

Reproducible QUBO–QAOA Benchmark for Cardinality-Constrained Mean–Variance Portfolio Selection

Ata Sani Kılıç
The MacDuffie School
Granby, MA 01033, USA
akilic2027@macduffie.org

Ercan Erkalkan
Vocational School of Technical Sciences, Marmara University
Istanbul, Türkiye
ercan.erkalkan@marmara.edu.tr

Abstract—Cardinality-constrained mean–variance portfolio selection is a practical decision-support problem in finance and business analytics and a compact benchmark for combinatorial optimization. A Quadratic Unconstrained Binary Optimization (QUBO) model is derived using an equal-weight utility proxy and an instance-adaptive quadratic penalty that enforces a fixed number of selected assets. A depth-one Quantum Approximate Optimization Algorithm (QAOA) solver is evaluated on a classical statevector simulator and compared against an exact feasible-enumeration baseline on small problem sizes. A fully reproducible Python implementation is described, including QUBO construction, penalty calibration, and a (γ, β) grid-search protocol that minimizes expected energy, followed by a top- M post-processing rule for returning a feasible portfolio. Across 10 synthetic instances per problem size, feasibility is achieved in all runs for $N \in \{8, 10, 12\}$ with $K = 3$ under the default setting ($\alpha = 10$, $M = 256$). An ablation quantifies that top- M post-processing is critical for maintaining feasibility under a standard mixer, while penalty scaling exhibits an explicit feasibility–quality trade-off at $N = 12$.

Keywords—portfolio optimization, quadratic unconstrained binary optimization (qubo), quantum approximate optimization algorithm (qaoa), artificial intelligence, optimization, data science

I. INTRODUCTION

Mean–variance portfolio theory is a foundational model for risk-aware investment selection [1]. In real decision settings, discrete constraints are common, including limits on the number of selected assets. Cardinality constraints transform the continuous mean–variance problem into a combinatorial optimization task, motivating exact and heuristic solvers.

Quadratic Unconstrained Binary Optimization (QUBO) provides a unified modeling layer for many combinatorial problems, including finance applications [2]–[4]. Quantum Approximate Optimization Algorithm (QAOA) [5] is a variational approach designed for approximate solutions of such objectives. In constraint-heavy regimes, extensions such as alternating-operator variants have been proposed [6].

Prepared for ICECET 2026. A reproducible simulator-based study implemented in Python is provided as supplementary material.

This paper presents a reproducible, conference-length study that links a finance decision problem (risk–return trade-off) to computer science methods (QUBO modeling and algorithmic evaluation) and to introductory quantum algorithm concepts (cost Hamiltonians and mixing operators).

Contributions:

- A compact QUBO formulation for cardinality-constrained mean–variance selection using an equal-weight utility proxy and an instance-adaptive penalty calibration rule.
- A simulator-based depth-one QAOA evaluation protocol (expected-energy grid search + top- M post-processing) implemented in pure Python/NumPy.
- A small-scale benchmark study with an exact feasible-enumeration baseline, reporting feasibility, exact-hit rate, and optimality-gap scaling for $N \in \{8, 10, 12\}$.

extbfPaper organization: Section II summarizes related work on QUBO modeling and quantum optimization in finance. Section III defines the cardinality-constrained mean–variance proxy and derives the penalized QUBO objective. Section IV describes the exact baseline and the depth-one QAOA simulator workflow. Section V reports experimental settings. Section VI presents results and ablations. Section VII concludes the study and outlines future directions.

II. BACKGROUND AND RELATED WORK

A. Discrete Portfolio Selection and Cardinality Constraints

Mean–variance portfolio selection is classically posed as a continuous quadratic program [1]. Operational constraints often introduce discreteness, including restrictions on the number of holdings, minimum lots, and eligibility rules. Cardinality constraints create a combinatorial search space of size $\binom{N}{K}$ and motivate specialized methods, ranging from branch-and-bound and mixed-integer quadratic programming to heuristic metaheuristics.

