REGIMEQ: ALLOCATING SEARCH AND READOUT AFTER PARAMETER REUSE IN QAOA

Anonymous authors
Paper under double-blind review

ABSTRACT

After QAOA angles have been inherited from a related problem, the remaining sampling budget can refine parameters or read out candidate solutions. We propose RegimeQ, which uses candidate feedback to choose between further search and candidate readout. Four low-shot probes inform the decision, with their cost included in the window budget and the rule frozen after development. We evaluate RegimeQ on sequential portfolio optimization problems with 18, 20, and 22 assets, using X-mixer circuits for the main experiments and additional depth and XY-mixer checks. At 16,384 shots per window, the recent eighteen-asset X-mixer evaluation reaches 3.2718% mean gap with six calls, versus 4.4052% with five for probe-then-readout and 3.6035% with 6.48 calls for random allocation. In a separate 2018–2019 evaluation period, the unchanged rule gives the lowest mean gap among the tested policies at both 18 and 20 assets. The eighteen-asset XY evaluation also shows lower mean gap and fewer calls than random allocation. Matched-sample analysis explains how candidate retention can reverse pooled action rankings, while search frequency and duration jointly determine call cost. These comparisons identify favorable quality–call tradeoffs in several settings and show how inherited parameters, probe geometry and retained candidates affect allocation outcomes.

1 INTRODUCTION

A repeated optimization task need not begin from scratch. In monthly portfolio selection, new return and covariance estimates change the objective while the asset encoding and much of the circuit remain compatible. QAOA alternates parameterized quantum execution with classical optimization (Farhi et al., 2014). Reusing compiled structure and previous angles can remove repeated work, but it also changes the question facing the optimizer: once an inherited distribution already produces useful candidates, what should the remaining samples buy?

The distinction matters because a sampling batch has two roles. Its mean estimates the optimizer's objective; its individual bit strings are possible solutions. More accurate estimates can improve parameter decisions without producing a better feasible return. Conversely, a coarse search can visit a distribution that supplies a useful candidate even when its mean is uncompetitive. Measurement-frugal optimizers (Kübler et al., 2020; Gu et al., 2021), distribution-based allocation (Kim et al., 2025), and few-shot QAOA fine-tuning (Hao et al., 2025) address related resource questions. Here the target is the best feasible candidate returned from a sequence of related problems, including the cost of every exploratory and final batch.

Preliminary controls reveal a mismatch between sampling feedback and the returned solution. In the tested larger portfolios, a mean-precision rule keeps every optimization batch at its initial budget, and a repeated-candidate rule uses the full readout budget in every window. Fixed allocation reproduces their shot counts and optimizer trajectory. These observations lead us to use candidate quality when choosing between further search and readout.

We propose RegimeQ, which spends 2048 shots on four low-shot probes, then chooses between continued search and concentrated readout. Its threshold is frozen after development. Shared-start comparisons measure local action value; continuous runs let each policy inherit its own angles. A

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

frozen-probability random policy tests the additional value of feedback selection. Cross-scale and transfer experiments measure the resulting quality–call tradeoffs.

1. **Understanding the value of search after reuse.** Matched-sample comparisons show how candidates found during probing affect the measured benefit of further search and readout. Retaining these candidates can change the aggregate ranking of the two actions. We also separate search frequency from search duration to explain their different contributions to call cost.
2. **A candidate-feedback sampling method.** RegimeQ uses observations from four budgeted probes to choose between continued parameter search and concentrated readout within a fixed budget. Shared problems, explicit parameter feedback, and batch resource accounting support reproducible comparisons of its allocation decisions.
3. **Evaluation against fixed and random allocation.** Shared-start comparisons and continuous trajectories test the frozen rule against both fixed actions and an executed random policy. Separate-period, depth and circuit checks identify favorable quality–call tradeoffs and failures under the same rule.

2 Background and Attribution Questions

2.1 Rolling portfolio selection

At month t, a binary vector selects B of n assets. With estimated returns μ_t, covariance Σ_t, and supplied holdings h_{t-1}, the objective is

$$f_t(x) = \lambda x^\top \Sigma_t x - \mu_t^\top x + \kappa \|x - h_{t-1}\|_1, \qquad x \in \{0,1\}^n, \quad \mathbf{1}^\top x = B. \tag{1}$$

This binary workload adds cardinality and turnover to mean–variance selection (Markowitz, 1952). Adding $P(\mathbf{1}^\top x - B)^2$ yields a QUBO, mapped to an Ising operator by $x_i = (1 - Z_i)/2$ (Lucas, 2014). In a holdings-feedback study, h_{t-1} is the preceding returned portfolio. In the main common-QUBO studies it is the same fixed supplied portfolio for every method. These experiments evolve parameters, not trading positions.

2.2 Templates, angles, and candidates

The cost operator is $H_t = c_t + \sum_i a_{i,t} Z_i + \sum_{i<j} J_{ij,t} Z_i Z_j$. Our reference circuit uses $p = 3$, a uniform initial state, an X mixer, and unscaled physical coefficients; Section 4 specifies the extensions. Six angles $\theta = (\gamma_1, \gamma_2, \gamma_3, \beta_1, \beta_2, \beta_3)$ define a sampling distribution $p_t(x \mid \theta)$. A shot draws one bit string from that distribution; it does not enumerate the feasible portfolios. At eighteen assets, eighteen logical qubits encode a choice of nine.

A *template* fixes compatible gate structure. The penalty couples every asset pair, so coefficients often change without changing connectivity. An inherited *angle vector* initializes a new solve, but its gates still change through $RZZ(2\gamma_\ell J_{ij,t})$. A retained *candidate* is a classical observation scored for the current QUBO. We reuse templates and angles across compatible windows; we clear candidate records between problems. Partial compilation and parameter binding provide the first capability (Gokhale et al., 2019; IBM Quantum, [n.d.]); parameter transfer provides precedents for the second (Galda et al., 2023; Shaydulin et al., 2023).

This separation determines the controls. Fresh versus Bind measures the effect of basic reuse; it does not isolate a sampling decision. Shared starts and actual budgets isolate local actions. Own-parameter trajectories test repeated use. Finally, holding samples fixed and changing only return eligibility isolates candidate retention.

2.3 Why search and readout can exchange roles

For intuition, fix a feasible target set $\mathcal{T}$, a predetermined sequence of parameter vectors, and independent samples. If batch k has N_k shots and target probability $p_k = \Pr_{x \sim p_t(\cdot \mid \theta_k)}[x \in \mathcal{T}]$, then

$$\Pr(\text{at least one target observation}) = 1 - \prod_k (1 - p_k)^{N_k}. \tag{2}$$

Readout increases N_k at one distribution; search tries to find a distribution with a larger p_k, spending samples and calls along the way. A lower mean energy does not order these target probabilities. Equation (2) is a fixed-schedule sampling identity, not an online estimator or a guarantee for an adaptive trajectory. It explains why candidate return, the distributions visited during search, and final readout must all enter the comparison.

3 REGIMEQ EXECUTION FRAMEWORK

Motivation. Preliminary 18/20-asset controls keep all 577 mean-precision checks at 512 shots and all 69 readouts at 2048; fixed allocation reproduces the trajectory with fewer calls. On identical ETF-10 samples, retaining all stages lowers the earlier Adaptive return gap from 0.7885% to 0.0850%. These controls motivate candidate feedback and all-stage returns (separate Supplement S1/S9; Appendices A–G are included here).

RegimeQ allocates a fixed shot budget after binding the current QUBO to a compatible circuit and inheriting starting angles. Figure 1 shows the candidate-feedback workflow: four paid probes inform a choice between continued search and concentrated readout, all stages contribute feasible candidates, and the selected readout parameters initialize the next window. The candidate-feedback experiments disable detector resets to isolate allocation after inheritance. Preliminary sampling controls are reported separately in the supplement.

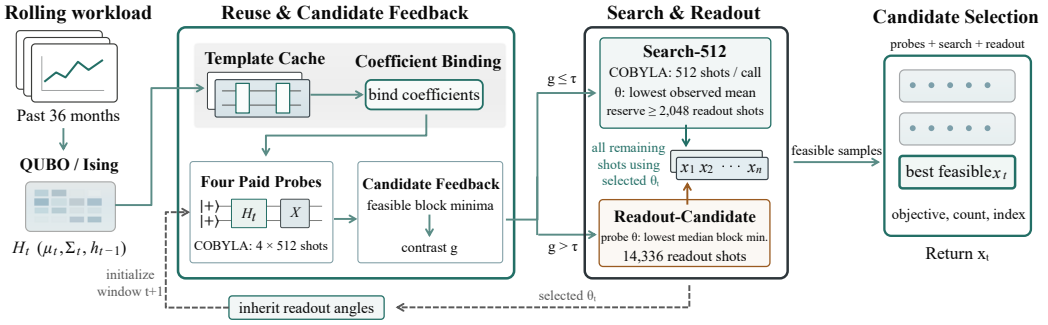


Figure 1: **RegimeQ candidate-feedback workflow.** After binding current QUBO coefficients to a compatible circuit template, RegimeQ runs four 512-shot probes. A frozen rule uses the candidate contrast g to select Search-512 or Readout-Candidate. Final readout uses the selected parameters and all remaining shots within a 16,384-shot window budget. Feasible samples from the probes, subsequent search when selected, and final readout enter one candidate record, which returns the best feasible observation. The dashed loop carries the selected readout parameters forward to initialize the next window. Table 1 specifies the detailed execution rules and boundary cases.

3.1 FOUR PROBES AND TWO ALLOCATION ACTIONS

Four COBYLA evaluations at 512 shots each form a common prefix. Every sample both estimates an objective and updates the candidate record. The policy then commits for the rest of the window: **Search-512** (L) continues mean-based optimization at 512 shots per evaluation; **Readout-Candidate** (R) selects one of the four probed vectors and spends the remaining $14,336$ shots there in one call. Search-512 reserves at least 2048 shots and assigns its entire unused budget to one final readout. Thus both actions spend exactly $16,384$ shots, but differ in calls and visited distributions. Readout-Candidate includes four probes and is not direct readout of an untouched inherited vector.

