Markowitz Portfolio Optimization with quantum computing

THOMASSIN Pablo, COZ Olivier, BERDOUS Louiza
M2MO - Université Paris Cité

April 6, 2025

Overview

Context & Literature review

Context: Quantum Computing Revolution

- **21st Century Frontier Technology**:
  - Potential to transform computational capabilities
  - Particularly promising for combinatorial optimization
  - Financial applications as primary use case
- **Current Industry Landscape**:
  - Major players: Google, IBM, Intel, Rigetti, QuTech, D-Wave, IonQ
  - Active development of quantum hardware and algorithms
  - Growing software ecosystem to enable practical applications
- **Significance**: Bridge between theoretical quantum advantage and real-world financial problem-solving

Portfolio Optimization & Mathematical structure

- **Portfolio Optimization**: Classical problem in finance
  - Introduced by Markowitz: efficient mean-variance combinations
  - Focus: Minimizing risk under budget and return constraints
  - Application: Family trusts, pension funds with future liabilities
- **Mathematical Nature**: Quadratic optimization with binary variables
  - Binary Quadratic Programming (BQP) problems are NP-hard
  - Classical solvers improving but face limitations for large instances

Quantum Computing & Finance

- **Quantum Computing**: Emerging computational paradigm
  - Leverages quantum phenomena: superposition, entanglement, interference
  - Two paradigms: gate-based and quantum annealers
  - Current state-of-art: $\sim$70 qubits (gate-based), 5000 qubits (annealers)
- **Portfolio Optimization**: "Sweet spot" for quantum computing
  - Quadratic structure aligns with quantum hardware strengths
  - Potential for computational advantage in financial applications

Previous Work in Quantum Portfolio Optimization

- **D-Wave Implementations**:
  - Rosenberg et al. (D-Wave One, 128 qubits): 63 potential investments
  - Reverse quantum annealing for Sharpe ratio optimization
  - Graph-theoretic maximum independent set (MIS) approaches
  - D-Wave 2000Q: Portfolio selection from 40-60 U.S. liquid equities
- **Gate-based Approaches**:
  - Algorithms with poly(log(N)) complexity
  - Quantum Amplitude Estimation (QAE) for objective function approximation

Focus of Our Study: Phillipson & Bhatia (2020)

- **Key Contributions**:
  - Assessment of D-Wave Advantage system (5000 qubits)
  - Hybrid quantum-classical approach to portfolio optimization
  - Empirical comparison with state-of-the-art classical solvers
  - Real-world test cases: Nikkei225 and S&P500 indices
- **Research Value**:
  - Practical rather than theoretical assessment
  - Benchmarking against commercial solvers (LocalSolver, Gurobi)
  - Evaluation of current quantum technology for financial applications

Theoretical Results

Theoretical Results: Portfolio Optimization Problem Formulation

- **Problem Components:**
  - N assets to invest in: $P_1, \ldots, P_N$
  - Expected return of asset i: μ_i
  - Risk (standard deviation) of asset i: σ_i
  - Correlation between assets i and j: ρ_{ij}
  - Return vector: $\mu = \{\mu_i\}$
  - Risk matrix: $\Sigma = \{\sigma_{ij}\}$ where:
    - $\sigma_{ij} = \sigma_i^2$ if $i = j$
    - $\sigma_{ij} = \rho_{ij}\sigma_i\sigma_j$ if $i \neq j$

Mathematical Formulation

Optimization Problem:

$$
\begin{aligned}
\min \quad & x^T \Sigma x \\
\text{s.t.} \quad & \sum_{i=1}^{N} x_i = n \\
& \mu^T x \geq R^* \\
& x_i \in \{0, 1\}
\end{aligned}
$$

Decision Variables:

- $x_i = 1$ if asset i is selected
- $x_i = 0$ otherwise

Problem Characteristics:

- Quadratic objective function
- Constrained optimization
- Binary decision variables
- Budget constraint: select n assets
- Return constraint: achieve target return R^*
- Goal: Minimize portfolio risk

QUBO Formulation Background

- **Quadratic Unconstrained Binary Optimization (QUBO):**
  - Standard form for quantum annealing problems
  - Expressed as: $\min / \max y = x^T Qx$
  - $x \in \{0, 1\}^n$ are binary decision variables
  - Q is an $n \times n$ coefficient matrix
- **Alternative QUBO Formulations:**
  - $\min / \max H = x^T q + x^T Qx$
  - $\min / \max H = \lambda_1 \cdot H_1 + \lambda_2 \cdot H_2 + \cdots$
  - Weights $\lambda_1, \lambda_2, \ldots$ enable constraint encoding

