Quantum Thinking in Finance: A Toy Model of Risk and Superposition.

Valdir Dumba
Ennis Mawa

Abstract: In this paper, we will be discussing how a Hybrid Quantum/Classical algorithm enhanced with Game theory would impact the Finance world, particularly in the Portfolio Optimization Sector. For that purpose, we provide an overview of the current state of the Portfolio Optimization Sector, and an example of a Hybrid Classical/Quantum QAOA optimization algorithm capable of generating Higher Annual Expected returns in comparison to current classical models, while also taking into consideration factors such as the covariance between the assets in the portfolio, and the behavior of other investors of Investment funds using Game theory, and then compare it to current Classical Strategies. We also discuss the current limitations and problems that modern Quantum computers endure, and what impact a Quantum computer with less noise and more qubits may have in Industries such as Mutual funds, Hedge Funds, ETF, and more.

Contents

1. Introduction 2

2. Methodology 4
- *2.1 Overview 4*
- *2.2 Mean-Variance Portfolio Mathematical Overview........ 4*
- *2.3 QUBO to QAOA translation 5*
- *2.4 Modified 2 qubits approach 6*
- *2.5 Modified 2 qubits Mean Variance Equation 7*
- *2.6 Mixed Strategy Nash Equilibrium for Portfolio Optimization 8*
- *2.7 Application of Mixed Strategy 9*
- *2.8 Hybrid Quantum/Classical Algorithm Structure 10*

3. Results 11
- *3.1 Covariance Matrix Table 12*
- *3.2 Mixed strategy asset probabilities Tables 13*
- *3.3 Investment Table 14*
- *3.4 Optimal Result 16*
- *3.5 Portfolio Growth Graph 18*
- *3.6 Future Prospects 20*

References 21

Appendix 22–26

1. Introduction

With the rise of the maritime trades in the 1700s, a common problem that arose between traders was the loss of profit that many Sailors and Traders had over their Ships due to external factors such as extreme weather or even the infamous pirates. With that in mind, many traders started to share their profits with investors, giving shares of their own trading profits in exchange for accountability for the losses if a ship didn't come back. With time, this process was more formalized until the creation of the First modern Stock exchange in 1602,Amsterdam, trading shares of the Dutch Indian Company. With more companies joining the Stock exchange, modern concepts such as diversification, "an investing strategy in which the investor spreads investments across different types of asset classes in order to reduce the risk of loss" (NerdWallet, 2024), and risk minimization started to evolve as many investors diversified their portfolios in order to minimize losses if a ship did not make it through, and the first forms of Portfolio Optimization, " the process of selecting and combining different assets with the aim to achieve the best possible outcome in terms of risk and return."(Financial Edge, 2024), started to appear. At the beginning of the 20th century, the first forms of Portfolio Optimization started to take form, but investment was still based on identifying good companies and accurately timing when to invest, something that would become formalized with the advent of Modern Portfolio Theory by Harry Markowitz, which mathematically defined the relationship between risk and return (Pierre 2025).

In the modern days while many of the contemporary stock exchanges don't depend on Ships not coming back, or accurately timing your investment, most of the essential concepts such as Diversification and Risk Minimization are still common, and an important part when deciding which asset to invest, and while modern Classical Portfolio Strategies have improved considerably in comparison to strategies used in the past, a completely new step is still in the merge of happening that could significantly influence most of the modern known portfolio management practices. If perfected such technology could not only change the way we look at Investing ,similar to the creation of the first Portfolio Theory's, but it would also not only affect individual investors but also larger investment Industries such as hedge funds and ETFs. Portfolio Industry. This new emerging technology is Quantum Computers (Symons et al. 2024).

However, before we dive into Quantum Computers, we must understand the scientific foundations behind their construction: Quantum Mechanics. Developed in the early 20th century by some of the most famous physicists, such as Planck, Einstein, Schrödinger, and Heisenberg. Quantum mechanics is the study of matter and energy at the atomic and subatomic level, where classical rules no longer apply, and we start seeing some unusual phenomena more noticeably the idea of Entanglement, which is the ability of atomic particle to correlate their state with other particles, and Superposition, where a particle can be in multiple states at once, until it is finally measured. These unusual properties give rise to effects that can be used in the everyday world, in some common modern technologies, like lasers, semiconductors,MRI machines ,etc. In the case of quantum computers, these principles enable quantum systems to explore many possible configurations simultaneously, laying the groundwork for quantum algorithms that can accelerate tasks such as search, simulation, and optimization far beyond classical limits (Hidary 2019; Microsoft Research 2017).