3.2 CANDIDATE FEEDBACK AND THE FROZEN DECISION

Each probe is summarized by its best feasible candidates in fixed-size sample blocks. We compare these block minima between the first two and last two probes. For normalization, remove the cardinality penalty and let c_t^b and $S_t^b = \sum_i |a_{i,t}^b| + \sum_{i<j} |J_{ij,t}^b|$ be the business operator's

offset and coefficient radius. When $2S_t^b > 10^{-12}$, a feasible observation has score $u_t(x) = [f_t(x) - (c_t^b - S_t^b)]/(2S_t^b)$, computed without an optimal solution. Divide each randomly ordered 512-sample probe into four blocks of 128. Let b_{ij} be the minimum feasible u_t in block j of probe i; a block without feasible samples scores one. Define

$$e_j = \min(b_{1j}, b_{2j}), \quad l_j = \min(b_{3j}, b_{4j}), \quad g = \min\left\{\frac{e_1 - l_1 + e_2 - l_2}{2}, \frac{e_3 - l_3 + e_4 - l_4}{2}\right\}. \tag{3}$$

The frozen rule chooses Search-512 when $g \leq \tau$, with $\tau \approx -0.02348$, and Readout-Candidate otherwise. Execution uses the full-precision threshold in Appendix B. At least one half-contrast must cross this negative threshold to select search; the two halves need not agree. The four-block, two-half construction is a fixed empirical feature, not a confidence test. Development fitted its direction and threshold, without comparing feature forms.

A negative contrast indicates worse block minima under later perturbations. Continuing then explores further directions rather than committing to a probe. The fitted rule treats this as a reason to search; matched-start and random-allocation controls test whether that choice improves the returned candidate.

When $2S_t^b \leq 10^{-12}$, blocks with feasible samples score zero and empty feasible blocks score one. Readout-Candidate ranks parameters by $\mathrm{median}_j\, b_{ij}$, then sampled normalized mean and probe index, including the zero-scale case. Search-512 selects the smallest observed search mean. The action changes allocation *and* parameter selection. Readout-Mean keeps Readout-Candidate's allocation but selects by prefix mean. Random-Mix uses the same paid probes and L/R actions as RegimeQ, but its allocation decision is a frozen random draw; readout parameter selection remains candidate-based.

Table 1: Window execution of the frozen candidate-feedback policy RegimeQ. The candidate record and optimizer state restart each window; only the selected angles and compatible template persist.

1	Bind current coefficients; inherit θ or draw initial angles; clear candidate record.
2	Run four 512-shot probes; charge once and retain all feasible observations.
3	If no feasible prefix: choose Probe-Search-2048 (2048-shot search); else if $2S_t^b \leq 10^{-12}$: choose Readout-Candidate; else choose Search-512 iff $g \leq \tau$, otherwise Readout-Candidate.
4	For Search-512/Probe-Search-2048 continue search with a 2048-shot readout reserve. For Readout-Candidate select by the block score.
5	Read out at selected θ with all remaining shots; return the best feasible observation; carry this θ to the next window.

3.3 Stopping, returns, and resource priority

COBYLA (Powell, 1994) uses `rhobeg` $= 0.5$, at most 250 evaluations, and `catol` $= 10^{-8}$. Every strictly lower observed mean updates the selected parameter vector. Separately, a stopping reference updates only after a coefficient-bound-normalized improvement exceeding 0.002; six stale evaluations permit stopping after at least eight evaluations. The hard shot cap and readout reserve take priority over this minimum. For Search-2048 (F), the budget allows only seven 2048-shot evaluations and one 2048-shot readout call. This budget-limited control differs from the earlier uncapped Fixed policy.

All stages retain feasible candidates, ranked by current business objective, cumulative count, and bit-string index. If no feasible sample is observed, the return is explicitly marked infeasible. First-window angles are independent $\gamma_\ell \sim U[0, 2\pi)$ and $\beta_\ell \sim U[0, \pi)$. Later windows inherit the selected readout vector, not the best candidate's generating vector or the optimizer's internal state. Deadline interruptions preserve completed batches and are excluded from complete-pair summaries.

4 Evaluation Methodology

We use nested FF48 industry subsets (French, 2026), with $B = n/2$, $\lambda = 0.5$, $\kappa = 0.001$, and $P = 0.1$. Each QUBO uses the preceding 36 monthly observations, excluding the decision month,

and supplied h_{t-1} selecting the first B assets. All methods share that problem. Appendix A records ordering and implementation.

Comparison protocols. Common-start experiments fix the QUBO and initial angles from a separately charged Bind-512 trajectory. Three prefix seeds yield 84 contexts: twelve months at 18/20 assets and four at 22 assets. The low-shot search, candidate readout, mean-selected readout and high-shot fallback share four probes, with two future sampling repetitions per action. Search-2048 starts independently from the same angles. In continuous experiments, policies share first-month angles and subsequently inherit their own selected vectors. Each trajectory has twelve warmup and twelve evaluation months; every window spends $16,384$ shots with RESET disabled.

Random-Mix chooses Search-512 with probability 0.25, the frozen RegimeQ search frequency in 72 development contexts, and Readout-Candidate otherwise. The decision stream is independent of sampling; boundary fallbacks match RegimeQ. Each mixed trajectory runs its decisions and inherits its own angles. It is not reconstructed from isolated action outcomes.

Evaluation scope. The original recent-period study uses X circuits at $p = 3$, three initializations per size and July 2024–June 2026 trajectories at 18/20/22 assets. Random-Mix has three nested switching streams per initialization. The separate-period check runs July 2017–June 2019 at 18/20 assets. A new recent-period batch tests raw X at $p = 2, 3, 4$, normalized X and XY at $p = 3$ (18/20 assets), and raw X at $p = 3$ on ETF-10, using price data from Tiingo (Tiingo, [n.d.]). Each setting has three new initializations; temporal and circuit checks use one switching stream per initialization. Normalization divides nonidentity cost coefficients by their Euclidean norm. XY uses a gate-prepared Dicke state, a ring mixer and no cardinality penalty. The rule and random probability remain unchanged. Appendices A–G give exact seeds, circuit details and completion records.

Quality and replication. For feasible return $\hat{x}_t$, offline feasible extrema $f_t^*, f_t^{\max}$ define

$$G_t = 100\frac{f_t(\hat{x}_t) - f_t^*}{\max(|f_t^*|, f_t^{\max} - f_t^*, 10^{-12})}, \quad G_t^{\text{range}} = 100\frac{f_t(\hat{x}_t) - f_t^*}{f_t^{\max} - f_t^*}, \quad A_t^{\text{ret}} = 1 - G_t^{\text{range}}/100. \tag{4}$$

The gaps coincide on the primary records. A^{ret} uses the feasible-range convention (Brandhofer et al., 2023; Hao et al., 2025), applied to the returned candidate. Constant feasible objectives give zero gap; infeasible returns miss all thresholds. Means weight initializations equally; switching repetitions are averaged within initialization. The p90 is a pooled, linearly interpolated quantile of observed returns. We report $G \leq 0$, 0.1%, 0.5%, and 1%, all three paired initialization differences, and switching variability. Overlapping months and nested repetitions are not independent market instances; three initializations support descriptive comparisons, not a reliable significance claim.

Resources. All probes, optimization and readout batches are charged; warmup, development and replay are separate. Ideal double-precision Aer CPU statevectors use two workers, one numerical thread and a 4 GiB limit each, on an i7-12700H with about 16 GiB RAM. The frozen stack is Python 3.11.3, Qiskit 2.5.2, Aer 0.17.2 and SciPy 1.17.1. Exact references are offline only. Simulated seconds, shots and calls are reported separately.

5 RESULTS

5.1 WHAT IS THE VALUE OF SEARCH AND READOUT FROM THE SAME START?

Table 2 separates parameter selection from action selection by fixing the QUBO, starting angles and shot budget.

Table 2: Common-start mean G (%) at 16,384 shots, three seeds and two future repetitions. Eighteen/twenty assets cover twelve months, twenty-two four. Readout-Mean and Readout-Candidate both use five calls; full resources are in separate Supplement S10, Table S25.

n	Search 2048	Search 512	Readout Candidate	Readout Mean	RegimeQ	RegimeQ calls
18	3.7474	3.2053	3.0891	3.0996	**3.0001**	6.83
20	5.7850	5.6662	**5.0199**	5.3810	5.1633	6.33
22	4.9474	5.3601	**4.6485**	5.1414	4.6661	6.00

Candidate selection. Readout-Candidate beats Readout-Mean at all three sizes under identical probe/readout budgets. Selecting the readout vector by block minima therefore helps even without changing the allocation. **Action selection.** At 18 assets, RegimeQ's 3.0001% mean gap is below both Search-512 and Readout-Candidate. At 20/22 assets, fixed candidate readout instead gives the lowest mean gap. The advantage of candidate-based parameter selection is thus distinct from the value of switching between search and readout. Bold values mark the lowest displayed quality metric within a matched comparison, including ties, rather than significance.

5.2 How much do quality gains cost in continuous use?

Figure 2 compares policies that inherit their own parameters at equal shots. A point farther down and left has both lower mean gap and fewer calls.

