-free sequential routing, it is the score gap of the repeat-supported QPU proposal. An exact classical fallback has $\delta_t^{\mathrm{C}} = 0$. Complete assumptions and proof are provided in Appendix A.

3.5 Optional Warm-Start Transfer

Warm-start transfer is an optional configuration choice for task-adapted Q-PEFT initialization. When enabled, a temporary rank-r_w LoRA adapter is trained for the first fraction ρ of the candidate's training schedule. Thin QR factorizations of its low-rank factors reduce the SVD to an $r_w \times r_w$ core. The leading singular components serve as fitting targets for rank-r truncated-Taylor Q-PEFT, without materializing the dense weight update.

After factor fitting, the temporary LoRA adapter and its optimizer state are discarded. Q-PEFT is trained for the remaining schedule with a newly initialized optimizer. Direct Q-PEFT trains the Q-PEFT adapter throughout the schedule. The QR–SVD derivation and factor-fitting details are provided in Appendix A.

4 Experiments

4.1 Experimental Setup

Benchmarks and models. We evaluate QPU-PEFT primarily on E2E-NLG (Novikova et al., 2017) with additional evaluations on WebNLG (Gardent et al., 2017), SST-2 (Socher et al., 2013), DART (Nan et al., 2021), and CommonGen (Lin et al., 2020). We use GPT-2 (Radford et al., 2019), Qwen2.5-0.5B, and Qwen2.5-1.5B (Qwen Team, 2024) as backbones. Candidate spaces cover structural and training settings, with fixed-learning-rate Qwen pools used to isolate structural selection. Experimental settings and supplementary results are provided in Appendices B and C.

Baselines. We compare feedback-refitted Ridge with frozen Ridge, structured GP-EI (Jones et al., 1998), random-forest RF-EI/RF-LCB (Breiman, 2001), categorical TPE (Bergstra et al., 2011), random search (Bergstra & Bengio, 2012), and SMAC on categorical candidate identifiers (SMAC-ID) (Hutter et al., 2011). RF-EI independently implements SMAC-style expected improvement over the finite pool. Ridge-LCB is the uncertainty-aware variant used in fixed-learning-rate Qwen and prospective comparisons; their frozen baseline uses mean predictions from the initial Ridge fit. Full-budget controls use fixed Q-PEFT; retrieval comparisons use constrained enumeration, classical QUBO solvers, idealized Ising retrieval, and reliability-gated real-QPU retrieval.

Evaluation. Offline replay shares initial observations, candidate outcomes, budgets, and parameter caps, revealing only queried outcomes. A noise stress test reveals one unseen seed-level outcome per GPU run and evaluates recommendations against frozen five-seed means, while prospective search trains candidates online. We report simple regret, regret AUC, high-quality and global-best hit rates, and recovery. After the selected configuration is fixed, paired full-budget retraining reports validation loss, BLEU (Papineni et al., 2002), ROUGE-L (Lin, 2004), parameters, training time, and peak memory; held-out generation remains excluded from selection. The audited study comprises 504 training runs (55.71 summed GPU-hours); evaluating all 1,120 legal configurations once is estimated at approximately 182 GPU-hours, while offline replay adds no training cost (Appendix B.8).

4.2 Feedback-Driven Search under Matched Budgets

We compare search policies on a fixed pool of 64 GPT-2/E2E-NLG configurations under three trainable-parameter caps. Each of 1,000 paired trials uses eight shared initial observations and a

Table 1: Paired complete-policy replay on 64 configurations over 1,000 trials with eight initial observations and 12 evaluations. Each cap reports final regret and global-best hit rate. Top-10% results and the fixed-LCB refitting ablation are reported in Appendix C.

	100% cap		75% cap		50% cap	
Policy	Regret ↓	Best (%) ↑	Regret ↓	Best (%) ↑	Regret ↓	Best (%) ↑
Refitted Ridge (mean)	**0.013987**	**81.9**	**0.018268**	**88.4**	**0.001413**	**98.5**
Frozen Ridge-LCB	0.022948	72.4	0.028206	80.4	0.004409	95.1
Structured GP-EI	0.044631	55.2	0.056173	65.6	0.004560	95.0
RF-EI	0.063803	31.5	0.093199	40.7	0.028300	61.2
TPE	0.040350	41.0	0.047109	65.4	0.020895	79.7
SMAC-ID	0.085206	21.8	0.102833	28.5	0.038445	51.8
Random	0.086014	22.7	0.105568	29.5	0.037877	53.4

Table 2: Five-seed full-budget confirmation of QPU-PEFT selections against fixed Q-PEFT controls. Deltas are selected minus control; joint wins count seeds improving both BLEU and ROUGE-L.

Comparison	Control	Δ Val. ↓	Δ BLEU ↑	Δ R-L ↑	Joint wins	Resource note
QPU-PEFT-selected vs. fixed Q-PEFT	Same structure/params.	-0.051	$\mathbf{+3.888}$	$\mathbf{+1.688}$	$5/5$	LR selected; same memory
`q27` vs. matched baseline	Same LR/params.	$\mathbf{-0.064}$	$\mathbf{+2.638}$	$\mathbf{+1.339}$	$5/5$	$+18.3\%$ train time
`q55` vs. anchor `q00`	Frozen promotion	$\mathbf{-0.066}$	$\mathbf{+2.708}$	$\mathbf{+1.278}$	$5/5$	$+55{,}080$ params.; -38.3% time

budget of 12 evaluations. As shown in Table 1, Refitted Ridge (mean) performs best under all three caps. At the reference cap, it attains regret 0.01399 and global-best hit 81.9%, compared with 0.02295 and 72.4% for Frozen Ridge-LCB; under the tightest cap, its best hit reaches 98.5%. This is a complete-policy comparison because both refitting and acquisition differ; Appendix C isolates refitting with LCB held fixed.

Fixed-learning-rate Qwen pools exclude gains from learning-rate tuning. On Qwen2.5-0.5B, Refitted Ridge-LCB reaches the global best in 98.8% of trials, compared with 89.4% for Frozen Ridge (mean). On Qwen2.5-1.5B, it obtains regret 0.002559 and Top-2 recovery 88.2%, compared with 0.003423 and 76.3% for RF-EI. Under single-seed noisy feedback, it retains significantly lower five-seed-mean regret than GP-EI, RF-EI, TPE, and random search.

In the prospective GPT-2/E2E-NLG experiment, Refitted Ridge-LCB reduces budget-12 audited-union regret from 0.0100 to 0.0069 relative to Frozen Ridge (mean), while the policies converge at larger budgets.

4.3 Full-Budget Adaptation Quality

We first assess whether low-fidelity screening predicts full-budget performance on GPT-2/E2E-NLG. We retrain 20 prespecified configurations with three seeds each. The validation-loss rankings under the two training budgets exhibit Spearman and Kendall correlations of 0.9308 and 0.7789, respectively. Three of the four highest-ranked configurations under full-budget training also appear in the low-fidelity Top-4. The selected configuration `q55` retains the best full-budget validation loss among the audited configurations, yielding zero promotion regret within this set.

Table 2 compares QPU-PEFT selections with fixed Q-PEFT controls. QPU-PEFT selects Direct Q-PEFT configuration `c11` over prespecified `c00`; differing only in learning rate, `c11` reduces validation loss by 0.051 and improves BLEU and ROUGE-L by 3.888 and 1.688 on all five paired seeds without increasing parameters or peak memory. Against a learning-rate- and parameter-matched Q-PEFT control, `q27` improves BLEU by 2.638 and ROUGE-L by 1.339 on all five seeds.

Compared with its prespecified baseline, `q55` improves BLEU by 2.708 and ROUGE-L by 1.278. It uses more trainable parameters but reduces training time by 38.3%, illustrating that parameter count and training cost need not increase together. The larger configuration `q56` performs worse than `q55`.

On Qwen2.5-1.5B, selected configuration `m03` lowers validation loss on all ten paired seeds relative to `m00`, while using 28.6% fewer trainable parameters and reducing training time by 41.0%. ROUGE-L

Table 3: Finite-pool resource-constrained QUBO audit for refitted mean acquisition over 12,000 decisions. Agreement compares exact QUBO retrieval with direct constrained enumeration of the same frozen score.

Nominal cap	Realized cap	Feasible pool	Unconstrained feasible	QUBO feasible	Exact agreement	Max. error
100%	55,368	51	23.53%	**100%**	**100%**	**0**
75%	36,960	40	15.03%	**100%**	**100%**	**0**
50%	24,544	22	3.70%	**100%**	**100%**	**0**

Table 4: Matched-shot backend comparison over 1,000 paired initial designs. All methods share the same frozen surrogate and use matched sampling budgets, shortlist sizes, gating rules, and evaluation budgets.

Retrieval backend	Regret@12 ↓	Top-10%@12 ↑	Recovery@12 ↑
Shot-based ideal Ising/QAOA	**0.04172**	**96.3%**	**89.9%**
Randomized Top-k	0.05296	84.5%	57.7%
Boltzmann/Gibbs	0.05524	83.3%	54.4%
Random search	0.05773	78.9%	42.3%

improves by 0.544, whereas BLEU decreases by 0.323, showing that improvements in validation loss and resource efficiency do not necessarily translate into gains across all generation metrics.

4.4 Constrained QUBO Optimization and Ising Sampling

We evaluate whether QUBO compilation preserves the constrained mean-acquisition decision across 1,000 initial designs, four adaptive steps, and three parameter caps, yielding 12,000 decisions. Unconstrained mean ranking produces budget-feasible proposals in only 3.70–23.53% of cases. Exact QUBO retrieval returns a feasible proposal and matches direct constrained enumeration in every case, with zero acquisition-objective discrepancy. These results establish decision consistency for the tested instances.

At the 75% parameter cap, five paired full-budget retrainings show that selected configuration `q07` reduces mean validation loss by 0.00859, trainable parameters by 33.4%, and mean training time by 52.4% relative to `q00` (Table 24). Peak GPU memory increases by 2.7%, while the 95% confidence intervals for BLEU and ROUGE-L differences include zero.

Matched-shot sampling control. We compare retrieval backends under identical 1,024-sample, Top-6 shortlist, energy-gating, and query-budget conditions. Table 4 shows that shot-based ideal Ising/QAOA provides the strongest budget-12 high-quality-region coverage and recovery.

Relative to Boltzmann sampling, Ising/QAOA reduces regret by 0.01352 (95% CI $[-0.01596, -0.01115]$) and improves Top-10% retrieval and recovery by 13.0 and 35.5 percentage points. Its advantage does not extend to exact global-best recovery (20.9% vs. 29.9% for random search). Complete sampling controls are reported in Appendix F.

4.5 Reliability-Gated Execution on Real QPUs

We first conduct a four-round sequential replay on TianYan-176 using the fixed-learning-rate Qwen2.5-0.5B pool. Starting from eight fixed observations, each round refits the Ridge surrogate, recomputes its LCB acquisition score, constructs a new eight-candidate, seven-qubit Ising instance, and executes three independent 2,048-shot jobs. Candidate outcomes are prespecified five-seed mean validation losses and remain hidden until the corresponding candidate is requested. All four rounds pass a frozen target-free gate requiring legal repeat support and mean pairwise TV at most 0.25.

The sequential replay reaches the empirical pool optimum in round 2 and queries four of the five best configurations by round 4. The QPU and deterministic classical routes agree in every round, establishing repeated hardware participation and preservation of the constrained decision rather than superiority over classical retrieval.

Table 5: Four-round sequential replay on TianYan-176. Each round uses three independent 2,048-shot executions. Pool ranks are computed post hoc and are not visible during retrieval.

Round	QPU route	Classical route	Pool rank	Revealed loss	Mean TV
1	s05	s05	5	1.421967	0.1102
2	s19	s19	**1**	**1.372486**	0.1162
3	s03	s03	2	1.392309	0.1172
4	s27	s27	3	1.407050	0.1190

A separate frozen end-to-end audit tests whether QPU routing can change the downstream decision. On TianYan-176, the gate accepts hardware candidate p01, whereas deterministic classical retrieval proposes p05. Across three new paired full-budget Qwen2.5-1.5B/E2E-NLG seeds, the differences p01−p05 are -0.101 BLEU (95% CI $[-0.372, 0.118]$) and $+0.164$ ROUGE-L (95% CI $[-0.090, 0.523]$). Both interval lower bounds exceed the prespecified -0.5 non-inferiority margin, and the candidates have the same trainable-parameter count. Thus, the accepted QPU proposal changes the routed configuration while preserving downstream quality; it does not establish QPU superiority.