Converting Constrained Problems to QUBO

- **For constrained binary programming problems:**

$$\begin{aligned} \min y &= c^T x \\ \text{s.t.} \quad Ax &= b \end{aligned}$$

- **Transformation to QUBO:**

$$\begin{aligned} \min y &= c^T x + \lambda(Ax - b)^T(Ax - b) \\ &= x^T P x + \lambda(Ax - b)^T(Ax - b) \\ &= x^T P x + x^T R x + d \\ &= x^T Q x \end{aligned}$$

where $P = \text{diag}(c)$ and d is a constant that can be neglected

QUBO to Ising Model Conversion

- **Ising Model:**
  - Alternative formulation used by quantum annealers
  - Variables $s_i \in \{-1, 1\}$ instead of binary $x_i \in \{0, 1\}$
  - Optimization problem:

$$\min y = \sum_{i=1}^{N} h_i s_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} J_{ij} s_i s_j$$

- **Conversion Between Models:**
  - Relationship: $s_i = 2x_i - 1$
  - Simple transformation between QUBO and Ising formulations

QUBO Formulation for Portfolio Optimization

- **For equality constraint case:**

$$\min \left\{ \lambda_0 x^T \Sigma x + \lambda_1 \left(\sum_{i=1}^{N} x_i - n \right)^2 + \lambda_2 \left(\mu^T x - R^* \right)^2 \right\}$$

- **For inequality constraint case:**

$$\min \left\{ \lambda_0 x^T \Sigma x + \lambda_1 \left(\sum_{i=1}^{N} x_i - n \right)^2 + \lambda_2 \left(\mu^T x - R^* - \sum_{k=1}^{K} 2^k y_k \right)^2 \right\}$$

where:

- $K = \lfloor \log_2 \left(\sum_{i=1}^{N} \mu_i \right) \rfloor$
- y_k are slack variables $(k = 1, \ldots, K)$

Implementation & Results

Determining QUBO Parameters

Finding optimal penalty coefficients:

- Set $\lambda_0 = 1$ (scaling factor)
- λ_1 controls budget constraint penalty
- λ_2 controls return constraint penalty

Rule of thumb:

- Gain from violating constraint must be lower than cost
- Values too low: constraints violated
- Values too high: search efficiency decreased

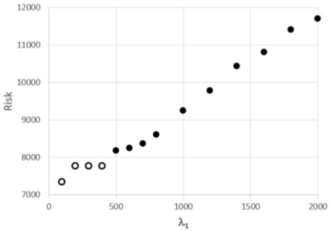

Figure: Relation between Risk and setting of λ. Open dots are violating the original constraints, closed dots are valid solutions.

Determining QUBO Parameters (continued)

- **Calculation for λ_1:**

$$\lambda_1^c = \max_i \sum_{j=1}^{n} \sigma_{i\{j\}}$$

  where $\sigma_{i\{j\}}$ is the j-th smallest covariance value for asset i

- **Calculation for λ_2:**
  1. $A_1 =$ average difference between smallest n sums $S_i = \sum_{j=1}^{n} \sigma_{i\{j\}}$
  2. $A_2 =$ average positive difference in μ_i between these n stocks
  3. $\lambda_2^c = A_1/A_2$
- **Note:** These values are first estimations and starting points for eventual grid search of optimal parameters

Solving QUBO on D-Wave Hardware

- **Problem Embedding:**
  - QUBO mapped to Pegasus topology of D-Wave Advantage
  - Minor Embedding problem is NP-Hard
  - D-Wave's system handles embedding automatically
- **Chain Strength (γ):**
  - Variables must be duplicated across multiple qubits
  - Connected qubits form "chains" representing same variable
  - Optimal when $\gamma \geq \sum_{i,j} |Q_{ij}|$
  - Finding minimal γ is NP-Hard
- **D-Wave Hybrid Solvers:**
  - Combine classical algorithms with quantum processing
  - Allocate QPU to problem parts where it benefits most
  - Can accommodate very large problems
  - Multiple parallel solvers including Tabu Search

Experiment Setup

- **Problem:** Portfolio optimization with risk minimization under budget and return constraints
- **Indices:** Nikkei225 and S&P500
- **Approach:** D-Wave quantum annealer vs. classical methods
- **Classical benchmarks:**
  - Simulated Annealing (SA)
  - Genetic Algorithm (GA)
  - Gurobi (GB)
  - LocalSolver (LS)
- **Parameters:** Various combinations of stocks pool size (N), selection size (n), and minimum return (R^*)

Results: Solution Quality

Instance Size	HQPU	SA	GA	GB	LS
Small ($N \leq 100$)	Optimal	Optimal	Near-opt	Optimal	Optimal
Medium ($100 < N \leq 200$)	Within 5%	Within 5%	35-80% gap	Memory issues	Best
Large ($N > 200$)	Within 18%	Within 22%	Poor	N/A	Best