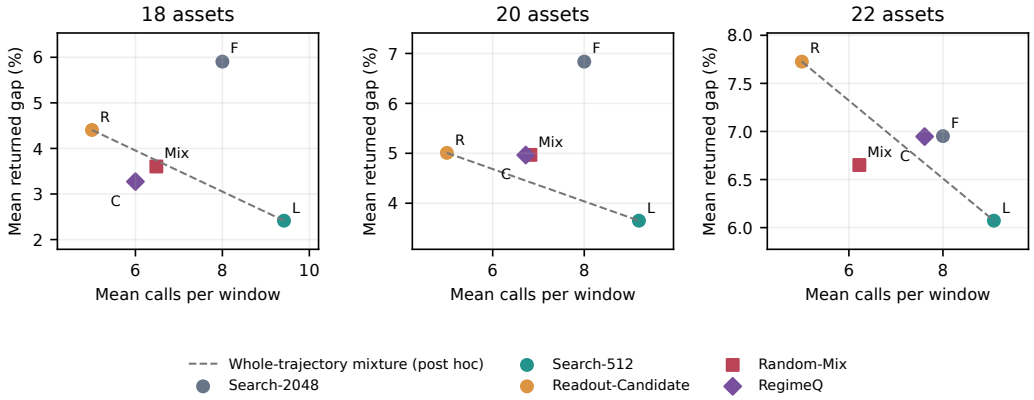


Figure 2: Recent continuous evaluation: July 2025–June 2026, three initializations, 16,384 shots per window. F: Search-2048; L: Search-512; R: Readout-Candidate; C: RegimeQ; Mix: executed Random-Mix with three nested switching streams. The dashed L/R line is a retrospective whole-trajectory mixture, not window-level random switching. Complete values and p90 appear in Appendix Table 6.

Fixed actions. Search-512 gives the lowest mean gap at each size, while Readout-Candidate uses five calls. At 18 assets, RegimeQ reduces readout's mean gap by 1.1333 percentage points for one additional call; p90 also improves. The twenty-asset mean gain is much smaller and p90 worsens. Quality and call ranks alone therefore do not establish a useful tradeoff.

Feedback versus random allocation. At 18 assets, RegimeQ lowers Random-Mix's mean gap from 3.6035% to 3.2718% and calls from 6.48 to 6.00. The two policies nearly tie at 20 assets; Random-Mix has lower mean gap, p90 and calls at 22. Mean gap improves in two of three paired initializations at 18 assets and one at each larger size (Appendix Table 9). At 22 assets, actual Random-Mix also falls below the retrospective L/R line: switching within a parameter trajectory cannot be inferred from fixed endpoints.

5.3 WHY DO THE ALLOCATION OUTCOMES DIFFER?

Table 3 connects call cost to the executed search branches and tests local action value from shared starts. These are different comparisons: the first follows each policy's trajectory; the second holds its starting parameters fixed.

Table 3: RegimeQ call sources (left) and conditional action value (right), at 16,384 shots. Left: 36 recent evaluation windows per size; $K \mid L$ includes all calls when search is chosen. Right: mean search-minus-readout gap in percentage points, grouped by the frozen choice, with context counts in parentheses; two future streams per context. Negative differences favor search.

	Continuous calls			Shared-start $G_L - G_R$	
n	Search (%)	$K \mid L$	Mean K	Search group	Readout group
18	19.44	10.14	6.00	-0.2003 (16)	0.3693 (20)
20	41.67	9.13	6.72	0.4302 (12)	0.7543 (24)
22	38.89	11.71	7.61	0.0704 (3)	0.9253 (9)

Search frequency and length. For ordinary L/R windows, $\mathbb{E}[K] = 5 + p_L(\mathbb{E}[K \mid L] - 5)$. From 20 to 22 assets, search frequency falls from 41.67% to 38.89%, but conditional calls rise from 9.13 to 11.71. Longer searches outweigh the lower frequency, increasing mean calls.

Selected contexts. Shared starts separate feedback selection from later trajectory differences. At 18 assets, search wins on average in its assigned group and loses in the readout group. At 20/22 assets, both groups favor readout, although search is less unfavorable in its assigned group. The feature separates relative action value without consistently identifying a group where extra search pays off.

Candidate retention. For a shared prefix best gap G_P, the all-stage return is $\min(G_P, G_{\text{continuation}})$. This preserves each paired context's action ordering or creates a tie, but attenuates gains and losses unequally across contexts. In the eighteen-asset search group, the mean search-minus-readout gap changes from 0.1322 points without prefix candidates to -0.2003 with them. Better pooled returns can therefore coexist with weaker continuation samples. Appendix D gives the full decomposition and Equation 5.

5.4 WHERE DOES THE FROZEN RULE TRANSFER?

Table 4 keeps the rule and random probability fixed while changing the decision period, depth, circuit or dataset. Each policy inherits its own parameters; the initialization batches remain separate.

Period. In July 2018–June 2019 (A), RegimeQ has the lowest evaluation mean gap at both sizes. At 20 assets it also uses fewer evaluation calls than Random-Mix, with gap improvements in all three initializations. Counting warmup reverses the call comparison: 193.67 versus 186.00 per trajectory. At 18 assets, the quality gain instead costs more calls.

Depth and circuit. RegimeQ gives the lowest mean gap at 18 assets with raw $p = 2$ and scaled X, and at 20 assets with raw $p = 4$ (B), using more calls than readout. At 18-asset raw $p = 4$, readout instead improves both metrics. The XY package lowers every policy's mean gap relative to scaled X. At 18 assets, RegimeQ beats Random-Mix in gap and calls in all three initializations, while Search-512 buys a slightly lower gap with about twice the calls. At 20-asset XY, RegimeQ chooses search only once in 36 evaluation windows and stays close to fixed readout.

Dataset and saturation. ETF-10 (C) has only 252 feasible portfolios. Every policy reaches zero gap in all 36 evaluation windows; RegimeQ finds an optimum within its probes in 33. Extra allocation decisions then have no measured quality return, and fixed readout uses the fewest calls.

5.5 HOW DO BUDGET AND PROBE ORDER CHANGE THE OUTCOME?

Probe coverage. With six angles, Search-2048 spends its seven evaluations on the initial simplex, whereas Search-512 can continue beyond it. The four $p = 3$ probes cover only cost angles. Table 5 changes budget and the first perturbed coordinates at fixed starting parameters.

Table 4: Continuous transfer at 16,384 shots per window: mean G (%) and calls. Each cell has three initializations and 36 evaluation returns after twelve warmup months. A uses an independent decision period; B/C use new recent-period initializations. Raw/scaled denote physical coefficient scales. Bold marks the lowest mean gap per row, including ties. Appendices E/G give p90, paired seeds and full-trajectory costs.

Setting	n	Search-512		Readout-Candidate		Random-Mix		**RegimeQ**	
		G	Calls	G	Calls	G	Calls	G	Calls
A. Temporal transfer: July 2018–June 2019									
X, raw, $p = 3$	18	5.6826	9.83	3.2268	5.00	2.8468	5.97	**2.5790**	7.44
X, raw, $p = 3$	20	3.5730	9.06	3.7688	5.00	4.4400	7.83	**3.3619**	7.58
B. Depth and circuit: July 2025–June 2026, new initializations									
X, raw, $p = 2$	18	4.8399	9.06	4.8364	5.00	2.9978	6.53	**2.4669**	7.33
X, raw, $p = 2$	20	**3.9681**	9.17	4.1836	5.00	5.2050	6.39	4.9375	7.64
X, raw, $p = 3$	18	4.2894	9.92	5.4336	5.00	**3.0062**	6.75	3.3171	7.61
X, raw, $p = 3$	20	**4.9438**	9.28	5.2471	5.00	6.3721	5.69	5.0429	9.19
X, raw, $p = 4$	18	**3.2467**	9.08	3.7504	5.00	3.5354	7.06	4.3067	7.39
X, raw, $p = 4$	20	5.4089	9.58	6.8751	5.00	6.3328	5.83	**5.1732**	7.61
X, scaled, $p = 3$	18	4.3218	9.56	4.6617	5.00	4.2274	6.42	**3.6798**	8.47
X, scaled, $p = 3$	20	5.1973	9.72	**4.3054**	5.00	5.5995	6.06	5.9359	7.19
XY, scaled, $p = 3$	18	**0.8980**	12.81	1.4824	5.00	1.4761	6.97	1.0049	6.42
XY, scaled, $p = 3$	20	**1.0325**	14.36	2.3554	5.00	1.5156	6.33	2.3277	5.17
C. ETF-10: July 2025–June 2026, new initializations									
ETF-10, X, $p = 3$	10	**0.0000**	9.03	**0.0000**	5.00	**0.0000**	6.42	**0.0000**	7.28

Table 5: Shared-start sensitivity: RegimeQ minus Readout-Candidate, written as mean gap difference (percentage points) / mean call difference. Each cell averages twelve seed–month contexts; readout uses five calls. Negative gap differences favor RegimeQ. All twelve budget–order settings are shown; Appendix Table 17 also compares Search-512.

n	Probe order	8192 shots	16,384 shots	32,768 shots
18	Cost	0.9707 / 2.00	0.8076 / 2.00	0.2502 / 2.00
18	Mixer	-0.1263 / 3.33	0.2235 / 3.33	0.1056 / 3.33
20	Cost	0.2944 / 1.00	-0.0580 / 1.00	0.2200 / 1.00
20	Mixer	-1.6405 / 2.67	-1.2879 / 2.67	-0.4013 / 2.67

Budget. Readout improves both metrics in 7 settings; in the other 5, RegimeQ lowers gap at more calls. The twenty-asset cost-first comparison changes from a loss at 8192 shots to a gain at larger budgets, despite identical prefixes and choices. The signal omits budget even though continuation value depends on it.

Coordinate order. Moving the first perturbations to mixer angles raises twenty-asset search assignment from 25.00% to 66.67%. The feedback therefore responds to probe geometry as well as candidate observations. These local checks identify that dependence; altered budgets and orders have not yet been tested in continuous trajectories.

6 Discussion

Tradeoffs in the evaluated settings. In the original recent-period X runs, Search-512 gives the lowest mean gaps and Readout-Candidate uses the fewest calls. RegimeQ improves the eighteen-asset readout gap by 1.1333 points for one additional call per window. At eighteen-asset XY, it lowers both gap and calls relative to Random-Mix in every initialization. At twenty-two-asset raw X, the measured comparison instead favors Random-Mix on both metrics. ETF-10 reaches zero gap

under all policies, so fixed readout's five calls suffice for the attained quality. These observations quantify choices in tested settings; they do not supply a reliable advance rule for selecting a policy on a new workload.