The rejection branch is exercised separately on TianYan-287: the gate rejects hardware proposal p02 and routes deterministic classical candidate p05. All six unique downstream training runs complete, populating nine arm–seed entries. Complete protocols, acceptance and fallback audits, and paired downstream results are reported in Appendix F. A retrospective counterfactual audit evaluates the rejected proposal p02 against the selected candidate p05 on the same three paired seeds. Their BLEU and ROUGE-L differences are inconclusive, while p02 has slightly lower validation loss but requires slightly more training time. Thus, the gate preserves a deterministic route but can conservatively reject a downstream-competitive proposal, exposing a safety–coverage trade-off rather than demonstrating that every rejected proposal is inferior.

4.6 Ablations and Scope

Holding the LCB acquisition rule fixed, feedback refitting reduces final regret by 0.0080, 0.0070, and 0.0028 under the 100%, 75%, and 50% parameter caps, respectively. Refitted LCB does not consistently outperform Refitted Ridge (mean) and yields higher regret under the 75% cap. Together with the Refitted Ridge (mean) comparison, these results support the benefit of feedback updating while showing no consistent additional benefit from the uncertainty bonus.

We isolate the gate components by replaying the same TianYan-287 observations under different acceptance rules. Both legality-only and legality-plus-stability checks accept hardware proposal p02, whereas adding the target-support requirement rejects it and selects classical fallback p05. Target-region support is therefore the decisive acceptance criterion in this hardware instance, despite stable sampling distributions.

Offline replay resamples initial designs over fixed candidate outcomes, so its conclusions are conditional on the evaluated pools. Prospective execution and paired full-budget retraining provide complementary evidence. The QUBO audits establish decision consistency at the tested scales, where exact and classical retrieval remain practical. The hardware experiments demonstrate reliability-gated execution but do not establish runtime acceleration or quantum advantage.

5 Conclusion

We presented QPU-PEFT, a feedback-driven framework for budgeted Q-PEFT configuration selection that combines constrained QUBO/Ising retrieval with reliability-gated QPU execution. Matched-budget experiments demonstrate the benefit of feedback refitting for search, while paired full-budget retraining shows that selected configurations can improve adaptation quality or reduce training costs, with trade-offs across generation metrics. QUBO validation confirms feasibility and agreement with direct constrained enumeration on the tested instances. Experiments on TianYan-176 and TianYan-287 demonstrate both acceptance of supported QPU proposals and deterministic classical fallback. Future work will examine larger configuration spaces and multi-objective acquisition to jointly account for task quality and resource costs.

AI USE STATEMENT

OpenAI Codex was used to assist manuscript drafting and editing, bilingual translation, LaTeX restructuring, and interpretation and consistency checking of author-provided experimental outputs. It was not used to conduct the reported training runs or generate the experimental observations. All AI-assisted text, numerical claims, citations, and artifacts were reviewed against the archived configurations, raw per-seed CSV files, and cited literature. The human authors take full responsibility for the research decisions and final submission.

ETHICS STATEMENT

This study uses publicly released pretrained language models and established data-to-text generation and text-classification benchmarks. It does not collect new human-subject data or process private or personally identifying information. Generated text may reproduce factual errors, stereotypes, or other harms inherited from pretrained models and benchmark data. Reported metrics are not safety guarantees, and any deployment requires dataset-license review, application-specific evaluation, and human oversight. We make no environmental or deployment-efficiency claims beyond the parameter counts, training times, and memory measurements obtained under the reported experimental protocols.

REPRODUCIBILITY STATEMENT

Section 4.1 defines the evaluation protocol. Appendix A provides the closed-loop procedure, constrained QUBO/Ising formulation, reliability-gate definitions, theoretical details, and optional warm-start derivation. Appendix B records candidate spaces, training, decoding, resource constraints, and hardware settings. Appendix C reports replay, baseline, noise-stress, prospective, and statistical analyses; Appendix D reports fidelity audits, paired full-budget confirmations, resource-constrained results, and optional warm-start experiments; Appendix E reports cross-task transfer and target-task reselection; real-QPU protocols, shot-level diagnostics, gate decisions, and deterministic routing outcomes are reported in Appendix F. The anonymous supplementary artifact contains frozen configurations, seed- and trial-level records, canonical CSV summaries, reconstruction scripts, and integrity manifests; Appendix G documents its organization and use.

REFERENCES

Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72, 2005.

Ricardo Baptista and Matthias Poloczek. Bayesian optimization of combinatorial structures. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 462–471, 2018.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10):281–305, 2012.

James S. Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Nicholas Chancellor. Domain-wall encoding of discrete variables for quantum annealing and QAOA. *Quantum Science and Technology*, 4(4):045004, 2019.

Aofei Chang, Jiaqi Wang, Han Liu, Parminder Bhatia, Cao Xiao, Ting Wang, and Fenglong Ma. BIPEFT: Budget-guided iterative search for parameter-efficient fine-tuning of large pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7429–7440, 2024.

Zhuo Chen, Rumen Dangovski, Charlotte Loh, Owen Dugan, Di Luo, and Marin Soljačić. QuanTA: Efficient high-rank fine-tuning of LLMs with quantum-informed tensor adaptation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 92210–92245, 2024.

George Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, pp. 138–145, 2002.

Stefan Falkner, Aaron Klein, and Frank Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 1437–1446, 2018.

Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pp. 124–133, 2017.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization*, volume 6683, pp. 507–523, 2011.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

Koki Kitai, Jiang Guo, Shenghong Ju, Shu Tanaka, Koji Tsuda, Junichiro Shiomi, and Ryo Tamura. Designing metamaterials with quantum annealing and factorization machines. *Physical Review Research*, 2(1):013319, 2020.

Toshiaki Koike-Akino, Francesco Tonin, Yongtao Wu, Frank Zhengqing Wu, Leyla Naz Candogan, and Volkan Cevher. Quantum-PEFT: Ultra parameter-efficient fine-tuning. In *International Conference on Learning Representations*, 2025.

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(185):1–52, 2018.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1823–1840, 2020.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, 2004.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 32100–32121, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2:5, 2014.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 121038–121072, 2024.

Linyong Nan, Dragomir Radev, Rui Zhang, Amrit Rau, Abhinand Sivaprasad, Chiachun Hsieh, Xiangru Tang, Aadit Vyas, Neha Verma, Pranav Krishna, Yangxiaokang Liu, Nadia Irwanto, Jessica Pan, Faiaz Rahman, Ahmad Zaidi, Mutethia Mutuma, Yasin Tarabar, Ankit Gupta, Tao Yu, Yi Chern Tan, Xi Victoria Lin, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. DART: Open-domain structured data record to text generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 432–447, 2021.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pp. 201–206, 2017.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, 2018.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

Snehal Raj and Brian Coyle. QuIC: Quantum-inspired compound adapters for parameter-efficient fine-tuning. *arXiv preprint arXiv:2502.06916*, 2025.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, 2013.

Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. DyLoRA: Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 3274–3287, 2023.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4566–4575, 2015.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*, 2023.

Han Zhou, Xingchen Wan, Ivan Vulić, and Anna Korhonen. AutoPEFT: Automatic configuration search for parameter-efficient fine-tuning. *Transactions of the Association for Computational Linguistics*, 12:525–542, 2024.

A Algorithmic and Mathematical Details

This appendix provides the complete QPU-PEFT procedure, the constrained-QUBO guarantee, the reliability-gate definition, the gate-conditioned selection bound, and the optional warm-start derivation corresponding to Section 3.

A.1 Closed-Loop Selection Procedure

Let $\mathcal{X}$ be the frozen set of legal configurations, B the GPU-evaluation budget, and $\mathcal{D}_0 = \{(x_i, f(x_i))\}_{i=1}^{n_0}$ the initial observations. At iteration t, the eligible set is

$$\mathcal{U}_t = \mathcal{X} \setminus \{x_i : (x_i, f(x_i)) \in \mathcal{D}_t\}. \tag{15}$$

QPU-PEFT repeats the following steps until B evaluations are consumed:

1. Fit the response model using $\mathcal{D}_t$ and construct the prespecified acquisition score.
2. Compile the acquisition score and frozen constraints as $E_t(z) = q_t(z) + \lambda C(z)$.
3. Freeze the eligible set, decoder, backend settings, shortlist size, support rule, gate thresholds, tie-breaking rules, and deterministic classical candidate.
4. Retrieve a candidate through direct classical retrieval or reliability-gated QPU routing.
5. Evaluate exactly one routed candidate and update
$$\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(x_{t+1}, f(x_{t+1}))\}. \tag{16}$$

After budget exhaustion, the configuration with the lowest observed validation loss is frozen before full-budget retraining and held-out evaluation. Paired policies share initial observations and budgets but maintain isolated histories; a cached outcome is revealed only after the corresponding policy requests that configuration.

A.2 Feedback Refitting and Acquisition

Each configuration has a binary encoding $z(x) \in \{0,1\}^n$. With a prespecified feature map ϕ, Ridge fitting gives

$$\widehat{w}_t = \arg\min_w \left[\sum_{(x_i,y_i)\in\mathcal{D}_t} \left(y_i - w^\top \phi(z(x_i))\right)^2 + \lambda_{\mathrm{R}} \|w\|_2^2 \right], \qquad \widehat{f}_t(x) = \widehat{w}_t^\top \phi(z(x)). \tag{17}$$

The predictive-mean and uncertainty-adjusted acquisition scores are

$$a_t^{\mathrm{mean}}(x) = \widehat{f}_t(x), \qquad a_t^{\mathrm{LCB}}(x) = \widehat{f}_t(x) - \kappa_t u_t(x), \tag{18}$$

where $u_t(x)$ is estimated from bootstrap refits. A refitted policy updates the model after each revealed outcome, whereas a frozen policy retains the model fitted on $\mathcal{D}_0$. The backend-ready quadratic score is written as

$$q_t(z) = \gamma_{0,t} + \sum_i \gamma_{i,t} z_i + \sum_{i<j} \gamma_{ij,t} z_i z_j. \tag{19}$$

For the constrained mean-acquisition audit, $q_t(z(x)) = \widehat{f}_t(x)$. Other frozen acquisition scores are compiled using the corresponding finite-pool encoding.

A.3 Constrained QUBO Formulation

The nonnegative constraint penalty is

$$C(z) = C_{\mathrm{cat}}(z) + C_{\mathrm{struct}}(z) + C_{\mathrm{resource}}(z), \tag{20}$$

where the three terms enforce categorical consistency, structural compatibility, and the trainable-parameter budget. Thus, $C(z) = 0$ for legal encodings and $C(z) > 0$ otherwise. Previously evaluated configurations are removed from the eligible set before retrieval.

The compiled objective is

$$E_t(z) = q_t(z) + \lambda C(z), \qquad z \in \{0,1\}^n. \tag{21}$$

Proposition 1 (feasibility and decision preservation). Let $\mathcal{Z}_{\text{legal}} = \{z : C(z) = 0\}$ be nonempty. When infeasible encodings exist, define

$$v_{\min} = \min_{z:C(z)>0} C(z), \qquad \Delta_t = \max_z q_t(z) - \min_z q_t(z). \tag{22}$$

If $\lambda v_{\min} > \Delta_t$, then

$$\arg\min_{z\in\{0,1\}^n} E_t(z) = \arg\min_{z\in\mathcal{Z}_{\text{legal}}} q_t(z). \tag{23}$$

Proof. Let $q_t^{\min}$ and $q_t^{\max}$ be the extrema of q_t over the binary domain. Every infeasible state satisfies

$$E_t(z_{\text{inf}}) \geq q_t^{\min} + \lambda v_{\min} > q_t^{\min} + \Delta_t = q_t^{\max},$$

whereas every feasible state satisfies

$$E_t(z_{\text{feas}}) = q_t(z_{\text{feas}}) \leq q_t^{\max}.$$

Hence no infeasible state is a global minimizer. Since $E_t = q_t$ on $\mathcal{Z}_{\text{legal}}$, the feasible minimizers coincide. □

Using $z_i = (1 - s_i)/2$ yields the equivalent Ising objective

$$H_t(s) = c_t + \sum_i h_{i,t} s_i + \sum_{i<j} J_{ij,t} s_i s_j, \qquad s_i \in \{-1, +1\}. \tag{24}$$

Exact enumeration, classical solvers, idealized Ising sampling, and real QPUs use the same frozen coefficients and decoder. The compilation audit is reported in Table 3.