Since the Theorization of Quantum Computers in early 80s by Richard Feynman, to the modern Quantum Computers in IBM, and Google, Quantum Computers have dragged the attention of the scientific community for decades with its enormous potential in almost every single sector, with its applications ranging from materials science to complex optimization problems.(Hidary 2019; Microsoft Research 2017) Moreover, although it is not intended to replace our current classical computers, Quantum Computers have the ability to simulate all possible Scenarios, using the previously stated Quantum processes such as Superposition, and Entanglement, thereby achieving results that would take sometimes even decades in the most advanced Classical Supercomputers of today enabling the simulation from the simplest things such as the best route to a restaurant, to overly complex structure such as long-chained proteins, and amino acid as shown by multiple modern institutions such as IBM and Google that are actively developing quantum systems capable of tackling optimization, logistics, and financial modeling at unprecedented scales (IBM Quantum 2025).

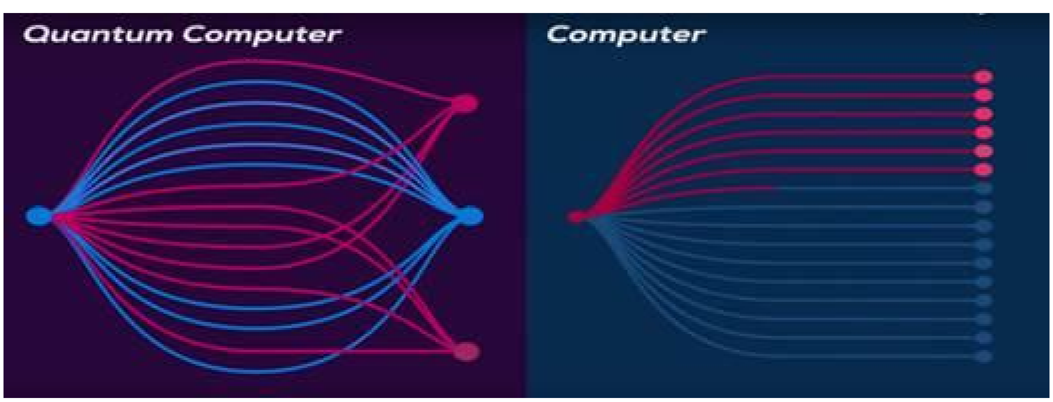


Figure 2: **Comparison between the Solution route of a Quantum, and a Classical Computer.**

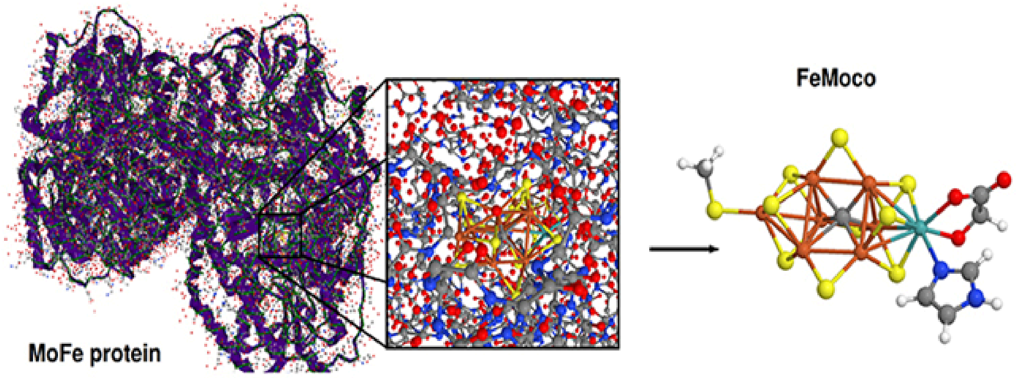


Figure 3: The MoFe protein, left, and the FeMoco, right, would be able to be analyzed by quantum computing to help reveal the complex chemical system behind nitrogen fixation by enzyme nitrogens.

While the full extent of potential is still being researched and discovered today, the outcomes and possibilities that such technology would have in the Portfolio Optimization sector are evident. Quantum computers would not only enable the simulation of complex economic models (Qiskit Community 2025) using a huge portion of data, but they will also help to reduce risks that come with investing by simulating Portfolios over certain periods of time, not only serving as an improvement to current strategies, but surpassing them with its edge over classical strategies Additionally, with the work of other modern fields such as Game theory, a field known for anticipating others' decisions, something crucial in the Financial world, a Hybrid Quantum/Classical enhanced using Game theory Portfolio Optimization algorithm would improve the way we currently look at this field by having the potential to improve portfolio optimization. As a way to prove and show the possibilities that a Perfectly stable Theoretical Quantum Computer may have, In this paper we will be looking at theoretical at the Hybrid Quantum Classical Computer enhanced with Game theory, that actively shows the full potential of such technology. Moreover, the development of a low-noise, fully operational quantum computer could transform fields such as cybersecurity and biochemistry., but the Financial sectors in specific the Portfolio Optimization Sector as well.