What the feedback measures. The feature combines finite-sample tail observations with the optimizer's early perturbations. Their usefulness depends on the inherited angles and the candidates already retained. Changing coordinate order alters the signal, while changing the remaining budget alters continuation value without changing the signal. The conditional controls show relative separation between the groups, but the search-assigned group still favors readout at 20/22 assets. Including remaining budget or broader probe coverage would require a new rule and independent evaluation.

Scope. The experiments span periods, depths two to four, X/XY circuits and ETF-10, but share one portfolio formulation, COBYLA and ideal simulation. FF48 subsets are nested, monthly estimates overlap, and each setting has three initialization trajectories. Other optimizers, probe lengths and continuous trajectories under changed budgets/orders remain untested. Calls measure submissions rather than hardware latency. Separate Supplement S9, S4 and S2 document earlier sampling, classical and circuit controls.

7 Related Work

Parameter choice and reuse. QAOA parameter strategies exploit depth structure (Zhou et al., 2020), annealing-inspired initialization (Sack & Serbyn, 2021), multistart search (Shaydulin et al., 2019), or clustering (Moussa et al., 2022). Typical-instance concentration (Brandao et al., 2018) and weighted-problem transfer and rescaling (Shaydulin et al., 2023; Sureshbabu et al., 2024) explain why useful initial parameters can exist before local optimization. Similarity-based transfer studies graph properties (Galda et al., 2023); warm-start QAOA instead changes the prepared quantum state using classical relaxations (Egger et al., 2021). Our policy starts after angle inheritance and decides how to use a remaining finite budget. It does not supply a new transfer theorem or state-preparation method.

Optimization under sampling noise. Hybrid optimization must balance noisy estimates and classical updates (Guerreschi & Smelyanskiy, 2017; Sung et al., 2020; Sweke et al., 2020). iCANS and gCANS adapt gradient-estimation effort (Kübler et al., 2020; Gu et al., 2021); operator sampling changes which Hamiltonian terms are measured (Arrasmith et al., 2020); latency-aware allocation also accounts for submission cost (Ito, 2023). DDS uses distribution feedback (Kim et al., 2025), and few-shot QAOA combines parameter initialization and economical fine-tuning (Hao et al., 2025). CVaR changes the optimization aggregate toward a low-energy tail (Barkoutsos et al., 2020). RegimeQ retains mean-based search but uses candidate observations to select a joint continuation/readout action. We compare against the same fixed actions because economical sampling alone does not establish the value of choosing between them.

Compilation and constrained circuits. Partial compilation (Gokhale et al., 2019), Pauli-block optimization (Li et al., 2022), and VQA compilation reuse (Jin et al., 2024) address circuit costs. Constraint-preserving alternating operators (Hadfield et al., 2019; Xiang et al., 2025), portfolio mixer benchmarks (Brandhofer et al., 2023), and initial-state/mixer alignment (He et al., 2023) change the quantum sampling family itself. These are complementary design axes. Within each comparison, policies share the same circuit, separating allocation effects from the X/XY circuit change. Sampling-based performance bounds (Lotshaw et al., 2021; Lykov et al., 2023) further motivate reporting candidate counts and the returned solution separately from an expectation value.

8 Conclusion

RegimeQ uses four budgeted probes to choose between further QAOA search and candidate readout. The frozen rule achieves favorable quality–call tradeoffs in several evaluated settings, including lower mean gap and fewer calls than Random-Mix in all three eighteen-asset XY initializations. Candidate retention changes pooled action rankings, while search frequency and conditional duration jointly determine call cost. The results connect allocation outcomes to inherited parameters, probe geometry and candidates already observed.

STATEMENT ON THE USE OF AI

Generative AI tools were used to assist with language polishing,wording refinement and consistency checks. There search design, algorithm development, experiments, analysis, and conclusions were carried out and verified by the authors. All AI-assisted edits were reviewed by the authors, who remainfully responsible for the manuscript.

REFERENCES

Andrew Arrasmith, Lukasz Cincio, Rolando D. Somma, and Patrick J. Coles. Operator sampling for shot-frugal optimization in variational algorithms. arXiv preprint, 2020. URL https://arxiv.org/abs/2004.06252.

Panagiotis Kl. Barkoutsos, Giacomo Nannicini, Anton Robert, Ivano Tavernelli, and Stefan Woerner. Improving variational quantum optimization using CVaR. *Quantum*, 4:256, 2020. doi: 10.22331/q-2020-04-20-256. URL https://quantum-journal.org/papers/q-2020-04-20-256/.

Andreas Bärtschi and Stephan Eidenbenz. Deterministic preparation of Dicke states. In *Fundamentals of Computation Theory*, pp. 126–139. Springer, 2019. doi: 10.1007/978-3-030-25027-0_9. URL https://arxiv.org/abs/1904.07358.

Dorian Baudry, Yoan Russac, and Emilie Kaufmann. Efficient algorithms for extreme bandits. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 2210–2248. PMLR, 2022. URL https://proceedings.mlr.press/v151/baudry22a.html.

Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermanni Heimonen, Jakob S. Kottmann, Tim Menke, Wai-Keong Mok, Sukin Sim, Leong-Chuan Kwek, and Alán Aspuru-Guzik. Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1):015004, 2022. doi: 10.1103/revmodphys.94.015004. URL https://doi.org/10.1103/revmodphys.94.015004.

Fernando G. S. L. Brandao, Michael Broughton, Edward Farhi, Sam Gutmann, and Hartmut Neven. For fixed control parameters the quantum approximate optimization algorithm's objective function value concentrates for typical instances. arXiv preprint, 2018. URL https://arxiv.org/abs/1812.04170.

Sebastian Brandhofer, Daniel Braun, Vanessa Dehn, Gerhard Hellstern, Matthias Hüls, Yanjun Ji, Ilia Polian, Amandeep Singh Bhatia, and Thomas Wellens. Benchmarking the performance of portfolio optimization with QAOA. *Quantum Information Processing*, 22:25, 2023. doi: 10.1007/s11128-022-03766-5. URL https://link.springer.com/article/10.1007/s11128-022-03766-5.

Alejandro Gomez Cadavid, Ananth Kaushik, Pranav Chandarana, Miguel Angel Lopez-Ruiz, Gaurav Dev, Willie Aboumrad, Qi Zhang, Claudio Girotto, Sebastián V. Romero, Martin Roetteler, Enrique Solano, Marco Pistoia, and Narendra N. Hegade. Large-scale portfolio optimization on a trapped-ion quantum computer. arXiv preprint, 2026. URL https://arxiv.org/abs/2602.23976.

M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021. doi: 10.1038/s42254-021-00348-9. URL https://doi.org/10.1038/s42254-021-00348-9.

Daniel J. Egger, Jakub Marecek, and Stefan Woerner. Warm-starting quantum optimization. *Quantum*, 5:479, 2021. doi: 10.22331/q-2021-06-17-479. URL https://arxiv.org/abs/2009.10095.

Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. arXiv preprint, 2014. URL https://arxiv.org/abs/1411.4028.

Kenneth R. French. Data library: 48 industry portfolios, 2026. URL https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/Data_Library/det_48_ind_port.html. Frozen monthly value-weighted return series ending June 2026; accessed September 19, 2026.

Alexey Galda, Eesh Gupta, Jose Falla, Xiaoyuan Liu, Danylo Lykov, Yuri Alexeev, and Ilya Safro. Similarity-based parameter transferability in the quantum approximate optimization algorithm. *Frontiers in Quantum Science and Technology*, 2:1200975, 2023. doi: 10.3389/frqst.2023.1200975. URL https://arxiv.org/abs/2307.05420.

Pranav Gokhale, Yongshan Ding, Thomas Propson, Christopher Winkler, Nelson Leung, Yunong Shi, David I. Schuster, Henry Hoffmann, and Frederic T. Chong. Partial compilation of variational algorithms for noisy intermediate-scale quantum machines. In *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 266–278, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3352460.3358313. URL https://doi.org/10.1145/3352460.3358313.

Jérôme F. Gonthier, Maxwell D. Radin, Corneliu Buda, Eric J. Doskocil, Clena M. Abuan, and Jhonathan Romero. Measurements as a roadblock to near-term practical quantum advantage in chemistry: Resource analysis. *Physical Review Research*, 4(3):033154, 2022. doi: 10.1103/physrevresearch.4.033154. URL https://doi.org/10.1103/physrevresearch.4.033154.

Andi Gu, Angus Lowe, Pavel A. Dub, Patrick J. Coles, and Andrew Arrasmith. Adaptive shot allocation for fast convergence in variational quantum algorithms. arXiv preprint, 2021. URL https://arxiv.org/abs/2108.10434.

Gian Giacomo Guerreschi and Mikhail Smelyanskiy. Practical optimization for hybrid quantum-classical algorithms. arXiv preprint, 2017. URL https://arxiv.org/abs/1701.01450.

Stuart Hadfield, Zhihui Wang, Bryan O'Gorman, Eleanor G. Rieffel, Davide Venturelli, and Rupak Biswas. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34, 2019. doi: 10.3390/a12020034. URL https://doi.org/10.3390/a12020034.

Tianyi Hao, Zichang He, Ruslan Shaydulin, Jeffrey Larson, and Marco Pistoia. End-to-end protocol for high-quality quantum approximate optimization algorithm parameters with few shots. *Physical Review Research*, 7(3):033179, 2025. doi: 10.1103/24gg-7p8z. URL https://doi.org/10.1103/24gg-7p8z.

Zichang He, Ruslan Shaydulin, Shouvanik Chakrabarti, Dylan Herman, Changhao Li, Yue Sun, and Marco Pistoia. Alignment between initial state and mixer improves QAOA performance for constrained optimization. *npj Quantum Information*, 9:121, 2023. doi: 10.1038/s41534-023-00787-5. URL https://www.nature.com/articles/s41534-023-00787-5.