A.4 Reliability-Gated QPU Routing

Suppose the QPU is executed in R repetitions. Repetition r produces an empirical distribution $\widehat{p}_{r,t}$ and a shortlist $\mathcal{S}_{h,r,t}$ containing its h most frequent legal decoded configurations. For any nonempty support set $\mathcal{A} \subseteq \mathcal{U}_t$, define

$$R_t(\mathcal{A}) = \sum_{r=1}^{R} \mathbb{I}\left[\mathcal{S}_{h,r,t} \cap \mathcal{A} \neq \varnothing\right]. \tag{25}$$

The retained set and QPU proposal are

$$\mathcal{S}_t^{\text{ret}} = \bigcup_{r=1}^{R} \mathcal{S}_{h,r,t}, \qquad x_{t+1}^{\text{QPU}} \in \arg\min_{x\in\mathcal{S}_t^{\text{ret}}} q_t(z(x)), \tag{26}$$

with deterministic tie breaking. Repeat stability is measured by

$$\overline{\text{TV}}_t = \frac{2}{R(R-1)} \sum_{1\leq a<b\leq R} \frac{1}{2} \sum_s \left|\widehat{p}_{a,t}(s) - \widehat{p}_{b,t}(s)\right|. \tag{27}$$

The support set is

$$\mathcal{A}_t = \begin{cases} \mathcal{T}_t, & \text{preregistered-target audit,} \\ \{x_{t+1}^{\text{QPU}}\}, & \text{target-free sequential replay,} \end{cases} \tag{28}$$

where $\mathcal{T}_t \subseteq \mathcal{U}_t$ is frozen before the corresponding target-audit execution. The target-free rule instead checks whether the routed QPU candidate itself is repeat-supported.

Let $L_t = 1$ indicate that the proposal is available and decodes completely and legally. The gate is

$$G_t = \mathbb{I}\left[L_t = 1 \wedge R_t(\mathcal{A}_t) \geq r_{\min} \wedge \overline{\text{TV}}_t \leq \tau_{\text{TV}}\right]. \tag{29}$$

The routed configuration is

$$x_{t+1} = \begin{cases} x_{t+1}^{\text{QPU}}, & G_t = 1, \\ x_{t+1}^{\text{C}}, & G_t = 0, \end{cases} \tag{30}$$

where $x_{t+1}^{\text{C}} \in \mathcal{U}_t$ is the deterministic classical candidate. The objective, decoder, shortlist size, support rule, thresholds, tie-breaking rules, and classical route are frozen before inspecting the corresponding hardware outcomes.

Frequency-shortlist support differs from a state being sampled at least once and from pooled probability mass on an exact-energy Top-K region. The latter quantities are reported only as hardware diagnostics.

Proposition 2 (gate-conditioned selection bound). Let

$$x_t^\star \in \arg\min_{x \in \mathcal{U}_t} f(x) \tag{31}$$

and assume

$$\sup_{x \in \mathcal{U}_t} |f(x) - q_t(z(x))| \le \epsilon_t. \tag{32}$$

Define

$$q_t^\star = \min_{x \in \mathcal{U}_t} q_t(z(x)), \qquad \Delta_{\mathcal{A},t} = \max_{x \in \mathcal{A}_t} q_t(z(x)) - q_t^\star, \tag{33}$$

and suppose

$$q_t(z(x_{t+1}^{\mathrm{C}})) - q_t^\star \le \delta_t^{\mathrm{C}}. \tag{34}$$

If $r_{\min} \ge 1$, then

$$f(x_{t+1}) - f(x_t^\star) \le 2\epsilon_t + G_t \Delta_{\mathcal{A},t} + (1 - G_t)\delta_t^{\mathrm{C}}. \tag{35}$$

Proof. If $G_t = 1$, then $R_t(\mathcal{A}_t) \ge 1$, so some $x' \in \mathcal{A}_t \cap \mathcal{S}_t^{\mathrm{ret}}$ exists. Since the QPU branch minimizes the frozen score over $\mathcal{S}_t^{\mathrm{ret}}$,

$$q_t(z(x_{t+1}^{\mathrm{QPU}})) - q_t^\star \le \Delta_{\mathcal{A},t}.$$

If $G_t = 0$, the classical condition gives retrieval error at most δ_t^{C}. Hence

$$q_t(z(x_{t+1})) - q_t^\star \le G_t \Delta_{\mathcal{A},t} + (1 - G_t)\delta_t^{\mathrm{C}}.$$

Using $q_t^\star \le q_t(z(x_t^\star))$ and the uniform approximation assumption yields

$$f(x_{t+1}) - f(x_t^\star) \le 2\epsilon_t + G_t \Delta_{\mathcal{A},t} + (1 - G_t)\delta_t^{\mathrm{C}}.$$

□

For an exact-score Top-K target,

$$\Delta_{\mathcal{A},t} = q_t^{(K)} - q_t^{(1)}.$$

For target-free sequential routing, $\mathcal{A}_t$ is the singleton containing the repeat-supported QPU proposal, so $\Delta_{\mathcal{A},t}$ is its frozen-score gap. A singleton ground-state target gives $\Delta_{\mathcal{A},t} = 0$, while exact classical retrieval gives $\delta_t^{\mathrm{C}} = 0$. The hardware protocols and support audits are reported in Appendix F.

A.5 Optional Warm-Start Transfer

Warm-start transfer is optional and does not alter the feedback, compilation, or gate definitions. When enabled, a temporary rank-r_w LoRA adapter is trained for a fraction ρ of the schedule. For $\Delta W_{\mathrm{L}} = \widetilde{B}A$, thin QR factorizations and a small-core SVD give

$$\widetilde{B} = Q_B R_B, \qquad A^\top = Q_A R_A, \qquad R_B R_A^\top = U_C \Sigma V_C^\top. \tag{36}$$

The transferred rank-r update is

$$\Delta W_r = (Q_B U_{C,r}) \Sigma_r (Q_A V_{C,r})^\top, \tag{37}$$

which is the best rank-r Frobenius approximation without materializing the dense update. Its retained energy is

$$\eta_r = \frac{\sum_{i=1}^r \sigma_i^2}{\sum_i \sigma_i^2}. \tag{38}$$

Let $\widehat{U}(\theta_U)$ and $\widehat{V}(\theta_V)$ be the truncated-Taylor Q-PEFT frames and let a denote their singular coordinates. Coordinate fitting minimizes

$$\mathcal{L}_{\mathrm{fit}} = \frac{\left\| \widehat{U}(\theta_U)\, \mathrm{diag}(a)\, \widehat{V}(\theta_V)^\top - U_r \Sigma_r V_r^\top \right\|_F^2}{\|U_r \Sigma_r V_r^\top\|_F^2 + \varepsilon}. \tag{39}$$

After fitting, the temporary LoRA module and optimizer state are discarded, and Q-PEFT continues with a fresh optimizer. Direct Q-PEFT disables this branch.

B Complete Experimental Configuration

This appendix records the candidate spaces, query budgets, baselines, training protocols, metrics, and hardware settings used in Section 4. Search policies observe validation loss only. Held-out generation metrics are revealed after the selected configuration and comparison protocol have been frozen. Detailed results are reported in Appendices C–F.

B.1 Protocol Summary

A candidate evaluation consists of one protocol-matched training run followed by validation-loss measurement. Table 6 summarizes the sequential-search and retrieval experiments.

Table 7 lists the fidelity and full-budget confirmation experiments. These studies use configurations frozen before held-out evaluation.

The 10-, 16-, 20-, 32-, 39-, and 64-configuration studies use frozen candidate sets. They are not presented as complete Cartesian products. Every candidate has a stable identifier and a complete configuration record in the anonymous artifact.

B.2 Candidate Spaces and Resource Constraints

Each configuration specifies structural and training choices through categorical indicators. Table 8 lists the fields used across the candidate spaces.

The matched-budget 64-candidate experiment evaluates three parameter caps. Because only discrete configurations are available, the realized limits are the largest attainable parameter counts below the nominal caps.

The Qwen2.5-0.5B and Qwen2.5-1.5B pools fix learning rate and vary structural fields, isolating structural selection from learning-rate tuning. Candidate labels are five-seed means for the 32-configuration 0.5B pool and three-seed means for the 16-configuration 1.5B pool. One-hot, structural, and resource constraints follow Appendix A.3; previously evaluated configurations are removed before retrieval.

B.3 Policies and Baselines

Refitted Ridge. The primary policy refits the Ridge response model after every revealed outcome and ranks eligible configurations by predicted validation loss.

Refitted Ridge-LCB. The uncertainty-aware variant uses 200 bootstrap refits and the frozen exploration coefficient specified by the corresponding protocol. It is used to distinguish the effect of feedback refitting from that of the uncertainty bonus.

Frozen Ridge. The same Ridge model is fitted using only the initial observations and remains unchanged during adaptive evaluation.

Structured GP-EI and RF-EI. Both baselines use min–max normalized numerical fields, one-hot encoded categorical fields, exact finite-pool expected improvement, and the same initial observations, feasible pools, and query budgets as Refitted Ridge. GP-EI uses a Matérn-$5/2$ kernel and normalized observed losses. RF-EI uses a 128-tree bootstrap random forest and estimates predictive uncertainty from variation across trees. Wider GP length-scale bounds and additional optimizer restarts are evaluated in Appendix C.

RF-LCB. RF-LCB uses a random-forest surrogate with a lower-confidence-bound acquisition rule and otherwise follows the same paired replay protocol.

TPE, SMAC-ID, and random search. Categorical TPE operates on the same candidate fields and isolated search histories. SMAC-ID treats candidate identifiers as categorical choices under the same feasible pool and budget. Random search samples uniformly without replacement.

Table 6: Search and retrieval protocols. All paired policies share the same initial observations and evaluation budget while maintaining isolated histories.

Study	Model / task	Pool	Initial	Budget	Replication / evidence
Matched-budget search	GPT-2 / E2E	64	8	12	1,000 trials; three parameter caps
Fixed-LR structural search	Qwen2.5-0.5B / E2E	32	8	12/16	1,000 trials; five-seed labels
Fixed-LR structural search	Qwen2.5-1.5B / E2E	16	6	10	1,000 trials; three-seed labels
Prospective search	GPT-2 / E2E	64	8	24	Seeds 400–404; prefixes 12/16/24
Prospective search	Model-matched / SST-2	16	4	8	Seeds 370–372
Compilation audit	GPT-2 / E2E	64	8	12	12,000 decisions over three caps
Matched-shot retrieval	GPT-2 / E2E	39	8	12	1,000 trials; 1,024 samples per step
Single-seed noisy replay	Qwen2.5-0.5B / E2E	32	8	12/16/24	1,000 trials; one seed-level outcome per query

Table 7: Fidelity and full-budget confirmation protocols. The seed is the unit of comparison.

Study	Comparison / scope	Replication	Evidence role
Primary confirmation	`c11` vs. `c00`	Seeds 321–325	Same structure and parameters
Matched structural control	`q27` vs. matched baseline	5 paired seeds	Same LR and parameters
Fidelity audit	20 prespecified configurations	3 seeds each; 60 runs	Low-to-full ranking agreement
Promotion confirmation	`q55` vs. `q00`/`q56`	Seeds 461–465	Frozen promotion decision
Resource-constrained confirmation	`q07` vs. `q00`	5 paired seeds	75% parameter cap
Modern-backbone confirmation	`m03` vs. `m00`	Seeds 704–713	Quality–efficiency trade-off
WebNLG reselection	Selected vs. fixed candidate	Seeds 42–46	Task-specific adaptation
Optional warm start	Direct vs. transfer variants	Seeds 421–428	Mechanism audit

Table 8: Q-PEFT configuration fields. Individual pools may freeze a subset of these fields.

Field	Symbol	Constraint or interpretation
Adapter rank	r	Retained update rank
Intrinsic rank	k	$k \leq r$
Taylor order	p	Truncated-expansion order
Adapter scale	α	Candidate-specific scale
Learning rate	η	Fixed or selected by protocol
Dropout	d	Candidate-specific dropout
Target modules / layers	m	Valid projection subset
Warm-start fraction	ρ	Optional; zero for Direct Q-PEFT
Trainable parameters	$P(x)$	$P(x) \leq P_{\max}$

Table 9: Resource-constrained candidate sets used in matched-budget search and QUBO compilation.

Nominal cap	Realized cap	Feasible candidates	Adaptive decisions
100%	55,368	51	$1{,}000 \times 4$
75%	36,960	40	$1{,}000 \times 4$
50%	24,544	22	$1{,}000 \times 4$

QUBO and Ising retrieval. Exact constrained QUBO minimizes the compiled score over legal configurations. Classical samplers, ideal Ising/QAOA sampling, and deterministic energy ranking receive the same frozen score, shortlist rule, and sample budget in matched-shot comparisons. Exact enumeration supplies reference feasibility, energies, and Top-K sets but is not counted as an adaptive search policy.

Real-QPU retrieval. Hardware samples are decoded and rescored using the frozen objective. A proposal enters model evaluation only when the frozen gate passes; otherwise, the preregistered classical candidate is routed. Hardware samples never modify the objective before the routing decision.