2. Methodology

2.1 Overview

For the purpose of this paper, we will be solving a multi-investment level portfolio optimization problem by encoding a mean-variance portfolio optimization function into QUBO(Quadratic Unconstrained Binary Optimization) and solving our problem with QAOA, a Hybrid Quantum/Classical Algorithm. Additionally in order to include more strategic interactions, and adaptability to our Algorithm we will be adding a mixed strategy Nash Equilibrium to our Algorithm that makes decisions not only based in the given data by Finance database's such as Yahoo Finance, but other investors or investment funds by randomly selecting from a set of actions according to a probability distribution of different investment strategies, and the growth of the asset on a selected period.

2.2 Mean-Variance Portfolio Mathematical Overview

We will be initially looking at the base mean variance function. A QUBO (Quadratic Unconstrained Binary Optimization) 1 qubit-based function that decides whether you should invest or not in an asset out of the number of assets selected with 1s being invest and 0s don't invest, by taking into consideration factors such as the covariance between the assets, and the Risks that you may face by investing on them represented by the subtraction of the risk factor to the expected return.

$$\min_{x \in \{0,1\}^n} q x^T \Sigma x - \mu^T x. \qquad \text{eq.1}$$

Subject to: $1^T x = B \qquad eq. 2$

$x \in \{0,1\}^n$: Denotes the vector of Binary Decision variables, which indicates which asset to pick(x[i]= 1), and which not to pick (x[i]= 0)

μ^T x: Defines the Expected Return

q>0: Controls the risk appetite of the decision maker

B: B denotes the budget, i.e., the number of assets to select out of n

μ∈R^n: defines the expected returns for the assets,

Σ∈R^(n×n): specify the covariances between the assets,

2.3 QUBO to QAOA translation

However, in order to solve this optimization problem on a quantum device, our main equation must be converted from QUBO to an Ising Hamiltonian that can be used as the cost operator in QAOA.

For that purpose, we must change our binary decision variables x_i∈{1,0} to a spin variable z_i∈ {+1,-1} via the transformation:

$$x_i = \frac{1 - z_i}{2} \qquad \text{eq.3}$$

And then add it to our main QUBO function reformulated into a Hamiltonian following the previously said mode:

Example: Original QUBO form Hamiltonian conversion

$$\min_{x\in\{0,1\}^n} \left(x^T Q x\right) \quad \text{eq. 4}$$

We substitute the eq.3 into the equation 4

$$\min_{z\in\{-1,1\}^n} \left(z^T Q z + b^T z\right) \quad \text{eq.5}$$

Proceed to convert the eq.5 into a Hamiltonian

$$H_G = \sum_{i,j} Q_{ij} Z_i Z_j + \sum_i b_i Z_i \quad \text{eq.6}$$

Modified Version:

$$\min_{x\in\{0,1\}^n} \left(q\, x^T \Sigma x - \mu^T x\right) \quad \text{eq. 7}$$

$$\min_{z\in\{-1,1\}^n} \left(q\, z^T \Sigma z + \mu^T z\right). \quad \text{eq.8}$$

$$\min_{z\in\{-1,1\}^n} \left(\sum_i h_i z_i + \sum_{i<j} J_{ij} z_i z_j\right) + \text{Constant}. \quad \text{eq. 9}$$

Where h_i, and J_(i,j) are coefficients determined by the expected return vector (μ), the Covariance Matrix(∑), and the risk factor (q). The Ising form can then be converted into a Hamiltonian:

$$\hat{H}_C = \sum_i h_i Z_i + \sum_{i<j} J_{ij} Z_i Z_j \quad \text{eq.10}$$

And then with Z_i denoting the Pauli–Z operator acting on qubit i. This Hamiltonian is used in the cost unitary of QAOA, being finally applicable for use in a quantum device:

$$U_C(\gamma) = e^{-i\gamma \hat{H}_C} \quad \text{eq.11}$$

2.4 Modified Two-Qubits Approach

One major limitation of this approach was its simplistic decision model: invest or not invest. In reality, investors and firms never operate in such simplistic binary terms; rather, they operate in different degrees of investment depending on facts such as trends, crises, or recessions. Moreover, limiting the number of choices to two reduces the number of available strategies that investors may have in cases of sudden changes in the market, where some assets may require bigger, medium, or even smaller investments than others. So, taking that into consideration, one of the solutions found was adding an extra qubit, which granted four different investment levels while keeping the simplicity of the algorithm.