IBM Quantum. QuantumCircuit: parameter assignment. Qiskit API documentation, version 1.3, [n.d.]. URL https://quantum.cloud.ibm.com/docs/en/api/qiskit/1.3/qiskit.circuit.QuantumCircuit#assign_parameters. Accessed September 10, 2026; cited for the parameter-binding capability, not the experiment's package version.

Kosuke Ito. Latency-aware adaptive shot allocation for run-time efficient variational quantum algorithms. arXiv preprint, 2023. URL https://arxiv.org/abs/2302.04422.

Naman Jain and M. Girish Chandra. Exploring the synergistic potential of quantum annealing and gate model computing for portfolio optimization. arXiv preprint, 2023. URL https://arxiv.org/abs/2305.01480.

Yuwei Jin, Zirui Li, Fei Hua, Tianyi Hao, Huiyang Zhou, Yipeng Huang, and Eddy Z. Zhang. Tetris: A compilation framework for VQA applications in quantum computing. In *Proceedings of the 51st Annual International Symposium on Computer Architecture*, pp. 277–292, United States, 2024. IEEE. doi: 10.1109/ISCA59077.2024.00029. URL https://arxiv.org/abs/2309.01905.

Youngmin Kim, Enhyeok Jang, Hyungseok Kim, Seungwoo Choi, Changheon Lee, Donghwi Kim, Woomin Kyoung, Kyujin Shin, and Won Woo Ro. Distribution-adaptive dynamic shot optimization for variational quantum algorithms. *Physical Review Research*, 7(4):043253, 2025. doi: 10.1103/vwhk-22b8. URL https://doi.org/10.1103/vwhk-22b8.

Jonas M. Kübler, Andrew Arrasmith, Lukasz Cincio, and Patrick J. Coles. An adaptive optimizer for measurement-frugal variational algorithms. *Quantum*, 4:263, 2020. doi: 10.22331/q-2020-05-11-263. URL https://quantum-journal.org/papers/q-2020-05-11-263/.

Gushu Li, Yufei Ding, and Yuan Xie. Tackling the qubit mapping problem for NISQ-era quantum devices. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 1001–1014, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3297858.3304023. URL https://doi.org/10.1145/3297858.3304023.

Gushu Li, Anbang Wu, Yunong Shi, Ali Javadi-Abhari, Yufei Ding, and Yuan Xie. Paulihedral: A generalized block-wise compiler optimization framework for quantum simulation kernels. In *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 554–569, New York, NY, USA, 2022. Association for Computing Machinery. doi: 10.1145/3503222.3507715. URL https://doi.org/10.1145/3503222.3507715.

Phillip C. Lotshaw, Travis S. Humble, Rebekah Herrman, James Ostrowski, and George Siopsis. Empirical performance bounds for quantum approximate optimization. *Quantum Information Processing*, 20(12):403, 2021. doi: 10.1007/s11128-021-03342-3. URL https://doi.org/10.1007/s11128-021-03342-3.

Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2:5, 2014. doi: 10.3389/fphy.2014.00005. URL https://doi.org/10.3389/fphy.2014.00005.

Danylo Lykov, Jonathan Wurtz, Cody Poole, Mark Saffman, Tom Noel, and Yuri Alexeev. Sampling frequency thresholds for the quantum advantage of the quantum approximate optimization algorithm. *npj Quantum Information*, 9:73, 2023. doi: 10.1038/s41534-023-00718-4. URL https://www.nature.com/articles/s41534-023-00718-4.

Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. doi: 10.1111/j.1540-6261.1952.tb01525.x. URL https://doi.org/10.1111/j.1540-6261.1952.tb01525.x.

Charles Moussa, Hao Wang, Thomas Bäck, and Vedran Dunjko. Unsupervised strategies for identifying optimal parameters in quantum approximate optimization algorithm. *EPJ Quantum Technology*, 9:11, 2022. doi: 10.1140/epjqt/s40507-022-00131-4. URL https://doi.org/10.1140/epjqt/s40507-022-00131-4.

Prakash Murali, Jonathan M. Baker, Ali Javadi-Abhari, Frederic T. Chong, and Margaret Martonosi. Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 1015–1029, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3297858.3304075. URL https://doi.org/10.1145/3297858.3304075.

Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1):4213, 2014. doi: 10.1038/ncomms5213. URL https://doi.org/10.1038/ncomms5213.

Michael James David Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In Susana Gomez and Jean-Pierre Hennart (eds.), *Advances in Optimization and Numerical Analysis*, volume 275 of *Mathematics and Its Applications*, pp. 51–67. Springer, Dordrecht, 1994. doi: 10.1007/978-94-015-8330-5_4. URL https://doi.org/10.1007/978-94-015-8330-5_4.

Qiskit Aer contributors. Building noise models. Qiskit Aer documentation, [n.d.]. URL https://qiskit.github.io/qiskit-aer/tutorials/3_building_noise_models.html. Accessed September 10, 2026.

Stefan H. Sack and Maksym Serbyn. Quantum annealing initialization of the quantum approximate optimization algorithm. *Quantum*, 5:491, 2021. doi: 10.22331/q-2021-07-01-491. URL https://doi.org/10.22331/q-2021-07-01-491.

Ruslan Shaydulin, Ilya Safro, and Jeffrey Larson. Multistart methods for quantum approximate optimization. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, pp. 1–8. IEEE, 2019. doi: 10.1109/hpec.2019.8916288. URL https://doi.org/10.1109/hpec.2019.8916288.

Ruslan Shaydulin, Phillip C. Lotshaw, Jeffrey Larson, James Ostrowski, and Travis S. Humble. Parameter transfer for quantum approximate optimization of weighted MaxCut. *ACM Transactions on Quantum Computing*, 4(3), 2023. doi: 10.1145/3584706. URL https://doi.org/10.1145/3584706.

Seyon Sivarajah, Silas Dilkes, Alexander Cowtan, Will Simmons, Alec Edgington, and Ross Duncan. t|ket⟩: a retargetable compiler for NISQ devices. *Quantum Science and Technology*, 6(1): 014003, 2021. doi: 10.1088/2058-9565/ab8e92. URL https://doi.org/10.1088/2058-9565/ab8e92.

Kevin J. Sung, Jiahao Yao, Matthew P. Harrigan, Nicholas C. Rubin, Zhang Jiang, Lin Lin, Ryan Babbush, and Jarrod R. McClean. Using models to improve optimizers for variational quantum algorithms. *Quantum Science and Technology*, 5(4):044008, 2020. doi: 10.1088/2058-9565/abb6d9. URL https://doi.org/10.1088/2058-9565/abb6d9.

Shree Hari Sureshbabu, Dylan Herman, Ruslan Shaydulin, Joao Basso, Shouvanik Chakrabarti, Yue Sun, and Marco Pistoia. Parameter setting in quantum approximate optimization of weighted problems. *Quantum*, 8:1231, 2024. doi: 10.22331/q-2024-01-18-1231. URL https://quantum-journal.org/papers/q-2024-01-18-1231/.

Ryan Sweke, Frederik Wilde, Johannes Jakob Meyer, Maria Schuld, Paul K. Fährmann, Barthélémy Meynard-Piganeau, and Jens Eisert. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314, 2020. doi: 10.22331/q-2020-08-31-314. URL https://quantum-journal.org/papers/q-2020-08-31-314/.

Tiingo. End-of-day stock price data documentation, [n.d.]. URL https://www.tiingo.com/documentation/end-of-day. ETF inputs frozen September 1, 2026; source accessed September 10, 2026.

Dave Wecker, Matthew B. Hastings, and Matthias Troyer. Progress towards practical quantum variational algorithms. *Physical Review A*, 92(4):042303, 2015. doi: 10.1103/physreva.92.042303. URL https://doi.org/10.1103/physreva.92.042303.

Debin Xiang, Qifan Jiang, Liqiang Lu, Siwei Tan, and Jianwei Yin. Choco-Q: Commute hamiltonian-based QAOA for constrained binary optimization. In *Proceedings of the IEEE International Symposium on High-Performance Computer Architecture*, pp. 275–289. IEEE, 2025. doi: 10.1109/HPCA61900.2025.00031. URL https://arxiv.org/abs/2503.23941.

Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Physical Review X*, 10(2):021067, 2020. doi: 10.1103/PhysRevX.10.021067. URL https://doi.org/10.1103/PhysRevX.10.021067.

A EXECUTION DETAILS AND REPRODUCIBILITY

Common-start prefix seeds are 119001–119003. At 18/20 assets they cover July 2025–June 2026; at 22 assets they cover July and October 2025 and January and April 2026, for 84 complete contexts. Original continuous seeds are 121001–121003 at 18/20 assets and 123001–123003 at 22. Search-2048, Search-512, Readout-Candidate and RegimeQ complete 864 main-study method windows and eighteen independent RegimeQ endpoint replays. Additional Random-Mix execution completes 648 windows and 54 endpoint replays. Completion and physical-resource ledgers for the temporal and circuit checks appear in Appendices E and G.

The artifact retains stable internal keys: C/Candidate-v2 denotes RegimeQ; F, L, R, M and H denote Search-2048, Search-512, Readout-Candidate, Readout-Mean and Probe-Search-2048. These are presentation names only; no solver, threshold, method output or historical Fixed/Adaptive label was changed.

The eighteen-asset order is Agric, Beer, Fun, Clths, Drugs, Txtls, Steel, ElcEq, Ships, Mines, Util, BusSv, LabEq, Trans, Meals, RlEst, Food, Smoke. Twenty adds Books and Hlth. Twenty-two further adds Chems and BldMt. FF48 inputs end in June 2026. The supplied holdings select the first $n/2$ entries in every window. The evidence package records data, source, configuration, threshold, and execution-order hashes.