B.4 Training and Full-Budget Confirmation

The primary GPT-2/E2E full-budget protocol uses 8,000 training examples, 1,000 validation examples, two epochs, micro-batch size 4, gradient accumulation 8, AdamW (Loshchilov & Hutter, 2019) with weight decay 0.01, gradient clipping at 1.0, and FP16 training. Paired arms share the seed, data order, evaluation examples, optimizer family, decoding procedure, and every setting not selected by the configuration policy.

Configurations `c00` and `c11` have the same adapter structure, target modules, 73,632 trainable parameters, and training schedule; their only selected difference is learning rate, which changes from 0.002 to 0.004. The matched `q27` control shares its learning rate, target modules, and 110,448 trainable parameters. The `q55`, `q07`, and `m03` comparisons use independently frozen baselines and new paired seeds.

The fidelity audit retrains 20 prespecified configurations under the full protocol with seeds 451–453. The promoted candidate is frozen before the five-seed held-out confirmation. Low-fidelity results are used for screening only and are not substituted for full-budget metrics.

B.5 Decoding, Metrics, and Statistical Units

Autoregressive generation uses the model-specific tokenizer, prompt-masked causal-language-model loss, deterministic greedy decoding, at most 48 new tokens, and end-of-sequence termination. Paired arms use identical prompts, references, decoding settings, and evaluation examples.

Candidate selection uses validation loss only. Held-out generation quality is measured by BLEU (Papineni et al., 2002), NIST (Doddington, 2002), METEOR (Banerjee & Lavie, 2005), ROUGE-L (Lin, 2004), and CIDEr (Vedantam et al., 2015). Resource measurements include trainable parameters, end-to-end candidate training time, and peak allocated GPU memory. Discarded warm-start modules are excluded from the retained parameter count.

For a feasible pool $\mathcal{X}_P$ and observations available by budget b, simple regret is

$$r_b = \min_{x \in \mathcal{D}_b} f(x) - \min_{x \in \mathcal{X}_P} f(x). \tag{40}$$

Regret AUC averages the best-so-far regret across the reported budget range. High-quality hit rate measures entry into the prespecified Top-K or top-decile set, global-best hit rate measures exact recovery, and recovery is conditioned on the initial design missing the target region.

In prospective E2E experiments, regret is computed against the best candidate in the post-hoc union evaluated by any policy on the same seed. Training seeds are the unit of comparison for full-budget experiments, while the shared initial design is the unit of comparison for replay. Paired differences are computed before aggregation, and no held-out generation outcome is used during selection.

B.6 Real-QPU Protocols

Table 10 records the frozen hardware protocols and observed routing outcomes. Hardware proposals are decoded and evaluated under the same objective used by the corresponding exact and classical controls. The sequential protocol uses four feedback-updated Ising objectives and three independent hardware executions per round.

Table 10: Frozen real-QPU protocols and observed routing outcomes. Frequency support means that the specified candidate enters the most frequent legal decoded configurations in the required number of repetitions.

Protocol	Variables	Repeats	Shots/repeat	Support criterion	Outcome
TianYan-176, end-to-end	7	3	2,048	Repeat support; TV	Accept `p01` ($3/3$, TV 0.1165); classical `p05`
TianYan-176, sequential	7 per round	4×3	2,048	Repeat support; TV	$4/4$ rounds pass
TianYan-176, enhanced	7	5	2,048	`c01` in frequency Top-4	Accept `c01`; support $3/5$
TianYan-176, scale audit	4/6/8	3 per scale	1,024	Ground state; TV	Reliability diagnostic
TianYan-287, routed	16	1	1,024	Ground state	Routing stress test
TianYan-287, prospective	7	3	2,048	`p05` in frequency Top-4	Reject `p02`; route `p05`

B.7 Software and Archived Records

The reference implementation uses Python 3.10 or later, PyTorch with CUDA, Transformers 4.41 or later and below version 5, PEFT 0.18.1, Datasets 2.19 or later, Accelerate 0.30 or later, NumPy 1.26 or later, SciPy 1.11 or later, SacreBLEU (Post, 2018) 2.4 or later, Rouge-score 0.1.2 or later, NLTK 3.9 or later, and pycocoevalcap 1.2 or later. Timing is compared only within protocol-matched experiments because runs span multiple GPU environments.

Each run stores its frozen configuration, seed, data identifiers, validation loss, generation metrics when enabled, parameter count, training time, and peak memory. Search records additionally contain initial observations, query history, visible outcomes, surrogate state, backend proposal, gate result, and routed candidate. Artifact organization and deterministic reconstruction are described in Appendix G.

B.8 End-to-End Cost Accounting

We distinguish total experimental effort, construction of the replay archive, the frozen replay pool, exhaustive-space extrapolation, and real-QPU records. Across all audited experiments, 504 completed training runs consume 55.71 summed GPU-hours. These runs span different models, tasks, and protocols and are therefore reported only as total experimental effort.

The final 39-candidate replay pool is derived from a broader archive containing 47 uniquely evaluated configurations and 79 training runs. Constructing this archive requires 12.81 GPU-hours, corresponding to a mean of 0.1622 GPU-hours per training run. Separately, the full Cartesian encoding contains 1,296 states, of which 1,120 satisfy the structural constraints. Applying the observed per-run mean, training each legal configuration once is estimated to require approximately 182 GPU-hours.

Table 11: Computational accounting. GPU-hours are summed training wall-clock hours. Exhaustive evaluation is an extrapolation, not a realized saving or a measure of QPU speedup.

Scope	Quantity	Value
All audited experiments	Completed training runs	504
	Summed training time	55.71 GPU-hours
Outcome archive	Unique evaluated configurations	47
	Training runs	79
	Summed training time	12.81 GPU-hours
	Mean time per training run	0.1622 GPU-hours
Offline replay	Final candidate pool	39
	Paired replay trials	1,000
	Additional model training	0 GPU-hours
Configuration space	Encoded Cartesian states	1,296
	Legal configurations	1,120
	Estimated one-run-per-configuration cost	$\approx$ 182 GPU-hours
Real-QPU audit	Audited jobs	33
	Total shots	67,584
	Timestamped batches / covered jobs	6 / 18
	Recorded submission-to-completion interval	263 s

GPU-hours denote summed training wall-clock time rather than energy use, monetary cost, or elapsed project duration. The approximately 182 GPU-hours estimate assumes one training run per legal configuration and does not match the multi-seed archive-construction protocol. It therefore quantifies the scale of exhaustive GPU training and evaluation rather than a realized saving. The 263-second hardware record aggregates six timestamped batch submissions covering 18 jobs; latency is unavailable for the remaining 15 jobs, and the logs do not separate queueing, circuit execution, decoding, and gate time.

C Feedback Search and Statistical Details

This appendix provides the statistical details supporting Sections 4.2 and 4.6. Shared initial designs are the paired units in replay, while training seeds are the paired units in prospective execution. "Mean" denotes ranking by predicted validation loss, whereas "LCB" denotes the uncertainty-adjusted score $\widehat{f}_t(x) - \kappa u_t(x)$. Full-budget adaptation, fidelity, and optional warm-start results are reported in Appendix D.

C.1 Feedback-Refitting Ablation

Table 12: Top-10% hit rates for the matched-budget complete-policy replay in Table 1. Each entry summarizes 1,000 paired trials with eight shared initial observations and a total budget of 12 evaluations.

Policy	100% cap	75% cap	50% cap
Refitted Ridge (mean)	$\mathbf{98.9}\%$	$\mathbf{99.2}\%$	$\mathbf{99.5}\%$
Frozen Ridge-LCB	97.4%	98.4%	98.4%
Structured GP-EI	92.9%	88.4%	98.6%
RF-EI	89.7%	85.2%	96.1%
TPE	94.6%	93.8%	90.2%
SMAC-ID	82.8%	79.8%	92.3%
Random search	84.1%	77.8%	91.0%

Table 1 compares complete policies: the primary policy uses Refitted Ridge (mean), whereas the frozen baseline uses Frozen Ridge-LCB. Because both feedback refitting and acquisition differ, this comparison is not interpreted as a refitting-only effect. To isolate feedback refitting, we compare Refitted Ridge-LCB with Frozen Ridge-LCB under the same 1,000 initial designs, feasible pools, eight initial observations, and 12-evaluation budget. We also compare Refitted Ridge-LCB with Refitted Ridge (mean) to isolate the additional uncertainty bonus. Negative differences in Table 13 favor the first policy.

Holding the LCB acquisition rule fixed, feedback refitting significantly reduces final regret under all three parameter caps. In this low-noise setting, Refitted Ridge-LCB does not significantly outperform Refitted Ridge (mean) and is significantly worse under the 75% cap. Thus, feedback refitting is the consistent source of improvement, whereas the uncertainty bonus becomes useful mainly under the noisier single-seed protocol reported below.

C.2 Strong Model-Based Baselines

Structured GP-EI and RF-EI use the same initial designs, candidate sets, and query budgets as the corresponding Ridge reference. The Qwen2.5-0.5B comparisons use Refitted Ridge-LCB, matching the fixed-learning-rate protocol, whereas the GPT-2 resource-cap comparisons use Refitted Ridge (mean). Table 14 reports baseline regret minus Ridge regret, so positive values favor the corresponding Ridge policy.

All paired intervals exclude zero. The feedback-driven Ridge policies therefore remain stronger than structured GP-EI and RF-EI under the corresponding matched protocols.

GP sensitivity. We repeat the Qwen2.5-0.5B budget-12 comparison with GP length-scale bounds widened from $[0.1, 10]$ to $[0.01, 100]$ and with ten optimizer restarts. The strengthened GP-EI baseline obtains regret 0.012841, a Top-10% hit rate of 88.2%, and a global-best hit rate of 62.5%, compared with 0.000386, 99.6%, and 98.8% for Refitted Ridge-LCB. The paired regret advantage is 0.012455 with a 95% interval of $[0.011293, 0.013636]$. Across 4,000 GP fits, the median length scale is 0.656; 17.4% of fits reach the lower bound and none reach the upper bound. Wider bounds and additional restarts therefore do not change the conclusion.

Table 13: Paired final-regret differences over 1,000 shared initial designs. Intervals are paired bootstrap 95% intervals from 20,000 resamples.

Cap	Comparison	Regret difference ↓	95% interval
100%	Refitted mean − Frozen LCB	$\mathbf{-0.008962}$	$[-0.011448, -0.006481]$
	Refitted LCB − Frozen LCB	$\mathbf{-0.008030}$	$[-0.010510, -0.005566]$
	Refitted LCB − Refitted mean	$+0.000932$	$[-0.000193, +0.002086]$
75%	Refitted mean − Frozen LCB	$\mathbf{-0.009938}$	$[-0.013077, -0.006842]$
	Refitted LCB − Frozen LCB	$\mathbf{-0.007046}$	$[-0.010069, -0.004131]$
	Refitted LCB − Refitted mean	$+0.002892$	$[+0.001583, +0.004351]$
50%	Refitted mean − Frozen LCB	$\mathbf{-0.002996}$	$[-0.004109, -0.001958]$
	Refitted LCB − Frozen LCB	$\mathbf{-0.002809}$	$[-0.003980, -0.001716]$
	Refitted LCB − Refitted mean	$+0.000188$	$[-0.000116, +0.000587]$

Table 14: Paired comparisons with structured GP-EI and RF-EI over 1,000 shared initial designs. Qwen rows use Refitted Ridge-LCB; GPT-2 rows use Refitted Ridge (mean).

Pool / budget	Baseline	Regret gap ↑	95% interval
Qwen2.5-0.5B, $B = 12$	GP-EI	0.012279	[0.011084, 0.013506]
	RF-EI	0.009229	[0.008149, 0.010314]
Qwen2.5-0.5B, $B = 16$	GP-EI	0.003414	[0.002777, 0.004071]
	RF-EI	0.001112	[0.000736, 0.001522]
GPT-2, 100% cap	GP-EI	0.030644	[0.026670, 0.034695]
	RF-EI	0.049816	[0.045567, 0.054067]
GPT-2, 75% cap	GP-EI	0.037905	[0.032588, 0.043310]
	RF-EI	0.074931	[0.068899, 0.080876]
GPT-2, 50% cap	GP-EI	0.003148	[0.001932, 0.004426]
	RF-EI	0.026887	[0.024469, 0.029335]

C.3 Fixed-Learning-Rate Modern-Model Pools

The Qwen2.5 pools fix the learning rate to isolate structural selection. Qwen2.5-0.5B contains 32 configurations labeled by five-seed mean validation loss, while Qwen2.5-1.5B contains 16 configurations labeled by three-seed means. Every policy receives the same initial design in each of 1,000 paired trials.