The 4 strategies come in the way shown in Table 1:

Bit Strings	Investment Levels	Reward Weights
0.0	Don't Invest	0
1.0	Invest Low	1
0.1	Invest Medium	2
1.1	Invest High	3

Table 1: 4 investment Strategies Table, in which each two-qubit String is given a reward and a readable caption so it can easily be manipulated and interpreted in the algorithm.

With 4 investment levels, and a compatible rewards system, we can model a broader algorithm following the same principles as before, and at the same time solving our previous problem of simplistic decisions. Additionally, for simplistic purposes, each investment level is assigned to a certain percentage of the Initial Investment or Income, with the following structure:

IV= Initial Investment

Don't Invest = 0%

Invest Low = 6% or IV/16

Invest Medium= 12,5% or IV/8

Invest High= 25% or IV/4

2.5 Modified Two-qubits Mean Variance Equation

With the addition of an extra qubit, our mean variance equations, we must undergo some changes in order to be fit for use in the algorithm, mainly changes in the vector notation, and changes in both Expected returns and Covariance Matrices in the following way:

$$\min_x \left(q \sum_{i,j=1}^{n} \Sigma_{ij} (x_{i,0} + 2x_{i,1})(x_{j,0} + 2x_{j,1}) - \sum_{i=1}^{n} \mu_i (x_{i,0} + 2x_{i,1}) \right)$$ eq. 11

Subject to: $\sum_{i=1}^{n} (x_{i,0} + 2x_{i,1}) \leq B.$ eq.12

x_i0,x_i1∈ {1,0}: The 2 qubits used to encode the investment

μ_i: arranged expected return

∑_(i,j): Arranged Covariance Matrix

q: Risk aversion Constant

m/n: Number of Assets chosen.

and with some simplifications, our final equation becomes:

$$\min_w \left(q \sum_{i=1}^{m} \sum_{j=1}^{m} \Sigma_{ij} w_i w_j - \sum_{i=1}^{m} \mu_i w_i \right)$$ eq. 13

"Subject to: $\sum_{i=1}^{m} w_i \leq B.$ eq.14

$$w_i = x_{i,0} + 2x_{i,1}$$

$$w_j = x_{j,0} + 2x_{j,1}$$

With this change, not only is the function suitable for a 2 qubits algorithm, but it is also well-suited for the rewards system adopted later on in the algorithm. For example, when investing in two assets, a High Investment, and the other, Don't Invest:

$$\sum_{j=1}^{m}(x_{j,0} + 2x_{j,1})$$. eq. 15

$$\sum_{j=1}^{2}(x_{j,0} + 2x_{j,1}) = (0 + 2 \cdot 0) + (1 + 2 \cdot 1) = 3$$. eq. 16

We end up with a reward of 3 incentives to the algorithm from these assets, and only one must undergo a high investment while the other is the complete opposite.

2.6 Mixed Nash Equilibrium Portfolio Optimization Strategy.

When choosing whether to invest in an asset or not, investors and firms not only consider visible data, and company policies that may indicate future growth but also take into consideration other factors that may not be visible to the human eye but still remain crucial for future profits. These factors are other investors, and firms' opinions, and future decisions that may cause an asset to go down or up, and in the interconnected and competitive world of investment, where one party's profit can be someone's loss, anticipating other investors can help to reduce risks and potential losses. In these situations, Game theory comes in. Game theory, a subject known for its analyses in strategic interactions, and decisions, in games or situations in which the outcomes depend on multiple participants', serves as one of the best options to solve such complex problems, a Game theory strategy known as mixed-strategy equilibrium is used for this sole purpose.

Mixed strategies are strategies that involve players making decisions not by choosing a single fixed action or a pure strategy, but by randomly selecting from a set of actions according to a certain probability distribution. Some of the most famous examples are rock paper scissors, where each play must be chosen randomly with a probability of 1/3, or the penalty kick game, where goalkeepers must choose whether they dive on the left, right, or remain, in the center depending on other external factors such as the penalty taker's handiness. In the context of Investment, investors find themselves in similar situations on a daily basis where they even unconsciously add probabilities, and randomize their strategies, depending on how they believe other investors may perceive certain assets, and extending it further into Portfolio Optimization, while not as commonly used in Traditional Portfolio Optimization mixed strategies can, and indeed provide an additional layer of critical thinking, and adaptability to the to the overall strategy impacting the algorithm, thus adding a deeper layer of other players insight beyond the data that the algorithm uses. Therefore, by incorporating a strategic interaction model when facing uncertainty and anticipating the actions of other market participants, the strategy fills the gaps by enhancing the algorithm's critical thinking beyond data.