The machine has an Intel Core i7-12700H, twenty logical processors and approximately 16 GiB RAM, running Windows build 26200. Experiments use at most two concurrent workers, each with one numerical thread and Aer `max_memory_mb=4096`, CPU statevector method and double precision. Dependencies are Python 3.11.3, NumPy 2.4.6, SciPy 1.17.1, Qiskit 2.5.2, Aer 0.17.2, pandas 3.0.5 and psutil 7.2.2. Classical postprocessing and document generation have their own timing; simulated execution seconds are not hardware timing estimates.

The prefix sampler randomly orders finite-shot outcomes before dividing each probe into four blocks. This permutation changes neither counts nor resource charges. L/H deterministically replay the optimizer's prefix to continue the same search; replayed observations are not physically resubmitted or billed twice. In the common-start experiment the prefix is physically generated once per context, but every logical policy is charged its full prefix. In the continuous experiment each policy is physically executed. Matching random streams do not imply matching samples once inherited parameters differ. Enumerated feasible extrema and exact probability diagnostics are withheld from the online controller.

For optimization statistics, with penalized Ising radius $S_t = \sum_i |a_{i,t}| + \sum_{i<j} |J_{ij,t}|$, use $\widetilde{E} = [E - (c_t - S_t)]/\max(2S_t, 10^{-12})$. COBYLA receives the sampled raw energy mean; the equivalent affine normalization governs selected means and the 0.002 stopping tolerance. The original X experiments do not rescale physical angles; the circuit-transfer settings in Appendix G additionally scale the physical operator. An invalid or interrupted job is logged and excluded, not assigned an artificial gap. No-feasible-prefix fallback H preserves a readout reserve; no-feasible final returns remain explicitly marked. These boundary paths are tested separately from the successful evaluation records.

A.1 DESIGN CONTEXT

Hybrid expectation minimization couples quantum measurements to classical updates (Peruzzo et al., 2014; Wecker et al., 2015); measurement and optimization costs are distinct parts of this loop (Cerezo et al., 2021; Bharti et al., 2022). Measurement-resource analyses motivate charging all estimates (Gonthier et al., 2022). Our diagonal QUBO additionally supplies a candidate bit string with each sample, so the same ledger records estimation cost and feasible return opportunities.

Routing, device calibration and compiler transformations affect the cost of executing a submitted circuit (Li et al., 2019; Murali et al., 2019; Sivarajah et al., 2021). We therefore report calls separately from simulated seconds and circuit counts. The ideal continuous comparisons isolate allocation; earlier noise-model controls using Aer (Qiskit Aer contributors, [n.d.]) appear in the separate supplement.

Other portfolio pipelines combine annealing and gate models (Jain & Chandra, 2023), or decomposition and counterdiabatic optimization (Cadavid et al., 2026). Our comparison instead holds the portfolio objective fixed and varies allocation within QAOA. For the XY circuit package, the Dicke

construction of Bärtschi & Eidenbenz (2019) supplies a feasible initial state; its preparation gates and depth are included in circuit costs. Appendix G specifies this package.

B THRESHOLD DEVELOPMENT AND SELECTION CONTROLS

The frozen execution threshold is $\tau = -0.02347878184789485$ with direction $g \leq \tau \Rightarrow L$. The displayed main-text value is rounded only for readability. Equal-sized block extrema are inspired by Quantile of Maxima (QoMax) for extreme bandits (Baudry et al., 2022); here lower energy is preferable and the readout score is the median of four block minima. Four blocks, the two-half contrast and its minimum were fixed before fitting. Development compares thresholds and directions, not alternative feature constructions; QoMax's theoretical guarantees are not assumed for this rule.

RegimeQ (artifact key Candidate-v2) was developed after earlier sampling rules failed their stated tests. It is a new frozen protocol, not a retroactive success under the earlier gate. Development uses 72 already-inspected common-start contexts at 18/20 assets, seeds 117001–117003, with five actions and two future streams (720 branch results). Candidate thresholds are zero and pooled feature quantiles 10%, 25%, 50%, 75%, 90%, each in both directions, giving twelve rules. An eligible rule must use both branches and have mean calls no greater than F separately at each scale. Eligible rules minimize the equal-scale mean gap, with ties by calls and fixed ordinal. This criterion did not require strict quality improvement at every size.

Leave-one-seed-out fits are internal development checks; each uses only its two training seeds. The final rule is fitted on all development records and frozen before seeds 119001–119003 or 121001–121003 are sampled. The threshold compares a normalized block-minimum contrast, not financial return or exact gap. In the common-start validation, 81/84 prefixes are the inherited vector followed by three $+0.5$ cost-angle probes; the remaining optimizer probes are retained as actually executed. All original $p = 3$ continuous prefixes probe cost angles only.

A retrospective diagnostic applies the original $g > 0 \Rightarrow L$ rule to the same new branches. Its mean gaps at 18/20/22 assets are 3.2286/5.1518/4.5637%; fitting improves the mean only at eighteen assets. It changes both threshold and direction. For a state-independent L/R mixture at C's measured branch proportions, the common-start gap advantage of C changes sign between future repetitions at eighteen assets. This expected mixture is computed offline, not a measured continuous random policy. The full records preserve these outcomes alongside the main table.

C CONTINUOUS QUALITY AND RESOURCE DETAILS

Table 6: Complete recent-period quality and calls, including the budget-limited Search-2048 control; the same evaluation records as Figure 2. Bold denotes the lowest mean or p90 gap at each size.

n	Policy	Mean G (%)	p90 (%)	Calls
18	Search-2048	5.9050	10.5119	8.00
18	Search-512	**2.4145**	**5.3049**	9.42
18	Readout-Candidate	4.4052	9.2526	5.00
18	Random-Mix	3.6035	7.1448	6.48
18	RegimeQ	3.2718	5.6993	6.00
20	Search-2048	6.8363	9.9936	8.00
20	Search-512	**3.6503**	**6.4364**	9.19
20	Readout-Candidate	5.0086	8.4410	5.00
20	Random-Mix	4.9688	8.7926	6.82
20	RegimeQ	4.9651	8.7221	6.72
22	Search-2048	6.9516	10.7790	8.00
22	Search-512	**6.0722**	**10.1308**	9.08
22	Readout-Candidate	7.7263	13.5217	5.00
22	Random-Mix	6.6512	10.4853	6.22
22	RegimeQ	6.9466	11.5370	7.61

Table 7: Recent continuous evaluation hit rates and mean feasible fractions, in percent. Fixed policies and RegimeQ have 36 returns; Random-Mix has 108 nested switching-repeat returns. All use 16,384 shots per window.

n	Policy	$G = 0$	$G \leq 0.1\%$	$G \leq 0.5\%$	$G \leq 1\%$	Feasible
18	Search-2048	2.78	2.78	5.56	8.33	9.07
18	Search-512	8.33	11.11	30.56	44.44	45.18
18	Readout-Candidate	13.89	16.67	27.78	27.78	18.29
18	Random-Mix	10.19	10.19	12.96	16.67	30.36
18	RegimeQ	11.11	11.11	13.89	16.67	38.84
20	Search-2048	2.78	2.78	2.78	2.78	14.12
20	Search-512	5.56	5.56	5.56	8.33	43.66
20	Readout-Candidate	0.00	0.00	0.00	0.00	20.05
20	Random-Mix	3.70	3.70	3.70	3.70	22.03
20	RegimeQ	5.56	5.56	5.56	5.56	17.68
22	Search-2048	0.00	0.00	0.00	2.78	12.03
22	Search-512	0.00	0.00	0.00	2.78	14.84
22	Readout-Candidate	0.00	0.00	2.78	2.78	8.68
22	Random-Mix	0.93	0.93	2.78	2.78	15.34
22	RegimeQ	0.00	0.00	2.78	5.56	18.42

Table 8: Mean G (%) per continuous initialization seed over the twelve primary evaluation months. Each policy uses 16,384 shots per window.

n	Seed	Search-2048	Search-512	Readout-Candidate	RegimeQ
18	121001	6.7631	**1.8197**	4.8266	4.0130
18	121002	7.1146	**1.9134**	3.5073	2.6412
18	121003	3.8374	3.5105	4.8816	**3.1613**
20	121001	6.5812	**3.7193**	5.2560	4.2227
20	121002	5.9289	**3.0948**	5.4420	5.1447
20	121003	7.9989	**4.1368**	4.3280	5.5278
22	123001	7.4029	5.5799	6.1845	**5.0076**
22	123002	7.7502	**6.5162**	7.6747	8.7629
22	123003	**5.7017**	6.1204	9.3196	7.0692

At eighteen/twenty assets, continuous execution completed six paired size–seed blocks, 576 windows and twelve independent replays, with no missing or interrupted windows. The formal run took 3415.745 seconds and physically submitted 9,633,792 shots in 4390 calls including replay. Earlier Candidate-v1/v2 development and common-start verification used 27,576,320 shots and 8912 calls. These initial three rounds total 37,210,112 shots and 13,302 calls; this excludes still earlier context preparation and other project experiments. Online feature processing and solver time remain separate from these development costs. The evidence artifact retains every batch checkpoint and outcome, rather than interpolating success within a submitted batch.

D Random allocation and feedback attribution

The Random-Mix protocol was frozen before its first solve. Its search probability is $18/72 = 0.25$, obtained by applying the already-frozen RegimeQ rule to the original development prefixes. The probability is shared by all sizes and months. A separate pseudorandom stream draws each trajectory's complete action schedule before sampling. L/R selection ignores candidate progress; the no-feasible and constant-objective fallbacks match RegimeQ. The four probes, candidate processing and final readout are all charged. The search and readout branches are unchanged.

The three switching streams at each initialization reuse the original sampling streams, so variation across them measures switching sensitivity conditional on those streams. They are not three

additional independent sampling experiments. Primary comparisons first average over the three switching streams at each initialization and then weight the three initializations equally. Table 9 exposes the paired differences; its standard deviation is descriptive switching variability, not a standard error over independent market instances. Treating months as independent bootstrap units would assume away the shared estimation data; we report the paired observations without claiming interval coverage.