At budget 12 on Qwen2.5-0.5B, Refitted Ridge-LCB reduces regret by 89.5% and raises global-best retrieval from 89.4% to 98.8% relative to Frozen Ridge (mean); its budget-16 global-best hit rate is 99.9%. On Qwen2.5-1.5B, Refitted Ridge-LCB reduces regret by 25.2% relative to RF-EI and improves Top-2 recovery from 76.3% to 88.2%. These are complete-policy comparisons because both feedback updating and acquisition differ; the refit-only effect is isolated in Section C.1.

C.4 Single-Seed Noisy Replay

We conduct 1,000 paired trials in which each query reveals one previously unseen seed-level GPU outcome. Repeated evaluations of a configuration are allowed up to its five archived seeds. Policies share the initial designs and per-configuration seed-reveal orders. The final recommendation minimizes the observed sample mean, while regret and hit rates are evaluated against the frozen five-seed configuration means.

Refitted Ridge-LCB remains significantly better than every baseline under seed-level feedback. Its regret decreases from 0.02039 at budget 12 to 0.01059 at budget 16 and 0.00238 at budget 24, while its global-best hit rate increases from 46.9% to 70.2% and 93.5%. Table 17 separates the effects of refitting and the LCB bonus.

At budget 12, feedback refitting reduces regret by 0.01032 with a 95% interval of $[0.00880, 0.01187]$, while the LCB bonus provides a smaller additional reduction of 0.00177 with an interval of

Table 15: Paired replay on fixed-learning-rate Qwen2.5 pools. High-quality hit denotes Top-10% hit for the 0.5B pool and Top-2 hit for the 1.5B pool. Recovery for the 1.5B pool is conditioned on an initial Top-2 miss.

Pool / budget	Policy	Regret ↓	AUC ↓	HQ hit ↑	Best hit ↑	Recovery ↑
Qwen2.5-0.5B, $B = 12$	Refitted Ridge-LCB	**0.000386**	**0.010759**	**99.6%**	**98.8%**	–
	Frozen Ridge (mean)	0.003687	0.012541	97.3%	89.4%	–
	RF-LCB	0.007750	0.018705	93.7%	78.8%	–
	TPE	0.011819	0.021245	92.0%	65.0%	–
	Random search	0.023170	0.028047	86.0%	36.7%	–
Qwen2.5-0.5B, $B = 16$	Refitted Ridge-LCB	**0.000035**	**0.006010**	**100.0%**	**99.9%**	–
	Frozen Ridge (mean)	0.001132	0.007760	99.6%	95.8%	–
	RF-LCB	0.000838	0.011680	99.4%	97.7%	–
	TPE	0.004431	0.015040	98.5%	84.5%	–
	Random search	0.015860	0.023800	94.7%	49.3%	–
Qwen2.5-1.5B, $B = 10$	Refitted Ridge-LCB	**0.002559**	**0.009308**	**95.6%**	89.1%	**88.2%**
	RF-LCB	0.002712	0.009488	92.9%	89.1%	80.9%
	RF-EI (SMAC-style)	0.003423	0.010022	91.2%	**89.2%**	76.3%
	Frozen Ridge (mean)	0.003805	0.009702	89.5%	81.6%	71.8%
	TPE	0.005042	0.010040	91.3%	70.8%	76.6%
	Random search	0.007004	0.011118	88.3%	62.3%	68.5%

Table 16: Single-seed noisy replay at a budget of 12 GPU runs. Paired differences are baseline regret minus Refitted Ridge-LCB regret.

Policy	Regret ↓	Top-10% hit ↑	Best hit ↑	Regret gap [95% CI]
Refitted Ridge-LCB	**0.02039**	**87.5%**	**46.9%**	–
GP-EI	0.03389	69.6%	29.8%	0.01350 [0.01157, 0.01541]
RF-EI	0.03482	69.8%	28.2%	0.01443 [0.01247, 0.01644]
TPE	0.03451	69.9%	27.4%	0.01412 [0.01224, 0.01602]
Random search	0.03626	67.1%	25.2%	0.01587 [0.01391, 0.01783]

Table 17: Ridge ablation under single-seed feedback. Each query consumes one seed-level GPU outcome.

	Regret ↓			Global-best hit ↑		
Policy	$B = 12$	$B = 16$	$B = 24$	$B = 12$	$B = 16$	$B = 24$
Frozen Ridge (mean)	0.03248	0.02632	0.01896	30.8%	43.5%	55.8%
Refitted Ridge (mean)	0.02216	0.01389	0.00440	45.5%	64.8%	89.4%
Refitted Ridge-LCB	**0.02039**	**0.01059**	**0.00238**	**46.9%**	**70.2%**	**93.5%**

$[0.00073, 0.00282]$. The same ordering holds at budgets 16 and 24. Feedback refitting accounts for most of the improvement, while uncertainty-aware exploration provides a smaller additional benefit under seed-level noise.

C.5 Prospective E2E Execution

Prospective execution reveals an outcome only after the corresponding configuration is requested and evaluated. Refitted Ridge-LCB and Frozen Ridge (mean) receive the same initial observations and may reuse a cached GPU outcome only after requesting the corresponding configuration; they otherwise maintain isolated histories. Because their acquisition rules differ, this is a complete-policy comparison rather than a refit-only ablation.

Audited-union regret is computed post hoc against the best configuration evaluated by either policy on the same seed. The union is not visible during search and does not reveal outcomes for unrequested configurations.

The endpoint difference disappears at budgets 16 and 24, while Refitted Ridge-LCB obtains approximately 12% lower normalized anytime regret AUC over the complete trajectory. The result supports faster entry into high-quality regions rather than guaranteed endpoint superiority.

Table 18: Seed-level prospective E2E results at budget 12. Differences are Refitted Ridge-LCB minus Frozen Ridge (mean) best validation loss; negative values favor Refitted Ridge-LCB.

Seed	Frozen cand.	Frozen loss	Refitted cand.	Refitted loss	Difference	Outcome
400	q55	1.5842	q27	1.5738	−0.0105	Win
401	q55	1.6010	q55	1.6010	0.0000	Tie
402	q40	1.6371	q07	1.6321	−0.0050	Win
403	q27	1.5720	q27	1.5720	0.0000	Tie
404	q55	1.5654	q55	1.5654	0.0000	Tie
Mean	–	1.5919	–	**1.5888**	**−0.0031**	2 wins / 3 ties

Table 19: Aggregate prospective E2E results for Refitted Ridge-LCB and Frozen Ridge (mean). Budgets 12 and 16 are prefixes of the same budget-24 trajectories.

Budget	Frozen best ↓	Refitted best ↓	Frozen AUC ↓	Refitted AUC ↓	Endpoint
12	1.5919	**1.5888**	0.0185	0.0185	2 wins / 3 ties
16	1.5819	1.5819	0.0121	**0.0106**	Same candidates
24	1.5819	1.5819	0.0064	**0.0056**	Same candidates

C.6 Prospective SST-2 Check

The SST-2 experiment uses 16 configurations, four initial observations, a budget of eight evaluations, and three prospective seeds. Refitted Ridge-LCB and Frozen Ridge (mean) receive the same initial observations and outcome-revelation protocol.

Table 20: Prospective SST-2 results over seeds 370–372. Final validation loss is mean ± sample standard deviation.

Policy	Final validation loss ↓	Trajectory-average loss ↓
Refitted Ridge-LCB	$\mathbf{0.3118 \pm 0.0052}$	$\mathbf{0.3207}$
Frozen Ridge (mean)	$\mathbf{0.3118 \pm 0.0052}$	0.3215
TPE	0.3157 ± 0.0109	0.3222
Random search	0.3259 ± 0.0116	0.3332

Refitted Ridge-LCB and Frozen Ridge (mean) return the same final recommendation on all three seeds. The slightly lower trajectory-average loss of Refitted Ridge-LCB is descriptive and does not establish endpoint superiority.

C.7 Statistical Conventions

Replay comparisons use the shared initial design as the paired unit. Final-regret intervals in Table 13 are obtained by resampling paired trial-level differences rather than resampling policies independently. Prospective experiments use training seeds as the paired unit and report seed-level wins, ties, losses, and trajectory summaries because five E2E seeds and three SST-2 seeds do not support broad distribution-free superiority claims.

For the full-budget experiments in Appendix D, differences are computed within training seed before aggregation, and reported standard deviations are sample standard deviations across seeds. Bootstrap intervals resample paired seed-level differences. The complete LoRA-r4 quality reference and optional warm-start analyses are also reported in Appendix D. Held-out examples are not treated as independent replicates, and held-out generation outcomes are never used for surrogate fitting, candidate promotion, or policy selection.

D Fidelity, Full-Budget, and Optional Transfer Audits

This appendix reports the full-protocol evidence supporting the configuration-selection results in Section 4. It covers low-to-full-fidelity agreement, paired confirmation of frozen selections, resource-constrained selection, modern-backbone trade-offs, task-specific adaptation, and the optional warm-start component. Search policies and candidate identities were frozen before the held-out outcomes reported below were inspected.

D.1 Low-to-Full-Fidelity Audit

We evaluate 20 prespecified E2E configurations using three full-protocol seeds per configuration, producing 60 full-training runs. Table 21 compares the resulting validation-loss ranking with the frozen low-fidelity ranking.

The strong rank agreement supports low-fidelity evaluation as a screening mechanism within this frozen pool. Because the ranking is not perfect, all final task-quality claims use independent full-budget retraining rather than low-fidelity outcomes alone.

D.2 Primary Full-Budget Confirmation

The primary GPT-2/E2E comparison freezes `c11` after search and evaluates it against the pre-existing `c00` configuration using paired seeds 321–325. Both configurations have the same adapter structure, target modules, 73,632 trainable parameters, and evaluation protocol; their only difference is the learning rate, which changes from 0.002 to 0.004.

The paired 95% intervals are $[-0.0540, -0.0485]$ for validation loss, $[2.7654, 5.0100]$ for BLEU, $[0.5679, 0.9089]$ for NIST, $[2.4557, 3.2273]$ for METEOR, $[1.3656, 2.0158]$ for ROUGE-L, and $[0.1264, 0.2329]$ for CIDEr. All five seed pairs favor `c11` on validation loss, BLEU, and ROUGE-L. This experiment demonstrates successful selection of a consequential training setting, while the structural contribution is tested separately below.

Table 21: Low-to-full-fidelity ranking audit over 20 prespecified configurations.

Statistic	Result
Configurations / full-training runs	$20/60$
Spearman correlation	$\mathbf{0.9308}$ ($p = 2.67 \times 10^{-9}$)
Kendall correlation	$\mathbf{0.7789}$ ($p = 5.84 \times 10^{-8}$)
Low-fidelity Top-4 recall	75%
Full-protocol Top-4	`q55`, `q56`, `q40`, `q31`
Low-/full-protocol rank of `q55`	$\mathbf{1/1}$
Promotion regret of `q55`	$\mathbf{0}$

Table 22: Five-seed full-budget GPT-2/E2E confirmation. Values are means over paired seeds 321–325; differences are `c11` minus `c00`.

Configuration	Val. loss ↓	BLEU ↑	NIST ↑	METEOR ↑	R-L ↑	CIDEr ↑	Params.	Time (s)	Mem. (MB)
Fixed `c00`	1.506	30.381	4.159	31.875	47.913	1.078	73,632	410.79	921.78
Selected `c11`	**1.454**	**34.268**	**4.900**	**34.708**	**49.600**	**1.255**	73,632	**408.55**	921.78
Paired difference	$\mathbf{-0.051}$	$\mathbf{+3.888}$	$\mathbf{+0.740}$	$\mathbf{+2.833}$	$\mathbf{+1.688}$	$\mathbf{+0.178}$	0	-2.24	0

D.3 Structural and Promotion Controls

To separate structural selection from learning-rate and adapter-capacity effects, we compare `q27` with a control having the same learning rate, rank, intrinsic rank, target modules, layer groups, and 110,448 trainable parameters. The two configurations differ in the remaining structural choices, including scale and Taylor order. We also confirm the low-fidelity winner `q55` against its frozen anchor and compare it with the larger `q56` configuration.

The matched `q27` result shows that the gain is not explained by learning rate or parameter count: validation loss, BLEU, and ROUGE-L improve in all five pairs with only a 1.27 MB mean memory increase. Relative to the original smaller fixed configuration, `q27` improves BLEU by 5.202 and ROUGE-L by 2.398, although that comparison also includes a capacity increase. The promotion result provides a separate fidelity check: `q55` remains the strongest of `q00`, `q55`, and `q56`, despite `q56` having the largest adapter.