2.7 Application of Mixed Strategies

To operationalize the mixed strategy described above, we model the strategic interaction between two players an Investor(Player A), and other investor or Hedge fund (Player B) both under the same conditions, with our game being defined by two states, the Up state (when the asset on that specific day or period trends positively), and Down (When the asset on the same conditions trends negatively) as in both commonly occasions the strategies may vary adding the enough adaptability necessary. Additionally, each player has access to the four previously stated investment levels: Don't Invest, Low, Medium, and High. These levels are then presented in two payoff matrices (Up, and Down) that encode the expected outcomes for each pair of strategies.

Up Matrix

A/B	Don't Invest	Invest Low	Invest Medium	Invest High
Don't Invest	(0.9,0.9)	(0.7, 0.6)	(0.5, 0.5)	(0.4, 0.4)
Invest Low	(0.8, 0.7)	(1.0, 1.1)	(0.9, 0.8)	(0.7, 0.7)
Invest Medium	(0.6, 0.6)	(1.2, 1.2)	(1.4, 1.5)	(1.3, 1.3)
Invest High	(0.5, 0.5)	(1.0, 1.0)	(1.3,1.3)	(1.6,1.6)

Table 2: Payoff matrix for an upward market trend (Up state). Rows represent Player A's investment strategy and columns represent Player B's strategy. Each cell shows the expected payoff pair (A, B) for each strategy combination.

Down Matrix

A/B	Don't Invest	Invest Low	Invest Medium	Invest High
Don't Invest	(1.0,1.0)	(0.6, 0.6)	(0.4, 0.4)	(0.3, 0.3)
Invest Low	(0.6, 0.6)	(0.9, 0.9)	(0.6, 0.6)	(0.5, 0.5)
Invest Medium	(0.5, 0.5)	(0.6, 0.6)	(0.8, 0.8)	(0.7,0.7)

Invest High	(0.3, 0.3)	(0.5, 0.5)	(0.6,0.6)	(0.8,0.8)

Table 3: Payoff matrix for a downward market trend(Down state). Rows represent Player A's investment strategy and columns represent Player B's strategy. Each cell shows the expected payoff pair (A, B) for each strategy combination.

The algorithm then proceeds to choose which strategy to use depending on the trend for that asset and applies a mixed strategy Nash equilibrium on the time chosen. Additionally, both matrices were designed to accommodate evenly pay off distributions across each row, accompanied by some noise to grant a more even probability strategy across all investment levels. With the necessary foundation in both the mathematical and game-theoretic formulations, we can now proceed to examine the structure of the Hybrid Quantum/Classical Algorithm.

2.8 Hybrid Quantum/Classical Algorithm

Having established the necessary Mathematical formulation for the portfolio optimization problem, as well as the necessary game theory mixed strategy equilibrium strategies, we now proceed to describe the Hybrid Quantum/Classical Algorithm that ties everything together.

The way the algorithm works is by dividing its workflow into nine steps:

-Imports

-Market Data Acquisition

-Statistical data preparation

-Covariance Matrix

-Mixed Strategy Set-up

-Quadratic Program set-up

-Rewards System, and QUBO Formulation

-QAOA Formulation, and Execution

-Result Interpretation

1. Imports

The algorithm is set up in a hybrid set of classical and quantum computing libraries In particular the following 5:

- Qiskit: Provides the necessary Quantum algorithm frameworks, and optimization, including other math formulations such as the Mean-variance Function formulation into QUBO, Ising Hamiltonian Conversion, and the QAOA algorithm formulation

- Nashpy : For the Implementation of the game-theoretic component of the model, enabling the computation of mixed-strategy Nash equilibriums.

- Panda: It is primarily used as a tool to facilitate data handling and processing of the data acquired from Yfinance

-Yfinance: It serves as the primary source of financial data collection, enabling a direct retrieval of stock prices from the Yahoo Finance database

-Matplotlib: Used for visualization of the results, such as the covariance matrix of assets, the Portfolio growth graph, and the Investment strategy table.

-NumPy: Serves as a way to support all the numerical computation, data and matrix operations required through the Optimization process.

2. Market data Acquisition

In order to build a realistic portfolio optimization problem, the algorithm is set up for a time period of 5 years, while other functions responsible for the data collection, and combining stocks using the pandas, and NumPy Libraries are created. Additionally, 15 stocks from ten different industries are added to the database, but only four are randomly selected by a random generator keeping simplicity, while granting a higher asset diversification across the results. Therefore, through a randomized asset selection we are able to grant more credibility and adaptability to the algorithm, showing how the results can be sustained regardless of the asset picks.

3,and 4. Statistical data preparation, and the Covariance Matrix

The data collected, and arranged before is then adapted to the once stated constants used in the mean -variance function with emphasis in the Expected return(μ), and Covariance Matrix($\sum$). Then a covariance Table of the four randomly selected stocks is set up enabling us to see the correlation between the stocks selected granting greater variety, while reducing the risks if one sector does not grow. As a result, through this process we are able to convert real life random data collections into forms that can actually be used in a Quantum Computer, as similarly to Classical Computers that work through a series of 1s and 0s (bits), in order to be operational their input variable need to be adjusted.