Table 9: RegimeQ minus actual Random-Mix over the primary twelve months, paired by original initialization. Negative differences favor RegimeQ. Mix SD is the sample standard deviation of three switching-trajectory mean gaps, in percentage points. All policies use 16,384 shots per window.

n	Seed	ΔG (pp)	Δ calls	Mix SD
18	121001	0.9451	-0.8889	1.4710
18	121002	-0.0922	-0.7222	0.9217
18	121003	-1.8480	0.1667	0.8290
20	121001	-1.2408	0.2778	1.6084
20	121002	0.2865	-0.5278	1.1171
20	121003	0.9431	-0.0556	0.2941
22	123001	-2.1612	0.4167	2.0277
22	123002	2.8257	2.3056	1.2048
22	123003	0.2217	1.4444	0.4908

Table 10: Evaluation-period call accounting. The readout action always has five calls; search duration depends on its sampled optimizer trajectory. Hence an allocation frequency does not by itself fix mean calls.

n	Policy	Search (%)	Calls given search	Mean calls
18	RegimeQ	19.44	10.14	6.00
18	Random-Mix	23.15	11.40	6.48
20	RegimeQ	41.67	9.13	6.72
20	Random-Mix	27.78	11.57	6.82
22	RegimeQ	38.89	11.71	7.61
22	Random-Mix	22.22	10.50	6.22

Writing K for calls in these ordinary L/R windows, $\mathbb{E}[K] = 5 + p_L(\mathbb{E}[K \mid L] - 5)$. This is an accounting identity, not a cost prediction for another trajectory: both the branch frequency and the conditional search duration can change when inherited angles change. The actual random-control means are therefore measured rather than inferred from the original fixed-action endpoints.

For comparison with the observed fixed endpoints, let $p = (c_C - c_R)/(c_L - c_R)$, using evaluation mean calls c. A random choice of an entire L or R trajectory would have expected mean gap $pG_L + (1-p)G_R$ at c_C calls. The dashed lines in Figure 2 use these observed endpoints; Supplement S10 retains the numerical interpolation. It is a retrospective diagnostic; unlike actual Random-Mix, it never switches actions within a parameter trajectory. Its weight is not proposed for deployment, and p90 cannot be obtained by averaging endpoint p90 values.

The effect of retention can be written exactly. Let G_P be the common prefix's best gap and G_L^+, G_R^+ the respective continuations' best gaps. For the feasible records here,

$$\Delta G^{\text{all}} = \min(G_P, G_L^+) - \min(G_P, G_R^+), \qquad \Delta G^{\text{future}} = G_L^+ - G_R^+. \tag{5}$$

Retention preserves or erases the sign of an individual contrast; it cannot reverse it. It can reverse a pooled mean because it attenuates contrasts unequally. In the eighteen-asset search-assigned group, positive and negative future contributions to the mean are 1.3675 and 1.2353 points. Retention reduces them to 0.9454 and 1.1457. Their difference changes sign. This identity is checked for every common-start L/R pair and explains the pooled reversal without attributing it to a better final distribution. Exact gap values are used only for this offline decomposition.

Table 11: Recent-series calls per trajectory, averaged over initializations and, for Random-Mix, nested switching repetitions. Each complete trajectory uses $24 \times 16,384 = 393,216$ shots, half in warmup. These per-trajectory values are separate from the total execution cost of running every repetition.

n	Policy	Warmup	Evaluation	All 24
18	Search-2048	96.00	96.00	192.00
18	Search-512	125.33	113.00	238.33
18	Readout-Candidate	60.00	60.00	120.00
18	Random-Mix	85.56	77.78	163.33
18	RegimeQ	82.00	72.00	154.00
20	Search-2048	96.00	96.00	192.00
20	Search-512	125.00	110.33	235.33
20	Readout-Candidate	60.00	60.00	120.00
20	Random-Mix	85.11	81.89	167.00
20	RegimeQ	94.00	80.67	174.67
22	Search-2048	96.00	96.00	192.00
22	Search-512	121.67	109.00	230.67
22	Readout-Candidate	60.00	60.00	120.00
22	Random-Mix	88.22	74.67	162.89
22	RegimeQ	76.67	91.33	168.00

Table 12: Common-start conditional contrasts, L minus R, grouped by the frozen RegimeQ decision. Each context has two future sampling repetitions. Negative gap differences favor continued search. "Future only" excludes prefix candidates from return eligibility without resampling.

n	Chosen	Contexts	ΔG all stages	ΔG future only	Δ calls
18	Search-512	16	-0.2003	0.1322	4.1250
18	Readout-Candidate	20	0.3693	0.5076	4.1000
20	Search-512	12	0.4302	0.2074	4.0000
20	Readout-Candidate	24	0.7543	1.0500	4.4792
22	Search-512	3	0.0704	0.0704	4.0000
22	Readout-Candidate	9	0.9253	1.2327	4.4444

This comparison fixes the starting vector and problem and measures complete L/R actions, including parameter selection. The search-assigned groups contain only 16/12/3 contexts at 18/20/22 assets; pooled differences vary across seeds. In continuous runs, inherited vectors diverge, so same-month comparisons cannot identify which branch would have been better at RegimeQ's own start. The monthly plots in Supplement S10 show accumulated outcomes rather than local counterfactual effects.

E TEMPORAL TRANSFER

Table 13: Temporal-transfer tail quality and calls, alongside the means in Table 4A. Bold denotes the lowest mean or p90 gap at each size.

n	Policy	Mean G (%)	p90 (%)	Calls
18	Search-512	5.6826	10.8163	9.83
18	Readout-Candidate	3.2268	5.6947	5.00
18	Random-Mix	2.8468	6.6054	5.97
18	RegimeQ	**2.5790**	**4.4262**	7.44
20	Search-512	3.5730	6.5908	9.06
20	Readout-Candidate	3.7688	6.8074	5.00
20	Random-Mix	4.4400	8.3490	7.83
20	RegimeQ	**3.3619**	**5.9525**	7.58

The temporal check fixes 18/20 assets and the original asset order. It runs July 2017–June 2019, warming up for twelve months and evaluating July 2018–June 2019. These decision months were absent from threshold development. New seeds 125001–125003 initialize all four policies identically before each follows its own trajectory. There is one random switching stream per initialization. Threshold, direction, $p(L) = 0.25$, optimizer and total budget stay fixed. The period was chosen before inspecting its policy outcomes. It tests transfer across decision periods within FF48, rather than transfer to a different problem family.

Table 14: Temporal-check paired differences, RegimeQ minus each alternative; twelve evaluation months per initialization, 16,384 shots per window. Negative values favor RegimeQ.

n	Seed	Contrast	ΔG (pp)	Δ calls
18	125001	C-L	-1.7864	-2.8333
18	125001	C-R	-0.9324	2.3333
18	125001	C-Mix	0.5168	2.0000
18	125002	C-L	-7.1283	-0.8333
18	125002	C-R	-1.0774	3.1667
18	125002	C-Mix	-1.2713	2.8333
18	125003	C-L	-0.3962	-3.5000
18	125003	C-R	0.0664	1.8333
18	125003	C-Mix	-0.0491	-0.4167
20	125001	C-L	-2.6676	-1.0000
20	125001	C-R	-2.1297	3.0000
20	125001	C-Mix	-2.1721	0.5000
20	125002	C-L	2.0345	-3.4167
20	125002	C-R	0.2136	0.7500
20	125002	C-Mix	-0.7507	-0.6667
20	125003	C-L	0.0000	0.0000
20	125003	C-R	0.6954	4.0000
20	125003	C-Mix	-0.3113	-0.5833

The new Random-Mix study completes 648 windows and 54 endpoint replays in 6514.69 seconds of formal wall time, with 11501568 simulator shots and 4869 calls. The temporal check adds 576 windows and 12 replays, 2275.30 seconds, 9633792 shots and 4525 calls. Warmup and replay are

Table 15: Temporal-check calls per trajectory, averaged over three initializations. Each complete trajectory uses 393,216 shots, including 196,608 warmup shots.

n	Policy	Warmup	Evaluation	All 24
18	Search-512	133.67	118.00	251.67
18	Readout-Candidate	60.00	60.00	120.00
18	Random-Mix	101.33	71.67	173.00
18	RegimeQ	106.67	89.33	196.00
20	Search-512	119.67	108.67	228.33
20	Readout-Candidate	60.00	60.00	120.00
20	Random-Mix	92.00	94.00	186.00
20	RegimeQ	102.67	91.00	193.67

Table 16: Temporal-check evaluation quality under both normalizations in Equation (4). Gaps are percentages and A^{ret} is a unitless ratio. The objectives make the two gap normalizations differ in some windows; both retain the same aggregate policy ordering within each scale.

n	Policy	G (%)	G^{range} (%)	A^{ret}
18	Search-512	5.6826	5.8615	0.9414
18	Readout-Candidate	3.2268	3.2991	0.9670
18	Random-Mix	2.8468	2.8894	0.9711
18	RegimeQ	**2.5790**	**2.6421**	**0.9736**
20	Search-512	3.5730	3.6422	0.9636
20	Readout-Candidate	3.7688	3.8202	0.9618
20	Random-Mix	4.4400	4.5139	0.9549
20	RegimeQ	**3.3619**	**3.4270**	**0.9657**

included in these executed-sampling totals. These studies and the sensitivity check below share a four-hour formal limit and run sequentially with at most two workers. Source, input and result hashes are preserved; offline references, analysis and writing are timed separately.

F Budget and probe-order sensitivity

A final protocol holds the inherited vector and QUBO fixed at the four recent evaluation months, with three new sampling seeds 127001–127003 at 18/20 assets. It crosses budgets $8192/16384/32768$ with two optimizer coordinate orders: cost angles first or mixer angles first. Physical binding still uses the original gamma/beta layout. The same four 512-shot probes, feedback formula, threshold, L/R branches and stopping rules are retained. Prefix and future random streams are paired across budgets and orders. These are sensitivity checks on already-inspected workloads, not new transfer instances.