D.4 Resource-Constrained Recommendation

Under the 75% trainable-parameter cap, feedback-refitted acquisition and constrained QUBO compilation select `q07`. We freeze this decision and compare `q07` with reference configuration `q00` using five new paired full-budget seeds.

Validation loss improves in all five pairs, with a paired interval of $[-0.01253, -0.00464]$. Training time decreases in all five pairs by 52.4%, with an interval of $[-357.18, -352.90]$ seconds, and trainable parameters decrease by 33.4%. BLEU and ROUGE-L differences are inconclusive: their intervals are $[-2.012, 0.499]$ and $[-1.189, 0.278]$, respectively. Peak memory increases by 25.19 MB, or approximately 2.7%. The result therefore supports improved validation objective and resource efficiency, but not improved held-out generation metrics.

D.5 Modern-Backbone Quality–Efficiency Trade-Off

The configuration selected on the fixed-learning-rate Qwen2.5-1.5B structural pool is evaluated against the fixed configuration using ten independent paired held-out seeds (704–713).

The selected configuration consistently improves the optimized validation objective while using 28.6% fewer trainable parameters and 41.0% less training time. Its held-out effect is metric dependent: ROUGE-L improves, whereas BLEU decreases slightly. This result motivates multi-objective acquisition when several generation metrics must be optimized simultaneously.

Table 23: Paired full-budget controls for structural selection and candidate promotion. Confidence intervals are paired bootstrap 95% intervals.

Comparison	Δ Val. loss ↓	Δ BLEU ↑	Δ R-L ↑	Joint wins	Resource difference
q27 vs. matched control	$\mathbf{-0.0640}$ $[-0.0661, -0.0619]$	$\mathbf{+2.6377}$ $[2.1112, 3.1531]$	$\mathbf{+1.3390}$ $[1.0083, 1.6968]$	5/5	Same LR/params.; +18.3% time
q55 vs. anchor q00	$\mathbf{-0.0660}$	$\mathbf{+2.7079}$ $[1.8647, 3.4790]$	$\mathbf{+1.2781}$ $[0.8655, 1.8425]$	5/5	+55,080 params.; −38.3% time
q56 vs. anchor q00	$\mathbf{-0.0344}$	$\mathbf{+1.5188}$ $[0.8390, 2.0843]$	$\mathbf{+0.8352}$ $[0.2713, 1.3580]$	4/5	+110,304 params.; −36.4% time

Table 24: Five-seed confirmation of the configuration selected under the 75% trainable-parameter cap. Values are mean ± standard deviation.

Configuration	Val. loss ↓	BLEU ↑	R-L ↑	Params. ↓	Time (s) ↓	Mem. (MB) ↓
Reference q00	1.4801 ± 0.0188	25.623 ± 1.426	48.638 ± 1.219	55,368	677.23 ± 3.87	918.15 ± 14.42
Selected q07	$\mathbf{1.4715 \pm 0.0169}$	25.133 ± 1.545	48.188 ± 0.532	**36,864**	$\mathbf{322.35 \pm 1.80}$	943.33 ± 14.41
Paired difference	$\mathbf{-0.0086}$	-0.490	-0.450	$-18{,}504$	$\mathbf{-354.88}$	$+25.19$

Table 25: Ten-seed Qwen2.5-1.5B held-out confirmation. Differences are selected m03 minus fixed m00.

Metric	Fixed m00	Selected m03	Paired difference	95% interval	Wins
Validation loss ↓	1.3380	**1.2922**	$\mathbf{-0.0458}$	$[-0.0478, -0.0436]$	10/10
BLEU ↑	**28.3291**	28.0057	-0.3234	$[-0.6217, -0.0356]$	3/10
ROUGE-L ↑	48.1515	**48.6959**	$\mathbf{+0.5444}$	$[0.2539, 0.8806]$	8/10
Time (s) ↓	358.0	**211.3**	$\mathbf{-146.7}$	$[-148.2, -145.2]$	10/10
Peak memory (MB) ↓	3183.5	**3182.3**	-1.24	–	10/10
Trainable parameters ↓	136,192	**97,200**	-28.6%	–	–

D.6 Task-Specific Adaptation

A configuration frozen on E2E transfers positively to DART but not uniformly to WebNLG or CommonGen: the mean BLEU/ROUGE-L changes are $+0.599/+0.584$ on E2E, $+1.136/+1.414$ on DART, $-0.302/-0.194$ on WebNLG, and $-0.322/-0.029$ on CommonGen. This task dependence motivates rerunning configuration selection rather than transferring a universal preset.

In the task-specific WebNLG study, the selected configuration improves BLEU by 0.8449 with a paired 95% interval of $[0.0224, 1.6673]$ and improves ROUGE-L by 1.1282 with an interval of $[0.5684, 1.7108]$. Four of five seeds improve both metrics. The contrast with frozen cross-task transfer supports task-adaptive selection.

D.7 Optional Warm-Start Transfer

Warm-start transfer is optional and is not required by the feedback, QUBO-compilation, or reliability-gating results. The following experiment evaluates the transfer mechanism on Qwen2.5-1.5B/E2E using eight paired seeds and the same 400 held-out examples.

Relative to Direct Q-PEFT, LoRA-SVD-QPEFT improves BLEU by 0.6839 with a paired 95% interval of $[0.3362, 1.0345]$ and ROUGE-L by 0.5305 with an interval of $[0.3282, 0.7327]$.

Relative to LoRA-r4, the paired BLEU difference is -0.0192 with a 95% interval of $[-0.2788, 0.2324]$, and the paired ROUGE-L difference is $+0.0301$ with an interval of $[-0.2459, 0.2741]$. Both lower confidence limits remain above the prespecified -0.5 non-inferiority margin. LoRA-SVD-QPEFT therefore satisfies the stated BLEU and ROUGE-L non-inferiority criterion in this eight-seed E2E protocol while using 272,160 rather than 544,768 trainable parameters, or 49.96% as many as LoRA-r4. This result supports quality preservation with fewer retained parameters and improved performance over Direct Q-PEFT; it does not establish superiority over LoRA-r4 or reduced end-to-end training time.

Coordinate-fitting ablation. The no-fit control uses the same rank-2 SVD targets and retained singular energy but performs no coordinate-fitting updates.

Table 26: Eight-seed held-out evaluation of the optional warm-start mechanism on Qwen2.5-1.5B/E2E. Values are mean ± standard deviation.

Method	n	BLEU ↑	ROUGE-L ↑	Trainable params. ↓
Direct Q-PEFT	8	33.166 ± 0.278	49.078 ± 0.309	272,160
LoRA-r2	8	33.632 ± 0.454	49.402 ± 0.372	272,160
LoRA-SVD-LoRA-r2	8	33.537 ± 0.574	49.442 ± 0.473	272,160
LoRA-r4	8	$\mathbf{33.869 \pm 0.370}$	49.578 ± 0.333	544,768
LoRA-SVD-QPEFT	8	33.850 ± 0.310	$\mathbf{49.609 \pm 0.166}$	272,160

Table 27: Eight-seed ablation of coordinate fitting. Both arms use seeds 421–428 and identical held-out examples.

Mapping	Fit steps	Residual ↓	Retained energy	Val. loss ↓	BLEU ↑	R-L ↑
No fit	0	1.4142	0.7351	1.2145	33.1791	49.1876
Optimized	200	**0.0392**	0.7351	**1.1398**	**33.8501**	**49.6085**

Optimized fitting reduces validation loss by 0.0747 with a paired interval of $[-0.0769, -0.0724]$, improves BLEU by 0.6710 with an interval of $[0.3820, 0.9288]$, and improves ROUGE-L by 0.4210 with an interval of $[0.0962, 0.7068]$. These results show that coordinate fitting is beneficial under the evaluated initialization and protocol. Retained SVD energy alone does not explain the observed downstream differences.

Warm-start configuration audit. Table 28 reports the frozen GPT-2/E2E search-stage candidates. These results are descriptive because each row corresponds to one search seed.

Table 28: Candidate-level audit of optional LoRA-SVD transfer. Loss is search-stage validation loss, η_r is retained SVD energy, and fit error is the normalized mapping residual.

ID	ρ	r	k	p	α	LR	Loss ↓	η_r	Fit error ↓
c00	.50	2	2	3	32	.002	1.5654	.9339	.0318
c01	.50	2	1	3	32	.002	1.6111	.9339	.6782
c02	.50	3	2	3	32	.002	1.5671	.9826	.4487
c03	.25	2	2	3	32	.002	1.6109	.9270	.4209
c04	.75	2	2	3	32	.002	**1.5501**	.9316	.1418
c05	.50	4	2	3	32	.002	1.5679	1.0000	.5110
c07	.50	2	2	3	64	.002	1.5628	.9339	.6649
c10	.50	2	2	3	32	.001	1.5849	.9339	.0318
c11	.50	2	2	3	32	.004	**1.5499**	.9339	.0318

Among the otherwise matched warm-start fractions, validation loss decreases from 1.6109 at $\rho = .25$ to 1.5654 at $\rho = .50$ and 1.5501 at $\rho = .75$. Rank 4 retains all measured SVD energy but does not improve validation loss over the compact candidates. Retained energy and mapping residual are therefore useful diagnostics, but neither substitutes for downstream validation.

E Task Dependence and Reselection under Distribution Shift

This appendix evaluates whether an E2E-selected Q-PEFT configuration transfers unchanged to other generation tasks and whether target-task feedback can recover a stronger configuration after transfer degrades. These experiments concern configuration selection rather than the optional warm-start mechanism.

E.1 Frozen Cross-Task Transfer

We first apply the frozen E2E-selected configuration to DART (Nan et al., 2021), WebNLG (Gardent et al., 2017), and CommonGen (Lin et al., 2020) without target-task reselection. Each transferred

configuration is compared with the corresponding task-specific incumbent under the same model, seed, data, decoding, and evaluation protocol.

Table 29: Frozen transfer of the E2E-selected configuration. Values are transferred minus task-specific incumbent; positive values favor the transferred configuration.

Target task	n	Δ BLEU ↑	Δ ROUGE-L ↑
E2E-NLG	3	$+0.5985$	$+0.5844$
DART	3	$+1.1361$	$+1.4141$
WebNLG	3	-0.3018	-0.1942
CommonGen	5	-0.3215	-0.0294

The frozen configuration improves both reported metrics on E2E-NLG and DART but not on WebNLG or CommonGen. A configuration selected for one data distribution therefore need not remain preferable after the task or generation objective changes.

E.2 WebNLG Reselection

We next expose the selection loop to WebNLG validation outcomes and freeze the resulting task-specific configuration before held-out evaluation. The selected configuration and the fixed WebNLG incumbent are retrained using five paired seeds.

Table 30: Frozen transfer and WebNLG-specific reselection. Deltas are relative to the fixed WebNLG incumbent. Intervals are paired bootstrap 95% intervals for the five-seed reselection experiment.

Protocol	n	Δ BLEU ↑	Δ ROUGE-L ↑	Joint wins
Frozen E2E configuration	3	-0.3018	-0.1942	$1/3$
WebNLG-selected configuration	5	$\mathbf{+0.8449}$	$\mathbf{+1.1282}$	$\mathbf{4/5}$
Task-specific 95% interval	–	$[+0.0224, +1.6673]$	$[+0.5684, +1.7108]$	–

The WebNLG-selected configuration improves ROUGE-L on all five paired seeds and improves both BLEU and ROUGE-L on four. The BLEU differences are -0.4775, $+0.0958$, $+1.5537$, $+0.8752$, and $+2.1771$; the ROUGE-L differences are $+0.5236$, $+0.3145$, $+1.1659$, $+2.1967$, and $+1.4405$. Both paired intervals exclude zero.

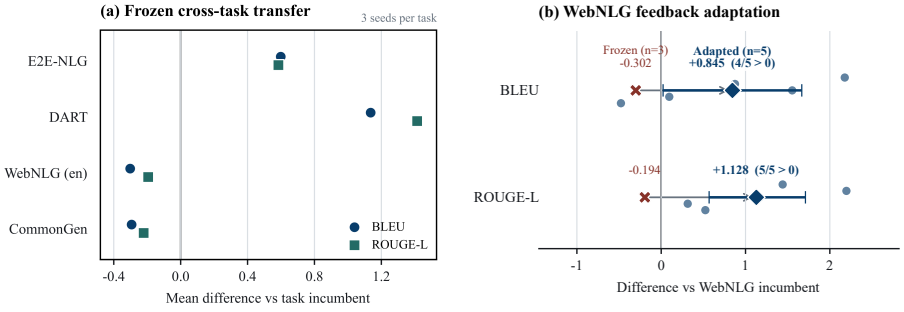


Figure 4: Task dependence and recovery through reselection. Left: frozen transfer of the E2E-selected configuration. Right: paired WebNLG differences after task-specific selection. The frozen-transfer and reselection protocols use different seed sets and are not compared as paired observations.