5. *Mixed strategy set-up*

In order to incorporate more strategic interactions and market dynamics into the optimization model, the previously stated mixed strategy was incorporated into the algorithm with the addition of small elements such as noise, and a process to smooth the results causing the probabilities to appear more evenly distributed, using the Nashpy library, adding another layer of thinking through game theory beyond what Quantum computers are able to provide.

6. Quadratic program set-up

To integrate the previously defined QUBO function, the algorithm first incorporates a quadratic program, followed by the incorporation of the 2-qubit system, with constraints to ensure that only one level is selected.

7. Reward System, and QUBO formulation

To align with the Mixed Strategies, and the two qubits formulation rewards system responsible to select the weight or value, is then adjusted to incorporate both elements, using both the two qubits weight (0,1,2,3) system, and the mixed strategy distributions as other support to the assets' weight and adjust the expected returns for each asset beyond data, granting the adaptability necessary to the algorithm.

The previous QUBO formulation elements, such as the Risk term, the Expected Return, and the Covariance Matrix, are then converted into Python and once more converted into an Ising Hamiltonian in the next step.

8. QAOA Formulation

The previous Quadratic program is then mapped into a QUBO formulation, and subsequently it is converted into an Ising Hamiltonian that preserves all the portfolio cost functions. Then a Hadamard gate is applied to all qubits, giving a state of superposition, and in doing so, representing all possible portfolio combinations simultaneously. As a Hybrid algorithm, an initial classical optimizer, COBYLA (Constrained Optimization by Linear Approximations), is first initialized and configured using a maximum of 250 iterations, allowing its adjustment into the changing parameters of a QAOA circuit. QAOA is then adjusted for three repetitions or layers , with the purpose of controlling the quantum circuit and balancing the trade-off between the solutions and computational costs. Moreover, a Quantum Sampler evaluates the circuit outcomes and provides the probability distributions used for Optimization. In order to execute the algorithm, the QAOA program is wrapped in the Minimum Eigen Optimizer, a high-level interface from Qiskit, enabling the execution of the algorithm on a perfect Simulated without noise Quantum Computer. To conclude, a call "result = qaoa.solve(qp)" is set up at the end of the algorithm, granting the execution of the Hybrid Quantum/Classical Algorithm loop and ensuring the solution that most approximates the minimum eigenvalue corresponds to the optimal investment strategy.

9. Result Interpretation

With all the previous data and Quantum/Classical formulations, the algorithm is then structured to tune all this information gathered, allowing a greater interpretation of the data and results, thus granting a clear way to compare the results with other strategies, such as Classical and one-qubit ones. For that purpose, five strategies are implemented:

Covariance Matrix Table: A 4 by 4 Covariance matrix table, showing the covariance between the randomly selected assets by the algorithm. With that information, the investor and the algorithm will have a higher awareness of the correlation between the assets invested, thus granting a greater diversification of the Portfolio, while reducing the risks if a certain sector trends negatively.

Mixed Strategy Investment strategy Tables: The previously established Mixed strategy developed by the algorithm can now be viewed by the investor running the algorithm, showing the probabilities of each of the four stated investment strategies in the following way:

Asset N (XXX):

Investor probabilities:

Don't Invest: 0,23

Invest Low: 0.25

Invest Medium: 0.29

Invest High: 0.23

With that not only can we assure the accuracy of the mixed strategy taken by the algorithm, but we can actually see how each probability is mapped, thus giving a better insight into which strategies to invest besides the usual data-based review given by the algorithm.

Optimal Investment Strategies tables: A 5x5 Matrix table with all the investment strategies taken by the algorithm for each of the assets selected.

Optimal Result Strategy Table: A report of all the possible strategies sampled by the QAOA algorithm, with their values, and probabilities serving as a way to show how the algorithm actually samples its final strategy in the Optimal Investment Strategy Table.

Performance Simulator: In order to more accurately compare the strategy taken by the algorithm with other classical and one-qubit-based a performance simulators, a graph showing the growth of the randomly selected portfolio over a period of 2 years is provided.