Table: Solution quality comparison across methods

- **Key findings:**
  - LocalSolver consistently found best solutions
  - HQPU and SA performed comparably well
  - GA underperformed significantly
  - Gurobi ran out of memory for $N > 150$

Results: Computational Performance

Instance	HQPU	SA	GA	GB	LS
Nikkei (N=50)	$\approx$ 1s	2-8s	53-61s	$<$ 1s	2s
Nikkei (N=225)	$\approx$ 1.2s	31-43s	131-169s	N/A	3-207s
S&P (N=500)	5.8s	135s	221s	N/A	16s

Table: Solver time comparison (seconds)

- **HQPU advantages:**
  - Very fast computation time ($\approx$ 1-6s)
  - Scales well with problem size
  - Time nearly independent of instance parameters
- **Note:** HQPU times shown are per query; actual implementation used 5 queries per instance plus parameter grid search

Conclusions & Insights

Strengths of D-Wave HQPU:

- Fast solution time
- Minimal scaling with problem size
- Near-optimal solutions for smaller instances
- Reasonable solutions for larger instances

Limitations & Considerations:

- Solution quality gap for largest instances
- Optimality not proven (unlike Gurobi)
- Parameter tuning required (λ_1 and λ_2)
- Multiple runs needed due to stochastic nature

Best approach by use case:

- **Need for provable optimality:** Gurobi (for $N \leq 150$)
- **Large problems with time constraints:** HQPU
- **Best solution quality:** LocalSolver (though cannot prove optimality in reasonable time)

Replication of the Results

Replication of Results Overview

In this study, we aimed to replicate and extend the work of Phillipson and Bhatia on portfolio optimization using quantum annealing. The problem was formulated as a QUBO, minimizing risk under budget and return constraints, and solved using hybrid quantum-classical solvers. Due to limitations accessing D-Wave's platform, we performed a simulation to demonstrate the mechanism and compare it with classical methods.

- Quantum algorithm used: QAOA (Quantum Approximate Optimization Algorithm)
- Classical optimizer: COBYLA
- Hybrid quantum-classical approach for quantum sampling and optimization

Methodological Changes and Rationale

- **QAOA vs VQE**: We used QAOA for combinatorial optimization, contrasting it with the VQE approach used by Phillipson and Bhatia. The motivation was QAOA's potential advantages in handling optimization problems with combinatorial nature.
- **Optimization Depth**: We varied the number of iterations and the maximum number of iterations to understand the impact of optimization depth on solution quality.
- **Hybrid Approach**: Unlike the original study, which relied solely on D-Wave's quantum annealer, we incorporated a hybrid quantum-classical solver for better solution flexibility.

Results and Differences from Original Study

- **QAOA Results**: The QAOA solver found optimal solutions for smaller instances, but slightly underperformed in return while matching classical methods in risk minimization.
- **Comparison with Original Work**: Results were comparable in terms of feasible solutions but QAOA didn't surpass classical solvers (e.g., Gurobi) in return maximization.
- **Performance vs Classical Solvers**: LocalSolver and Gurobi performed better in return maximization but quantum methods like QAOA showed potential scalability for larger problems.
- **Key Differences**: QAOA required more careful tuning of parameters compared to D-Wave's quantum annealer, highlighting the sensitivity of quantum optimization methods to settings like chain strength and annealing schedules.

Conclusion

Conclusion Overview

The application of quantum computing to portfolio optimization demonstrates both promise and challenges. Our study aimed to replicate and extend previous work on quantum annealing for portfolio optimization, focusing on hybrid quantum-classical approaches.

- Quantum computing shows potential for portfolio optimization, especially for large-scale problems.
- However, classical solvers outperform quantum approaches in terms of solution quality, particularly for large instances.
- Quantum algorithms like QAOA need further refinement to meet return constraints consistently.

Challenges with Quantum Approaches

- **Stochastic Nature**: The quantum algorithms' probabilistic nature necessitates multiple runs, leading to variability in results.
- **Parameter Tuning**: The performance of quantum algorithms, such as QAOA, is highly sensitive to parameter tuning, requiring careful adjustments.
- **Scalability Issues**: While quantum methods show promise for scalability, they struggle with return maximization, especially in larger problems compared to classical solvers.

Future Directions and Hybrid Approaches

- **Hybrid Quantum-Classical Approaches**: These approaches have potential for certain use cases where time constraints are critical and exact optimality isn't required.
- **Improvement in Quantum Algorithms**: Future development is necessary to refine quantum algorithms, particularly in handling constraints and achieving reliable return maximization.
- **Classical Solvers Remain Strong**: For smaller problems with high return requirements, classical solvers like Gurobi remain the more effective solution.

In conclusion, while quantum computing is promising for financial optimization, its practical application requires further refinement and development.

The End