Figure 4 separates static transfer from target-task adaptation. The negative frozen WebNLG means show that an E2E-selected preset does not transfer uniformly, while WebNLG validation feedback identifies a configuration with positive mean BLEU and ROUGE-L differences. One seed retains a negative BLEU difference, so the result supports improved average task-specific adaptation rather than universal per-seed dominance.

E.3 SCOPE

These experiments support treating Q-PEFT adaptation as task-conditional configuration selection rather than assuming a universally transferable preset. The evidence covers four generation datasets and modest paired samples under frozen GPT-2 protocols; it does not establish universal transfer failure or guarantee that reselection improves every task. Its role is to show that distribution shift can change the preferred configuration and that target-task feedback can recover performance when frozen transfer is unfavorable.

F RELIABILITY-GATED ISING AND REAL-QPU AUDIT

Language-model training and downstream evaluation are performed on conventional GPUs. Ising and QPU backends retrieve candidates from frozen discrete objectives but do not compute model gradients or generation metrics. This appendix reports matched-shot controls, the four-round TianYan-176 sequential replay, a decision-changing end-to-end audit, preregistered-target audits, scale and routing diagnostics, gate ablations, and completion of the deterministic classical route.

The hardware experiments use the two support-set instantiations defined in Appendix A.4. Preregistered-target audits set $\mathcal{A}_t = \mathcal{T}_t$, whereas target-free execution sets $\mathcal{A}_t = \{x_{t+1}^{\mathrm{QPU}}\}$. Frequency-shortlist support is distinct from a state being sampled at least once and from pooled probability mass on an exact-energy Top-K region.

F.1 MATCHED-SHOT ISING SAMPLING CONTROLS

We test whether the early-coverage behavior of ideal Ising retrieval can be explained by generic randomization or low-energy sampling. All methods use the same 39 frozen candidates, 1,000 paired initial designs, eight initial observations, total budget 12, 1,024 samples per adaptive step, Top-6 shortlist, predicted-energy gate, and surrogate frozen after warm-up.

Table 31: Matched-shot backend replay at budget 12. Recovery is conditioned on an initial Top-10% miss.

Retrieval backend	Regret ↓	Top-10% hit ↑	Recovery ↑	Best hit ↑
Shot-based ideal Ising/QAOA	**0.04172**	**96.3**%	**89.9**%	20.9%
Randomized Top-k	0.05296	84.5%	57.7%	27.5%
Boltzmann/Gibbs	0.05524	83.3%	54.4%	25.4%
Simulated annealing	0.05524	83.3%	54.4%	25.4%
Deterministic energy ranking	0.05524	83.3%	54.4%	25.4%
Random search	0.05773	78.9%	42.3%	**29.9%**

Relative to Boltzmann sampling, ideal Ising/QAOA has a paired regret difference of -0.01352 with a 95% interval of $[-0.01596, -0.01115]$, and improves Top-10% hit rate and recovery by 13.0 and 35.5 percentage points. Relative to randomized Top-k, the corresponding improvements are -0.01125 regret, 11.8 points in Top-10% hit rate, and 32.2 points in recovery; all paired intervals exclude zero.

The advantage concerns early high-quality-region coverage rather than exact global-best recovery. Ideal Ising/QAOA has a lower global-best hit rate than several more diffuse controls. Boltzmann sampling, simulated annealing, and deterministic energy ranking produce identical aggregate results because the frozen 1,024-sample Top-6 protocol recovers the same shortlist in this small candidate space.

F.2 FOUR-ROUND TIANYAN-176 SEQUENTIAL REPLAY

We evaluate repeated hardware participation on the 32-configuration fixed-learning-rate Qwen2.5-0.5B pool. The protocol begins with eight fixed observations and executes four adaptive rounds. In each round, Ridge-LCB is refitted using only the outcomes revealed so far, a new eight-candidate, seven-qubit domain-wall Ising instance is compiled, and TianYan-176 is executed in three independent 2,048-shot jobs.

The target-free gate sets $\mathcal{A}_t = \{x_{t+1}^{\mathrm{QPU}}\}$ and requires the proposal to be legal, repeat-supported in at least two of three executions, and associated with mean pairwise TV no greater than 0.25. Candidate labels are prespecified five-seed mean low-fidelity validation losses and remain hidden until requested.

As reported in Table 5, all four rounds pass the frozen gate. Across 12 hardware jobs and 24,576 shots, every routed candidate is legal and repeat-supported. The trajectory reaches empirical pool optimum `s19` in round 2 and queries four of the five best configurations by round 4.

The QPU and deterministic classical routes agree in all four rounds. This experiment therefore establishes repeated real-QPU participation under feedback-updated objectives and preservation of the constrained acquisition decision, rather than superiority over classical retrieval. Because the revealed outcomes are archived five-seed means, this is a sequential hardware replay rather than prospective GPU retraining after every QPU decision.

F.3 TianYan-176 Decision-Changing End-to-End Audit

Before hardware execution, we freeze the eligible candidate pool, seven-qubit Ising objective and coefficients, decoder and bit order, frequency Top-4 support rule, repeat-support threshold, TV threshold $\tau_{\mathrm{TV}} = 0.25$, deterministic tie breaking, and classical candidate `p05`. The frozen instance is then executed in three independent 2,048-shot repetitions. Hardware candidate `p01` enters the frequency Top-4 in all three repetitions, and the mean pairwise TV is 0.1165. The gate therefore accepts `p01`; the routing decision is frozen before downstream training on seeds 671–673. Candidate `p01` differs from deterministic classical candidate `p05`, both contain 272,160 trainable parameters, and held-out generation outcomes are not inspected during routing.

Table 32: Seed-level downstream results for the decision-changing TianYan-176 audit. Candidate selection is frozen before these outcomes are observed.

Seed	Candidate	BLEU	ROUGE-L
671	QPU `p01`	33.9007	49.9687
671	Classical `p05`	34.2730	49.4453
672	QPU `p01`	33.7288	49.5022
672	Classical `p05`	33.7758	49.4421
673	QPU `p01`	34.4598	50.2035
673	Classical `p05`	34.3422	50.2939

Table 33: Fresh-seed downstream comparison of accepted TianYan-176 candidate `p01` and deterministic classical candidate `p05`. Differences are `p01`$-$`p05`. Intervals are paired bootstrap 95% confidence intervals from 20,000 resamples. The protocol-level -0.5-point non-inferiority margin was fixed before held-out evaluation and is not treated as a universal benchmark threshold.

Metric	Paired difference	95% interval	Non-inferior
BLEU	-0.101	$[-0.372, 0.118]$	Yes
ROUGE-L	$+0.164$	$[-0.090, 0.523]$	Yes

Across three new paired full-budget Qwen2.5-1.5B/E2E-NLG seeds, both confidence-interval lower bounds exceed the prespecified non-inferiority margin. All six unique downstream training runs complete. The accepted QPU proposal therefore changes the routed configuration while satisfying the protocol-level downstream-quality criterion. Because both intervals include zero, the result does not establish improvement over classical retrieval or quantum speedup.

F.4 TianYan-176 Preregistered-Target Audit

A separate TianYan-176 audit uses eight completed Qwen2.5-0.5B/E2E configurations encoded as legal states of a seven-spin domain-wall encoding (Chancellor, 2019). Exact enumeration identifies `c01` as the feasible ground-state candidate. Candidate identities, Ising coefficients, decoder, bit order, target set, frequency-shortlist rule, and integrity hashes are frozen before hardware submission.

Under the enhanced protocol, `c01` enters the frequency Top-4 in three of five repetitions and passes the stability requirement. Because measurement order, circuit angles, optimization budget, and shot

Table 34: Frozen TianYan-176 preregistered-target outcomes. Frequency Top-4 counts repetitions in which the target enters the four most frequent legal decoded configurations.

Protocol	Sampling	Proposal	Exact rank	Hardware evidence	Decision
Initial	Low budget	`c00`	4	Below frozen Top-3	Reject
Enhanced	$5 \times 2{,}048$	`c01`	**1**	Freq. Top-4 in $3/5$; TV 0.1306	**Accept**

count change jointly relative to the initial run, this is treated as a frozen operational follow-up rather than a single-factor ablation.

F.5 Scale and Routing Diagnostics

We execute nested 4-, 6-, and 8-variable conditional Ising instances on TianYan-176. Each scale receives three independent 1,024-shot repetitions. The separately frozen diagnostic rule requires completion of all repeats, at least one ground-state sample in two or more repetitions, a normalized best-energy gap no greater than 5%, and mean pairwise TV no greater than 0.25.

Table 35: TianYan-176 scale audit. Ground-state probabilities are pooled over three 1,024-shot repetitions.

Variables	CX/layer	Ideal ground p	QPU ground p	Mean TV	Gate
4	6	0.1634	0.1032	0.2095	**Accept**
6	10	0.0410	0.0127	0.3313	Reject
8	14	0.0188	0.0049	0.3502	Reject

Only the four-variable instance passes the complete rule. We therefore expose the 16-variable objective through seven conditional four-variable windows rather than treating the complete instance as one accepted optimization. All seven windows complete three repetitions, three pass every diagnostic component, and one accepted update reduces the complete objective from -0.62008673 to -0.66497243. Unsupported or non-improving proposals are discarded.

Full 16-variable TianYan-287 stress test. A separate experiment retains all 16 variables and 54 quadratic couplings. Nonlocal interactions are implemented using shortest-path SWAP routing, with the logical-to-physical assignment restored after each interaction.

Table 36: Full 16-variable TianYan-287 routing stress test. The matched-shot reference uses 10,000 ideal 1,024-shot draws.

Metric	Value	Metric	Value
Logical / physical qubits	$16/16$	Ising couplings	54
Round-trip SWAPs	86	Routed CX gates	366
QCIS instructions	1,216	QAOA depth / shots	$1/1{,}024$
Ideal ground p/shot	2.8065×10^{-6}	QPU ground hits	$0/1{,}024$
Exact ground energy	-0.624248	Best sampled energy	-0.510881
Hardware-to-ideal TV	0.9828	Matched-shot TV mean	0.9500
Matched-shot 95% interval	$[0.9473, 0.9526]$	Reference draws ≥ 0.9828	$0/10{,}000$
Gate outcome	Reject		

The circuit completes and returns all 1,024 samples. Zero ground-state hits occur with probability 0.9971 even under the ideal distribution and are therefore not diagnostic alone. However, the hardware TV exceeds all 10,000 matched-shot ideal draws, and the best sampled energy remains above the exact optimum. The routed proposal fails the frozen diagnostic rule.

F.6 PROSPECTIVE TIANYAN-287 GATE AUDIT

This audit uses a separate eight-candidate pool p00–p07. Candidate p05 is the preregistered target and deterministic classical candidate. Three independent TianYan-287 executions return 2,048 shots each. The decoder, circuit, target, shortlist size, thresholds, and classical route are frozen before hardware submission.

Across 6,144 pooled shots, the ground-state probability is 2.65% with a Wilson 95% interval of $[2.28\%, 3.09\%]$, exact-energy Top-4 mass is 12.50% with an interval of $[11.70\%, 13.35\%]$, and feasible-state probability is 28.03% with an interval of $[26.92\%, 29.16\%]$. These quantities are diagnostics rather than the gate's frequency-support statistic.

Target p05 is sampled 56, 59, and 48 times but does not enter the frequency Top-4 in any repetition. Its frequency support is therefore $0/3$, although it appears in every run. Mean pairwise TV is 0.1328 and passes the stability threshold.

Table 37: Frozen prospective TianYan-287 gate evidence. Frequency support is distinct from pooled exact-energy Top-4 mass.

Component	Frozen requirement	Observed result
Sampling	Three repeats, 2,048 shots each	$3 \times 2{,}048$ completed
Proposal legality	Legal and complete	Pass (p02)
Feasible-state mass	Diagnostic only	28.03%
QPU proposal	Lowest-score retained legal candidate	p02, exact rank 3
Target observations	Archived per repetition	$56,\ 59,\ 48$ samples
Frequency support	Target in frequency Top-4 in at least $2/3$	$0/3$
Repeat stability	Mean TV ≤ 0.25	0.1328
Gate decision	Legality, frequency support, and stability	Reject p02
Classical route	Frozen deterministic candidate	p05

A component replay isolates the decisive gate condition.

Table 38: TianYan-287 gate-component ablation using the frozen hardware observations.