10. Final Result

The full Python implementation, including data preprocessing, QAOA solver, Nash equilibrium, and performance evaluation, is available in the following GitHub repository: https://github.com/valdirdumba/Hybrid-Quantum-Classical-Portfolio-Optimization-Algorithm-Enhanced-with-Game-theory.git

3.Results

3.1 Covariance Matrix Table

The covariance Matrix of the four randomly selected assets (NVDA,TSLA,WMT,NFLX) in Figure 5,highlights the correlation between the assets chosen by the algorithm. For instance, the assets combinations such as Tesla, and Nvidia present a darker color showing their correlation in the Technology field, but other assets chosen such as WMT from the Retail industry or Netflix from the Media Streaming Industry present a lighter color in comparison, highlighting the algorithm's stronger role in diversifying the Portfolio. Therefore, demonstrating the algorithm's incorporation of Covariance between its randomly selected choices, adding the diversification needed to reduce concentration risk, a feature central in mean variance optimization.

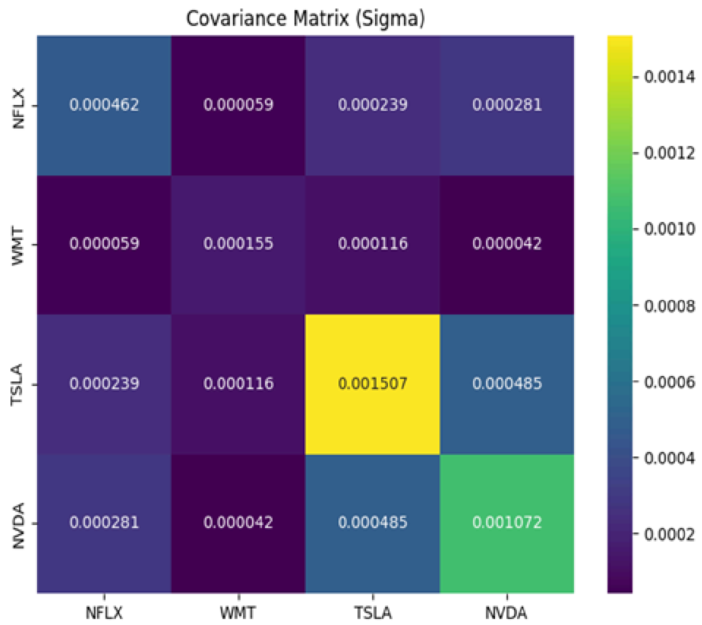


Figure 4: Covariance Matrix Table

Furthermore, by looking at other randomly selected Covariance matrices generated by running the algorithm multiple times the same arguments sustain, as we can see the algorithm's prevalence in selecting the most diverse assets persists, thus granting a lower risk with some other examples such as :

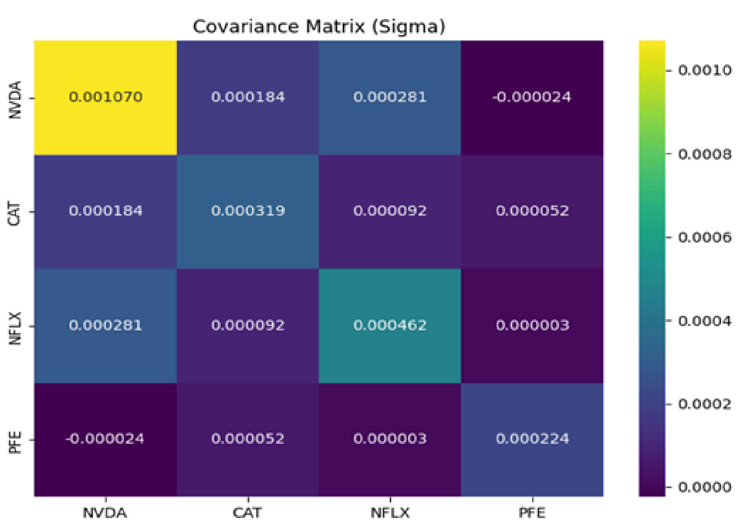


Figure 5: Covariance Matrix Table with 4 different assets (Pfizer, Netflix, CAT, NVDA)

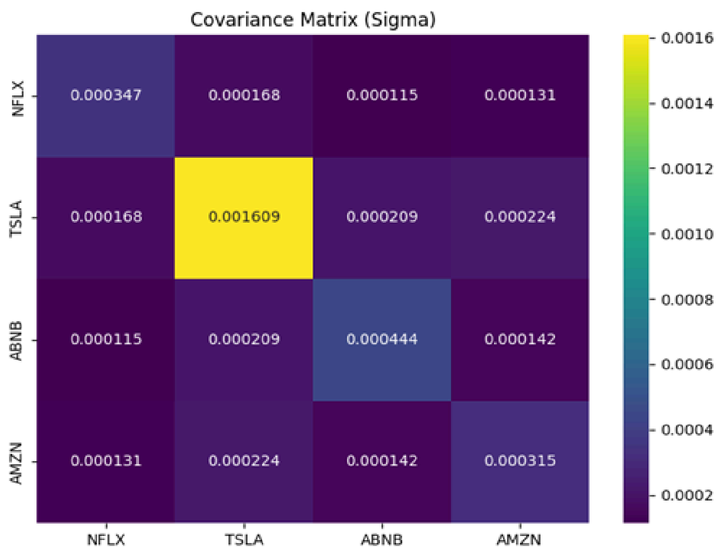


Figure 6: Covariance Matrix Table with 4 different assets(Amazon, Air BNB, Tesla, NFLX)

Where some assets can range from the Semiconductor and Microchip industry to other non-related industries such as the Pharmaceuticals and Construction industries.