B. QUBO/Ising Modeling as a Unifying Layer

Binary quadratic models provide a convenient interface between finance-motivated selection problems and generic combinatorial optimization engines. The Quadratic Unconstrained Binary Optimization (QUBO) form is closely related to Ising formulations used in physics and in adiabatic/anncaling-style optimization [4]. Penalty terms are a standard mechanism for converting constrained problems into unconstrained quadratic objectives. Tutorial treatments highlight practical design choices for penalty magnitudes, scaling, and verification of constraint enforcement [7]. The present study adopts an instance-adaptive penalty scaled by return and covariance magnitudes, providing a simple rule that stabilizes feasibility across randomly generated instances.

C. Quantum Optimization in Finance

Quantum optimization has been explored in finance through both annealing-style approaches and gate-model variational algorithms. Reverse quantum annealing has been studied for portfolio optimization as a hybrid workflow combining classical seeds with quantum-assisted local improvement [2]. Quantum-annealer case studies and implementation notes have also been reported for small portfolio instances [3]. Broader overviews of quantum computing in finance survey portfolio selection together with option pricing, risk analysis, and optimization pipelines [8], [9].

D. QAOA and Constraint Handling

QAOA alternates between a cost Hamiltonian and a mixing Hamiltonian, producing a parameterized state whose measurement distribution is biased toward low-energy bitstrings [5]. Empirical and theoretical analyses discuss parameter structure, optimization strategies, and implementation patterns on near-term devices [10]. For hard constraints, alternating-operator formulations and constraint-aware mixers have been proposed to preserve feasible subspaces, reducing reliance on large penalties [6]. The present study intentionally retains a penalty-based formulation to keep the benchmark compact and reproducible, while quantifying the impact of a simple post-processing rule (top- M selection) that mitigates non-cardinality-preserving mixing behavior.

III. PROBLEM FORMULATION

Let N denote the number of assets with expected returns $\boldsymbol{\mu} \in \mathbb{R}^N$ and covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{N \times N}$. A binary selection vector $\mathbf{x} \in \{0, 1\}^N$ indicates asset inclusion. To emphasize the discrete selection decision, an equal-weight utility proxy is adopted:

$$J(\mathbf{x}) = \frac{1}{K} \boldsymbol{\mu}^\top \mathbf{x} - \lambda \frac{1}{K^2} \mathbf{x}^\top \boldsymbol{\Sigma} \mathbf{x}, \quad \text{s.t.} \quad \mathbf{1}^\top \mathbf{x} = K, \quad (1)$$

where $\lambda > 0$ tunes risk aversion and K is the desired number of selected assets. The scaling by K and K^2 matches equal-weight aggregation and improves penalty calibration stability.

A. Penalty Relaxation to QUBO

Constraint handling uses a quadratic penalty:

$$E(\mathbf{x}) = -J(\mathbf{x}) + A (\mathbf{1}^\top \mathbf{x} - K)^2, \quad (2)$$

where $A > 0$ is selected to discourage infeasible cardinalities. Because $x_i^2 = x_i$ for binary variables, (2) is a QUBO objective.

B. Penalty Calibration

A simple instance-adaptive choice is used:

$$A = \alpha \left(\mu_{\max} + \lambda \frac{\Lambda_{\max}(\boldsymbol{\Sigma})}{K} \right), \quad (3)$$

where $\mu_{\max} = \max_i \mu_i$, $\Lambda_{\max}(\boldsymbol{\Sigma})$ is the largest eigenvalue of the covariance matrix, and α is a safety factor. This scaling is designed so that the penalty term dominates typical variations of the risk–return proxy under the equal-weight normalization in (1), improving cardinality enforcement stability.

This choice scales the penalty to dominate the magnitude of the utility term under the equal-weight proxy.

C. Explicit QUBO Coefficients

For implementation, (2) can be expanded into a standard quadratic form with linear and pairwise terms. Using symmetry of $\boldsymbol{\Sigma}$ and the identity $x_i^2 = x_i$ for binary variables,

$$\mathbf{x}^\top \boldsymbol{\Sigma} \mathbf{x} = \sum_{i=1}^N \Sigma_{ii} x_i + 2 \sum_{1 \leq i < j \leq N} \Sigma_{ij} x_i x_j. \quad (4)$$

Similarly, the penalty expands as

$$(\mathbf{1}^\top \mathbf{x} - K)^2 = \sum_{i=1}^N x_i + 2 \sum_{1 \leq i < j \leq N} x_i x_j - 2K \sum_{i=1}^N x_i + K^2. \quad (5)$$