Enabled gate components	Routed decision
None	Accept p02
Legality only	Accept p02
Legality and TV	Accept p02
Legality and frequency support	Route p05
Complete gate	Route p05

Frequency support, rather than illegality or repeat instability, determines the fallback. Stable distributions and occasional target samples are insufficient when the preregistered target does not enter the frozen frequency shortlist.

Execution time. The complete prospective workflow takes 42 seconds from creation to completion; 11 seconds elapse between the submission receipt and completion of all three tasks. The logs do not separately identify queueing, circuit execution, decoding, and gate time, so no finer timing decomposition is reported.

F.7 DOWNSTREAM COMPLETION OF THE CLASSICAL ROUTE

After the gate decision is frozen, deterministic candidate p05 is evaluated using seeds 621–623. The hardware-controlled path and classical closed-loop control use the same immutable p05 runs because the QPU proposal is rejected; they are not independent observations. Consequently, the three protocol arms produce nine arm–seed entries from six unique downstream training runs.

All six unique downstream training runs complete and populate all nine arm–seed entries. Relative to the fixed incumbent, p05 has an inconclusive BLEU difference of -0.0197, with a paired interval of

Table 39: Blinded downstream evaluation after the TianYan-287 gate decision. The nine arm–seed entries are populated from six unique training runs because the hardware-controlled and classical-control arms share the same immutable `p05` runs. Values are mean ± sample standard deviation over seeds 621–623.

Protocol arm	Evaluated candidate	BLEU	ROUGE-L
Hardware path after gate	`p05`	33.6916 ± 0.2641	49.3325 ± 0.1105
Classical closed loop	`p05`	33.6916 ± 0.2641	49.3325 ± 0.1105
Fixed incumbent	`fixed_c01`	33.7113 ± 0.3063	49.3947 ± 0.0853

$[-0.0733, 0.0606]$. Its ROUGE-L is lower by 0.0622, with an interval of $[-0.1558, -0.0007]$, but the decrease remains within the prespecified 0.5-point non-inferiority margin. This result validates completion of the deterministic route and is not attributed to the rejected QPU proposal.

F.8 Retrospective Counterfactual Audit

To assess the downstream consequence of the TianYan-287 rejection, we retrospectively retrain rejected proposal `p02` and routed classical candidate `p05` on the same three paired seeds.

Table 40: Retrospective full-budget comparison of rejected QPU proposal `p02` and routed classical candidate `p05`. Differences are `p02`−`p05`; intervals are paired bootstrap 95% confidence intervals over three seeds.

Metric	`p02`	`p05`	Difference	95% interval
Validation loss ↓	1.138512	1.139042	$\mathbf{-0.000530}$	$[-0.001361, -0.000062]$
BLEU ↑	33.705	33.692	$+0.014$	$[-0.156, +0.124]$
ROUGE-L ↑	49.309	49.333	-0.024	$[-0.058, +0.044]$
Training time (s) ↓	4,224	4,181	$+43$	$[+16, +76]$
Trainable parameters ↓	272,216	272,160	$+56$	–
Peak memory (MB) ↓	6,146	6,146	0	–

The rejected proposal is downstream-quality competitive with the deterministic fallback: neither BLEU nor ROUGE-L shows a detected paired difference, while `p02` obtains slightly lower validation loss on all three seeds. The classical route uses 56 fewer trainable parameters and is approximately 1.0% faster, with identical peak memory. The TianYan-287 decision should therefore be interpreted as a conservative rejection under the frozen support rule, not as evidence that the rejected proposal was inferior. This case exposes a safety–coverage trade-off and motivates future gates that combine sampling reliability with score competitiveness.

F.9 Scope of the Hardware Evidence

The matched-shot replay supports an early high-quality-region coverage advantage for the ideal Ising distribution, but not exact global-best recovery. The four-round TianYan-176 replay demonstrates repeated real-QPU participation under feedback-updated objectives, while agreement with the deterministic route limits that result to decision preservation. The separate end-to-end audit shows that an accepted TianYan-176 proposal can differ from the deterministic classical candidate and remain non-inferior after fresh-seed full-budget evaluation. Preregistered-target audits exercise both acceptance and rejection, while the scale and routing studies identify reliability degradation as problem size and routing overhead increase. TianYan-287 further confirms that a stable but unsupported proposal is rejected and that the deterministic route completes the downstream schedule.

Together, these experiments establish auditable QPU participation, decision-changing accepted retrieval, and deterministic containment of unsupported proposals under the evaluated protocols. They do not establish quantum speedup, universal backend superiority, or a need for QPU execution to complete the closed loop.

G Reproducibility and Anonymous Artifact

The anonymous artifact separates immutable observations from reconstruction code. Reported tables, figures, paired statistics, search trajectories, QUBO audits, and hardware decisions can be reconstructed from archived records without rerunning model training or submitting new QPU jobs. Full retraining remains available through the frozen workflow definitions.

G.1 Artifact Organization

The anonymous artifact is organized as follows:

- `src/qpeft_opt/`: training, candidate encoding, surrogate refitting, QUBO/Ising compilation, decoding, gating, and hardware interfaces.
- `configs/`: frozen model, dataset, candidate-space, resource-budget, search, decoding, and hardware protocols.
- `scripts/workflows/`: training, replay, prospective search, aggregation, hardware analysis, and paper reconstruction.
- `artifacts_metrics/`: immutable run-level metrics, configuration metadata, generation outputs, runtime, memory, and adapter diagnostics.
- `results/`: candidate pools, trajectories, paired observations, QUBO audits, hardware counts, and protocol manifests.
- `results/final_iclr_summary/`: canonical CSV, JSON, and figure inputs used by the paper.
- `tests/`: encoding, constraint, conversion, decoding, routing, gating, and workflow-consistency tests.

The artifact excludes pretrained weights, dataset caches, private hardware credentials, and server-specific paths. Public models and datasets are identified by frozen configurations, while archived numerical observations are sufficient to audit the reported results.

G.2 Evidence-to-Artifact Mapping

Every replay record contains its initial design, feasible pool, isolated policy state, query sequence, revealed outcomes, budget, and final recommendation. Full-budget studies retain both paired arms at the seed level.

Table 41: Archived evidence for the principal experiments.

Experiment	Archived evidence
Feedback replay and BO controls	Frozen 64-candidate outcomes, three resource caps, 1,000 paired initial designs, policy trajectories, paired bootstrap differences, structured GP-EI/RF-EI results, and GP sensitivity diagnostics.
Modern-model structural search	Qwen2.5-0.5B and Qwen2.5-1.5B outcome pools, fixed-learning-rate spaces, replay seeds, initial designs, and budget traces.
Prospective E2E and SST-2	Chronological queries, isolated policy states, revealed outcomes, cached-label indices, and evaluated budget prefixes.
Fidelity and full-budget tests	Fidelity-audit records and seed-level `c11`, `q27`, `q55`, `q07`, and `m03` results with paired intervals and resource measurements.
Transfer and WebNLG reselection	Frozen cross-task outcomes, target-task search history, paired WebNLG confirmations, and bootstrap summaries.
QUBO and Ising audits	Resource caps, unconstrained and constrained decisions, exact-enumeration checks, matched-shot trials, and paired backend comparisons.
Real-QPU execution	Four-round TianYan-176 trajectories, round-specific acquisition and Ising instances, raw hardware counts, frequency shortlists, support and TV statistics, accepted and fallback decisions, and TianYan-287 downstream completion records.
Optional warm-start transfer	Seed-level comparisons, retained energy, coordinate-fitting residuals, parameter counts, and mapping ablations.

Canonical paper values are reconstructed from `results/final_iclr_summary/`. Screenshots and exploratory reports are not used as paper-level numerical sources.

G.3 Environment and Validation

Python dependencies and installable project metadata are frozen in `requirements.txt` and `pyproject.toml`. Appendix B summarizes the protocol-relevant training and hardware settings. Experiment-specific execution metadata—including the operating system, Python and PyTorch/CUDA versions, GPU model, and, for QPU runs, SDK/compiler versions and archived job identifiers—are stored in the corresponding artifact manifests and frozen configuration records. A reference POSIX installation is:

```
python -m venv .venv
.venv/bin/python -m pip install --upgrade pip
.venv/bin/python -m pip install -r requirements.txt
.venv/bin/python -m pip install -e .
```

On Windows, `.venv/Scripts/python` replaces `.venv/bin/python`. GPU retraining requires a CUDA-compatible PyTorch installation and the public model and dataset snapshots identified in the configurations. Reconstruction from archived observations does not require a GPU.

Tests are executed with:

```
.venv/bin/python -m pytest -q
```

The suite covers configuration encoding, categorical and resource constraints, QUBO/Ising conversion, decoding, deterministic routing, TianYan bit order, reliability-gate logic, and workflow consistency. Hardware tests use archived responses and do not submit QPU jobs.

G.4 Deterministic Paper Reconstruction

The principal reconstruction commands are:

```
.venv/bin/python scripts/workflows/run_final_report.py
```

```
.venv/bin/python scripts/workflows/build_iclr_submission_snapshot.py
```

The first validates completed records and rebuilds canonical summaries. The second constructs the frozen paper snapshot containing table inputs, figure data, claim records, and integrity manifests. Both terminate when a required observation is missing rather than imputing or replacing it.

Offline replay is deterministic conditional on the archived outcome pool and initial-design seeds. Paired policies receive the same initial observations but maintain isolated histories. A cached outcome is revealed to a policy only after that policy requests the corresponding configuration.

G.5 Statistical Reconstruction

The paired units and interval definitions follow Appendix C. Replay intervals resample trial-level paired differences, while full-budget intervals resample within-seed differences. Calculations use unrounded observations, although paper tables display rounded values.

The replay archive retains all initial designs and budget traces required to reconstruct simple regret, regret AUC, high-quality-region hit rate, global-best hit rate, and recovery. Structured GP-EI and RF-EI use the same feasible pools, initial designs, and budgets as feedback-refitted Ridge. GP sensitivity records additionally retain kernel parameters and fit-level boundary diagnostics.

Held-out generation metrics are read only after the corresponding selection state is frozen. BLEU, ROUGE-L, and other held-out metrics do not enter surrogate fitting, candidate promotion, or policy selection.

G.6 Protocol Freezing and Integrity

Each finalized study includes a manifest identifying its configuration, workflow, candidate pool, resource cap, seed set, query budget, decoding settings, and expected outputs. Hardware manifests additionally record the Ising objective, bit order, circuit representation, shot schedule, acceptance thresholds, and deterministic classical route.

Where applicable, SHA-256 digests cover the configuration, workflow, trainer, evaluator, candidate pool, and analysis inputs. The integrity audit verifies that every canonical summary traces to immutable seed- or trial-level records and that no held-out outcome appears in a search history before selection is frozen.

G.7 Reconstructing Ising and Real-QPU Analyses

Exact enumeration, classical retrieval, matched-shot sampling, decoding, target-based and target-free gate evaluation, and analysis of archived QPU counts can be reproduced without TianYan access. The artifact contains frozen objectives, legal-state mappings, count distributions, shot schedules, repetition identifiers, routing metadata, and gate thresholds.

The matched-shot audit retains all 1,000 paired trials, the common 1,024-sample budget, Top-6 shortlist, frozen energy gate, and random seeds for the classical controls. TianYan-176 records include the accepted-proposal audit, scale audit, and four-round sequential replay. The sequential archive contains the fixed initial design, hidden-label pool, four round-specific acquisition and Ising instances, twelve hardware responses, frequency shortlists, repeat-support and TV statistics, and QPU and classical routing decisions. TianYan-287 records include the routed stress test, three prospective 2,048-shot repetitions, proposal `p02`, target `p05`, component ablations, timing records, and downstream fallback runs.

Future hardware executions may produce different counts because of calibration drift and finite sampling. Reproducibility of the reported decisions therefore relies on archived counts and the frozen gate rules rather than identical future device samples.

G.8 ANONYMOUS-SUBMISSION CHECKLIST

Before release, the artifact is scanned for author names, affiliations, usernames, absolute paths, hostnames, repository remotes, credentials, document metadata, and identifying comments. Stored paths are converted to artifact-relative paths without changing numerical observations.

The anonymous package contains:

- source code, dependency files, and tests;
- frozen model, candidate-space, resource-budget, search, and hardware configurations;
- policy trajectories, seed-level metrics, and canonical statistical outputs;
- QUBO audits, matched-shot records, and archived QPU counts;
- deterministic table, figure, and report reconstruction workflows; and
- protocol and file-integrity manifests.

The artifact is complete when every paper-level numerical value can be reconstructed from immutable observations and traced to its configuration, seed or replay trial, selection state, and frozen protocol.