The paired table separates simultaneous losses from quality–call exchanges. Seven configurations are dominated by R on both reported means, while five lower gap at additional call cost. Because the four probes and rule are budget-independent, all 48 shared prefixes choose the same branch at all three budgets. For example, at twenty assets with cost-first probes, C minus R gap is -0.0580 points at 16,384 shots and 0.2200 at 32,768 shots, with the same assignments. The rule responds to which directions were probed but not to this change in action payoff. Some losses also occur at the original budget and order, so budget extrapolation alone cannot explain them. A deployment condition would require an observable predictor validated beyond these inspected records; no such condition is inferred from the offline optimum.

The study executes 48 prefixes and 288 executed continuation branches, plus 6 independent C replays at the first eighteen-asset context, covering both orders and all budgets. Each logical policy pays for its prefix; actual prefix reuse is recorded separately. The study uses 5128192 simulator shots and 1148 calls in 576.39 formal seconds. It runs last within the same combined four-hour limit. The original public trajectories are reused with their previously reported preparation costs.

Table 17: Shared-start sensitivity: RegimeQ (C) minus readout (R) or search (L), twelve seed-month contexts per row. Gap differences are percentage points; K denotes calls. Negative values favor C. R uses five calls in every row. C is reconstructed from the frozen choice among actually sampled L/R branches; absolute means are retained in Supplement S10.

n	First	Shots	ΔG_{C-R}	ΔK_{C-R}	ΔG_{C-L}	ΔK_{C-L}
18	Cost	8192	0.9707	2.00	-0.4176	-2.00
18	Cost	16384	0.8076	2.00	-0.3712	-2.00
18	Cost	32768	0.2502	2.00	-0.4127	-2.00
18	Mixer	8192	-0.1263	3.33	0.4657	-0.67
18	Mixer	16384	0.2235	3.33	0.7376	-0.67
18	Mixer	32768	0.1056	3.33	0.2622	-0.67
20	Cost	8192	0.2944	1.00	0.2203	-3.33
20	Cost	16384	-0.0580	1.00	1.2384	-4.33
20	Cost	32768	0.2200	1.00	0.5116	-4.33
20	Mixer	8192	-1.6405	2.67	0.0162	-1.67
20	Mixer	16384	-1.2879	2.67	0.4342	-1.75
20	Mixer	32768	-0.4013	2.67	0.2200	-1.75

G DEPTH, CIRCUIT AND ETF CHECKS

Table 4B/C uses three new initialization seeds per setting: 129001–129003 for FF48 and 131001–131003 for ETF-10. Each policy runs July 2024–June 2026, with the first twelve months treated as warmup. FF48 uses the same supplied holdings and common QUBOs as the earlier recent-period experiments; ETF-10 uses the last 24 months of its existing frozen common-QUBO sequence. The first vectors are shared across policies, and each policy subsequently inherits its own readout vector. Across depths, the number of angles differs, although their generating stream is shared. Each size–seed pair shares one pre-drawn Random-Mix action sequence across circuit settings. These are new sampling trajectories on examined decision periods. They are reported separately from the original batch and from the independent temporal check.

The raw X circuit retains the original coefficients and penalty at $p = 2, 3, 4$. The scaled X and XY circuits divide their respective nonidentity coefficients by $s_t = \max(\sqrt{\sum_i a_{i,t}^2 + \sum_{i<j} J_{ij,t}^2}, 10^{-12})$; their inherited angles are expressed in that scale. XY removes only the cardinality penalty, preserving the business objective. It prepares a Dicke state using the ancilla-free gate construction of Bärtschi & Eidenbenz (2019), then uses one ring-XY Trotter step per layer, even edges before odd edges including the wraparound edge. Each edge applies $RXX(\beta)RYY(\beta)$. No statevector injection is used. Its optimization/stopping normalization uses the unpenalized operator's coefficient bound. This comparison changes initial state, mixer, constraint encoding and the associated normalization together. It does not isolate a mixer effect. All feasible returns are evaluated with the original common business objective.

The four 512-shot probes, threshold, direction, $p(L) = 0.25$, mean objective, COBYLA settings, all-stage candidate retention and $16,384$-shot window budget remain unchanged. No detector RESET is used. Each of the 33 paired blocks completes all four 24-month trajectories: 3168 method windows, plus 66 independent RegimeQ first/last-window replays, with no missing or interrupted windows. Formal wall time is 12,883.0873 seconds; including replay, physical execution uses 52,985,856 shots and 23,746 calls. Validation covers backend probability agreement, Dicke preparation, XY feasibility, frozen-rule replay and resource accounting.

Table 18: Tail quality for Table 4B/C: p90 G (%) over 36 evaluation returns per cell. Bold marks the lowest p90 in each row; all ties are shown.

Setting	n	Search-512	Readout-Candidate	Random-Mix	**RegimeQ**
X, raw, $p = 2$	18	9.1796	8.5835	5.7595	**5.3483**
X, raw, $p = 2$	20	6.9762	**6.8799**	8.3646	8.4180
X, raw, $p = 3$	18	8.1307	10.8561	**5.9830**	6.8162
X, raw, $p = 3$	20	**8.6478**	10.4636	9.5124	9.9145
X, raw, $p = 4$	18	**6.5388**	6.9946	7.0729	7.3800
X, raw, $p = 4$	20	**8.2918**	11.4089	10.3069	9.7482
X, scaled, $p = 3$	18	7.8559	8.1775	8.5936	**7.1386**
X, scaled, $p = 3$	20	**8.4899**	10.0721	9.8381	11.8173
XY, scaled, $p = 3$	18	3.5101	3.6464	3.5654	**3.0856**
XY, scaled, $p = 3$	20	**2.9720**	4.6024	3.5834	4.6024
ETF-10, X, $p = 3$	10	**0.0000**	**0.0000**	**0.0000**	**0.0000**

Table 19: Full 24-month calls per trajectory, averaged across three initializations, including warmup. Every trajectory uses 393,216 shots. Subtracting twelve times the evaluation mean calls in Table 4B/C recovers its mean warmup cost.

Setting	n	Search-512	Readout-Candidate	Random-Mix	**RegimeQ**
X, raw, $p = 2$	18	232.67	120.00	156.67	187.33
X, raw, $p = 2$	20	232.67	120.00	165.33	165.00
X, raw, $p = 3$	18	244.67	120.00	174.33	172.33
X, raw, $p = 3$	20	242.00	120.00	163.33	201.67
X, raw, $p = 4$	18	234.67	120.00	168.00	178.67
X, raw, $p = 4$	20	251.67	120.00	167.00	161.67
X, scaled, $p = 3$	18	239.00	120.00	161.00	189.67
X, scaled, $p = 3$	20	236.00	120.00	167.00	185.67
XY, scaled, $p = 3$	18	279.67	120.00	166.67	145.00
XY, scaled, $p = 3$	20	300.67	120.00	160.33	122.00
ETF-10, X, $p = 3$	10	229.67	120.00	165.33	172.67

Table 20: RegimeQ minus Random-Mix for each initialization: evaluation mean gap difference ΔG (percentage points) and mean call difference ΔK. Seeds 1–3 denote 129001–129003 for FF48 and 131001–131003 for ETF-10. Each difference compares twelve months at 16,384 shots per window. Negative values favor RegimeQ.

Setting	n	Seed 1		Seed 2		Seed 3	
		ΔG	ΔK	ΔG	ΔK	ΔG	ΔK
X, raw, $p = 2$	18	1.3421	2.58	-0.7476	0.17	-2.1872	-0.33
X, raw, $p = 2$	20	0.2308	1.83	0.0312	0.83	-1.0645	1.08
X, raw, $p = 3$	18	0.6039	2.75	-1.4582	-0.33	1.7871	0.17
X, raw, $p = 3$	20	0.7485	4.75	0.9624	2.00	-5.6984	3.75
X, raw, $p = 4$	18	0.4704	1.50	1.5162	0.42	0.3275	-0.92
X, raw, $p = 4$	20	-1.3171	3.92	-3.5268	1.00	1.3649	0.42
X, scaled, $p = 3$	18	0.2749	2.42	-2.0724	1.67	0.1546	2.08
X, scaled, $p = 3$	20	2.0600	1.75	2.1716	0.50	-3.2224	1.17
XY, scaled, $p = 3$	18	-0.7859	-0.33	-0.1808	-0.75	-0.4469	-0.58
XY, scaled, $p = 3$	20	1.2077	-0.92	0.2761	-1.83	0.9523	-0.75
ETF-10, X, $p = 3$	10	0.0000	0.08	0.0000	0.75	0.0000	1.75

At eighteen-asset XY, RegimeQ improves evaluation mean gap and calls over Random-Mix for each initialization. Including warmup, its three trajectories total 435 calls, versus 500 for Random-Mix. At twenty-asset XY, it chooses readout in all 36 warmup windows and 35 of 36 evaluation windows; two complete initialization trajectories are identical to fixed readout. This explains the small C/R difference without treating it as a broadly improved allocation. In both scales, all ideal XY samples are feasible. Its gate cost is higher: at 18/20 qubits, scaled X has 918/1140 two-qubit gates and depth 228/252, whereas XY has 2068/2552 and depth 1712/1913, including Dicke preparation. These counts are simulator-basis circuit costs, not routed hardware costs.

Depth changes the meaning of a fixed-length probe prefix. At $p = 2$ the first four evaluations reach one mixer coordinate; at $p = 3, 4$ they only reach cost coordinates. At eighteen-asset raw $p = 4$, RegimeQ selects search in 17 evaluation windows, but 11 stop after eight evaluations, before completing the nine-point initial simplex. This is an observed interaction between probe coverage and stopping, rather than an isolated causal explanation of the quality loss. ETF-10 has 252 feasible portfolios; RegimeQ already finds an optimum during the four probes in 33 of 36 evaluation windows, and every method eventually reaches zero gap. At this budget it is a saturation check, with no evidence of improved candidate quality from switching.