Collecting terms yields

$$E(\mathbf{x}) = \sum_{i=1}^N h_i x_i + \sum_{1 \leq i < j \leq N} J_{ij} x_i x_j + AK^2, \quad (6)$$

with coefficients

$$h_i = -\frac{\mu_i}{K} + \lambda \frac{\Sigma_{ii}}{K^2} + A(1 - 2K), \quad (7)$$

$$J_{ij} = 2A + 2\lambda \frac{\Sigma_{ij}}{K^2} \quad (i < j). \quad (8)$$

This explicit form supports direct construction of the diagonal cost Hamiltonian H_C and enables verification of penalty dominance for cardinality enforcement.

IV. METHODS

A. Exact Baseline (Feasible Enumeration)

For fixed K and small N , the exact optimum is obtained by enumerating all $\binom{N}{K}$ feasible selections satisfying $\mathbf{1}^\top \mathbf{x} = K$ and minimizing the unpenalized objective $-J(\mathbf{x})$ in (1). This is equivalent to minimizing the penalized energy $E(\mathbf{x})$ in (2) over the feasible set, because the penalty term $A(\mathbf{1}^\top \mathbf{x} - K)^2$ is identically zero for all feasible configurations. The resulting optimum provides a ground-truth reference for exact-hit and optimality-gap reporting in the studied regime.

Algorithm 1 Depth-one QAOA with grid search and top- M post-processing

- 1: **Input:** energy table $E(z)$ for $z \in \{0, 1\}^N$, grids $\Gamma, \mathcal{B}$, top set size M
 - 2: **Output:** selected bitstring $\hat{z}$
 - 3: Initialize $|\psi_0\rangle \leftarrow |+\rangle^{\otimes N}$ (uniform amplitudes)
 - 4: **for all** $\gamma \in \Gamma$ **do**
 - 5: Apply phase operator: $|\psi_C\rangle \leftarrow e^{-i\gamma H_C} |\psi_0\rangle$
 - 6: **for all** $\beta \in \mathcal{B}$ **do**
 - 7: Apply mixer: $|\psi\rangle \leftarrow e^{-i\beta \sum_j X_j} |\psi_C\rangle$
 - 8: Compute probabilities $p(z) = |\langle z|\psi\rangle|^2$
 - 9: Compute expected energy $\mathbb{E}[E] \leftarrow \sum_z p(z)E(z)$
 - 10: **end for**
 - 11: **end for**
 - 12: Select (γ^*, β^*) minimizing $\mathbb{E}[E]$
 - 13: Compute $p^*(z)$ under (γ^*, β^*) and take top- M indices $\mathcal{T}$
 - 14: Return $\hat{z} \leftarrow \arg \min_{z \in \mathcal{T}} E(z)$
-

B. Depth-One QAOA on a Classical Simulator

Depth-one QAOA prepares the uniform superposition state and applies one phase-separation and one mixing step:

$$|\psi(\gamma, \beta)\rangle = e^{-i\beta \sum_{j=1}^N X_j} e^{-i\gamma H_C} |+\rangle^{\otimes N}, \quad (9)$$

where H_C is the diagonal cost Hamiltonian encoding $E(\mathbf{x})$ and X_j is Pauli-X on qubit j .

Parameter selection: (γ, β) are selected by minimizing the expected energy $\mathbb{E}_{p_{\gamma, \beta}}[E]$ over a grid.

Returned portfolio: after selecting (γ, β) , the output portfolio is chosen as the minimum-energy bitstring among the top M most probable measurement outcomes (with $M = 256$), which improves feasibility and reduces sensitivity to probability ties.

Algorithm 1 summarizes the procedure.

C. Computational Complexity and Resource Scaling

For each instance, feasible enumeration evaluates $\binom{N}{K}$ selections. With a dense covariance matrix, direct evaluation of $\mathbf{x}^\top \Sigma \mathbf{x}$ costs $\mathcal{O}(N^2)$ per candidate, yielding $\mathcal{O}\left(\binom{N}{K} N^2\right)$ time overall. The statevector simulator requires an explicit energy table over all 2^N bitstrings and therefore incurs $\mathcal{O}(2^N)$ memory. For a grid of $|\Gamma| \times |\mathcal{B}|$ parameters, depth-one QAOA evaluates expected energy values over all bitstrings and applies N single-qubit mixing operations, yielding a dominant scaling of

$$T_{\text{sim}} = \mathcal{O}(|\Gamma| |\mathcal{B}| N 2^N). \quad (10)$$

The selected study regime ($N \leq 12$ and a 21×21 grid) remains computationally lightweight and is suitable for classroom-scale reproducibility.

V. EXPERIMENTAL SETUP

A. Instance Generation

To isolate algorithmic behavior from data-access constraints, synthetic instances are used. For each $N \in \{8, 10, 12\}$, 10

TABLE I
DEPTH-ONE QAOA VS EXACT FEASIBLE OPTIMUM (10 INSTANCES PER N , $K = 3$, $\lambda = 0.6$, GRID SIZE 21×21 , $M = 256$)

N	Mean gap	Median gap	Std gap	Exact-hit rate	Feasible rate
8	0.000	0.000	0.000	1.00	1.00
10	0.013	0.000	0.041	0.90	1.00
12	0.250	0.268	0.056	0.00	1.00

instances are generated with fixed seeds. Expected returns are sampled uniformly from $[0.02, 0.15]$. Covariances are generated as $\Sigma = \mathbf{A}\mathbf{A}^\top$ and scaled so that the maximum variance (diagonal entry) equals 0.05. Parameters are fixed to $\lambda = 0.6$, $K = 3$, $\alpha = 10$, and penalty A is computed by (3).

B. QAOA Parameter Search

Depth is set to $p = 1$. A grid search uses 21 values for γ in $[0, 2\pi]$ and 21 values for β in $[0, \pi]$. Top-set size is fixed to $M = 256$.

C. Metrics

- **Feasibility rate:** fraction of runs with selected cardinality equal to K .
- **Exact-hit rate:** fraction of runs matching the exact optimum energy within numerical tolerance.
- **Relative optimality gap:** $(E_{\text{qaoa}} - E_{\text{opt}})/|E_{\text{opt}}|$.

D. Additional Evaluations

Two focused ablations are included to clarify design choices that affect feasibility and solution quality.

Effect of top- M post-processing: The post-processing size M is varied in $\{1, 16, 256\}$ while keeping all other settings fixed. $M = 1$ corresponds to returning the single most probable bitstring (maximum-likelihood decoding), whereas larger M allows selection of a low-energy candidate among the dominant measurement outcomes.

Penalty scaling sensitivity: The safety factor α in (3) is varied in $\{1, 2, 5, 10, 20\}$ at $N = 12$ with $M = 256$. This study characterizes the trade-off between conservative constraint emphasis (larger α) and objective fidelity (smaller α) under a standard, non-cardinality-preserving mixer.

VI. RESULTS AND DISCUSSION

Table I reports averaged results across 10 instances per N . Feasibility is achieved in all runs under the instance-adaptive penalty. Exact optima are recovered consistently for $N = 8$ and frequently for $N = 10$. For $N = 12$, gaps increase under depth-one QAOA, consistent with limited circuit expressivity at fixed depth.

A. Ablation and Sensitivity Findings

Table II demonstrates that maximum-likelihood decoding ($M = 1$) frequently returns infeasible cardinalities as N increases. This behavior is expected because the standard mixer $e^{-i\beta \sum_j X_j}$ does not preserve Hamming weight, and the

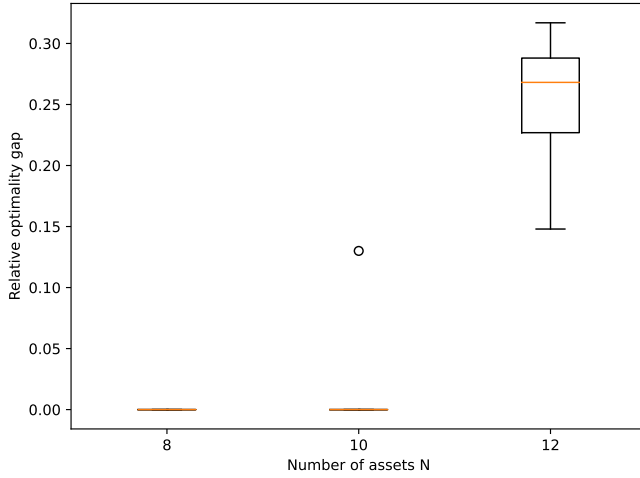


Fig. 1. Distribution of relative optimality gaps across 10 instances per N under depth-one QAOA.

TABLE II
EFFECT OF TOP- M POST-PROCESSING ON FEASIBILITY AND SOLUTION QUALITY ($\alpha = 10$; 10 INSTANCES PER N).

M	Feasible rate			Mean gap		
	$N = 8$	$N = 10$	$N = 12$	$N = 8$	$N = 10$	$N = 12$
1	0.900	0.500	0.000	0.488	0.130	0.953
16	1.000	0.700	0.100	0.301	0.241	0.780
256	1.000	1.000	1.000	0.000	0.013	0.250

measurement mode can shift toward high-probability but infeasible bitstrings when the penalty is not sufficiently dominant in the sampled distribution.

Increasing the post-processing set size substantially improves feasibility by allowing the selection of a low-energy candidate among the dominant measurement outcomes. In particular, $M = 256$ restores feasibility to 1.00 across $N \in \{8, 10, 12\}$ and simultaneously reduces mean gaps, indicating that feasibility and solution quality can be jointly improved via hybrid decoding when a non-cardinality-preserving mixer is used.

Penalty scaling sensitivity is summarized in Table III for $N = 12$ with $M = 256$. Small α weakens constraint emphasis and leads to non-negligible feasibility loss, even under top- M decoding. Larger α values recover feasibility but can alter the energy landscape by overweighting the penalty component, which may increase or decrease approximation gaps depending on the instance distribution. The selected default $\alpha = 10$ is therefore a conservative feasibility-first setting used for the main benchmark, while Table III quantifies the trade-off between constraint enforcement robustness and objective fidelity.

B. Interpretation

Under a penalty formulation, the standard QAOA mixer does not preserve the fixed-cardinality subspace, so feasibility

TABLE III
PENALTY SCALING SENSITIVITY AT $N = 12$ WITH $M = 256$ (10 INSTANCES; $\lambda = 0.6$, $K = 3$).

α	Feasible rate	Exact-hit rate	Mean gap	Median gap
1	0.60	0.10	0.41	0.43
2	0.90	0.10	0.31	0.29
5	1.00	0.20	0.21	0.19
10	1.00	0.00	0.25	0.27
20	1.00	0.10	0.19	0.21

cannot be guaranteed at the circuit level. The benchmark therefore relies on two stabilizers: (i) an instance-adaptive penalty magnitude and (ii) top- M post-processing that selects the best-energy candidate among high-probability outcomes. The growth of optimality gaps at $N = 12$ under depth-one QAOA is consistent with limited circuit expressivity at fixed depth. Improved approximation quality at larger N is expected to require higher depth and/or constraint-aware mixers and alternating-operator constructions that act directly on the feasible subspace [6].

C. Limitations

Synthetic statistics and an equal-weight utility proxy are used to isolate algorithmic behavior. Real-market studies require return estimation, rolling-window stability analysis, transaction-cost modeling, and robustness checks under regime shifts. The penalty formulation and standard mixer do not enforce cardinality at the circuit level; feasibility is supported here through a conservative penalty and post-processing, and Table III indicates that penalty scaling can materially affect approximation quality. Simulator-based runtime does not represent execution time on quantum hardware, and noise effects (gate errors and measurement errors) are not modeled.

D. Reproducibility

A public reproducibility repository is provided at github.com/ErcanErkalkan/icecet-qubo-qaoa-portfolio. All experiments in this paper are regenerated by a single script, `scripts/reproduce.py`, using a fixed seed schedule to define the complete benchmark set (no instance filtering or cherry-picking).

End-to-end regeneration: after installing dependencies from `requirements.txt`, the full set of tables/figures and CSV logs is produced by

```
python scripts/reproduce.py --outdir artifacts
```

The command writes the following outputs into the specified `artifacts/` directory: (i) per-instance log `results_instances.csv`, (ii) aggregate summary `results_summary.csv`, (iii) ablation logs `results_ablation_topM.csv` and `results_alpha_sensitivity.csv`, and (iv) the figure file `fig_gap_boxplot.pdf`. The script regenerates all synthetic instances (with fixed seeds), constructs the QUBO (equal-weight proxy + calibrated quadratic penalty), computes exact optima via feasible enumeration over $\binom{N}{K}$

selections, runs depth-one QAOA statevector simulation with a 21×21 (γ, β) grid that minimizes expected energy, and applies the reported hybrid top- M post-processing rule.

Link to manuscript artifacts: the `results/` directory in the repository contains the reference CSV outputs used in the manuscript, while the reproduction run writes fresh outputs to `artifacts/` for direct comparison. In addition, the repository includes a GitHub Actions workflow (`.github/workflows/ci.yml`) that executes `reproduce.py` as a regression test on each update.

As a result, Table I, Fig. 1, and the ablation summaries in Tables II–III are reproducible from a clean environment using the single command above.

VII. CONCLUSION

A QUBO formulation for cardinality-constrained mean-variance selection was evaluated with depth-one QAOA on a classical statevector simulator and compared to an exact feasible-enumeration baseline. Instance-adaptive penalties combined with a top- M post-processing rule yield feasible portfolios consistently for $N \in \{8, 10, 12\}$ under the studied setting. Exact solutions are recovered on smaller instances, while optimality gaps increase with asset count at fixed depth and with conservative penalty scaling.

Future work includes (i) higher QAOA depth with structured parameter schedules, (ii) constraint-preserving mixers and alternating-operator variants that operate directly on fixed-cardinality subspaces [6], and (iii) a real-market evaluation pipeline with rolling estimation windows, stability analysis, and transaction-cost-aware performance metrics. The provided reproducibility package is intended as a compact benchmark and an educational bridge between portfolio theory, binary quadratic modeling, and variational quantum algorithms.

ACKNOWLEDGMENT

Support and mentorship contributions are acknowledged. Background preparation included an introductory machine learning course with Python (July 2024) and an introductory quantum physics course (June 2025), together with applied software practice through internship experience.

REFERENCES

- [1] H. Markowitz, “Portfolio selection,” *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.
- [2] D. Venturelli and A. Kondratyev, “Reverse quantum annealing approach to portfolio optimization problems,” *Quantum Machine Intelligence*, vol. 1, no. 1-2, pp. 17–30, 2019.
- [3] F. Phillipson and H. S. Bhatia, “Portfolio optimisation using the D-Wave quantum annealer,” 2020. [Online]. Available: <https://arxiv.org/abs/2012.01121>
- [4] A. Lucas, “Ising formulations of many NP problems,” *Frontiers in Physics*, vol. 2, p. 5, 2014. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fphy.2014.00005/full>
- [5] E. Farhi, J. Goldstone, and S. Gutmann, “A quantum approximate optimization algorithm,” 2014. [Online]. Available: <https://arxiv.org/abs/1411.4028>
- [6] S. Hadfield, “From the quantum approximate optimization algorithm to a quantum alternating operator ansatz,” *Algorithms*, vol. 12, no. 2, p. 34, 2019. [Online]. Available: <https://www.mdpi.com/1999-4893/12/2/34>
- [7] F. Glover, G. Kochenberger, and Y. Du, “A tutorial on formulating and using QUBO models,” 2018. [Online]. Available: <https://arxiv.org/abs/1811.11538>
- [8] R. Orús, S. Mugel, and E. Lizaso, “Quantum computing for finance: Overview and prospects,” *Reviews in Physics*, vol. 4, p. 100028, 2019.
- [9] D. J. Egger, C. Gambella, J. Marecek, S. McFaddin, M. Mevissen, R. Raymond, A. Simonetto, S. Woerner, and E. Yndurain, “Quantum computing for finance: State of the art and future prospects,” *IEEE Transactions on Quantum Engineering*, vol. 1, pp. 1–24, 2020, art. no. 3101724. [Online]. Available: <https://arxiv.org/abs/2006.14510>
- [10] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, “Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices,” *Physical Review X*, vol. 10, no. 2, p. 021067, 2020.