3.2 Mixed Strategies Assets Probabilities Tables

By incorporating the game-theoretical strategy into the algorithm, the algorithm generates four strategy investment probability distributions for each randomly selected combination of assets. For instance, with the selected strategies:

_Asset 1 (NFLX):

Investor probabilities:

Don't Invest: 0.23

Invest Low: 0.25

Invest Medium: 0.28

Invest High: 0.24

Asset 2 (WMT):

Investor probabilities:

Don't Invest: 0.24

Invest Low: 0.24

Invest Medium: 0.30

Invest High: 0.21

Asset 3 (TSLA):

Investor probabilities:

Don't Invest: 0.23

Invest Low: 0.25

Invest Medium: 0.29

Invest High: 0.23

Asset 4 (NVDA):

Investor probabilities:

Don't Invest: 0.24

Invest Low: 0.26

Invest Medium: 0.28

Invest High: 0.22

These distributions are more evenly spread compared to a classical mean-variance optimizer, which on average, is known for allocating most of its resources in a few selected strategies. Moreover, with the even distribution probabilities between the 4 randomly selected assets the results reflect the adaptive nature of mixed strategies. For example, by looking at the leanness probability towards certain strategies such as higher leanness to Don’t Invest in Comparison to High Invest the investor may change their strategies towards certain actions that sometimes can be opposite to what the Quantum algorithm proposes due to data misconceptions that are flagged by game theory strategies thereby hedging against market uncertainty by balancing multiple possible outcomes rather than committing to a single deterministic allocation.

3.3 Investment Table

The algorithm translates the randomly selected strategies into a proper, clear 5x3 Investment Matrix Table, that can be understood by anyone without prior knowledge of Quantum Computing. In other words, converting the 1s, and 0s generated by the algorithm into clear, understandable ways that can be interpreted by anyone. Additionally, each randomly selected asset is given an Investment Strategy based on the QAOA algorithm set-up, and the Mixed strategy. For example, Nvidia is assigned a Medium Investment strategy while others may receive a High Investment, or Low Investment. This results in a diversified portfolio in which each asset is distributed across multiple Investment levels rather than concentrated in a single strategy.

Optimal Investment Strategy

Asset	Investment Level	Investment Amount ($)
NFLX	Invest Medium	$12,500.00
WMT	Invest Medium	$12,500.00
TSLA	Invest Medium	$12,500.00
NVDA	Invest Medium	$12,500.00
Total	—	$50,000.00

Figure 7: Optimal Investment Two qubits-based Strategy Table, in which every asset is given an investment strategy based on its predictable performance using the Algorithm

Optimal Investment Strategy

Asset	Investment Level	Investment Amount ($)
NVDA	Invest Medium	$12,500.00
RACE	Invest Medium	$12,500.00
PFE	Don't Invest	$0.00
WMT	Invest Medium	$12,500.00
Total	—	$37,500.00

-

Figure 8: Optimal Investment Two qubits-based Strategy Table, in which every asset is given an investment strategy based on its predictable performance using the Hybrid Quantum/Classical Algorithm.

Furthermore, in contrast to 1 qubit-based strategies, even though it may grant a higher return due to the greater money Investment adding an extra qubit, not only grants a higher pull of Investment strategies as shown on the previous Optimal Investment Strategies, but a higher expected return on Investment, and a higher Sharper Ration in the Long run due to the proportional increase in Risk that a 1 qubit Strategy provides, something we will be looking for in other sections in specific when showing the results of the Portfolio Growth graph.

3.4 Portfolio Growth graph

The Hybrid Quantum/Classical Game theory algorithm's effectiveness over a simulated period of 5 years has shown impressive results, being able to generate a higher average Annual Expected Return than most known Classical strategies, and funds such as Nasdaq and SMP 500. For instance, in a period of 5 years both Nasdaq, and SMP 500 experienced a Total Annual return of over 100%, more specifically 118,8% for Nasdaq, and 109% for SMP 500, as shown in both figures 11, and 12 taken from (Backtest curvo..)

Figure 10: Nasdaq, and SMP 500 growth over a period of 5 years

Average annualised return

Index	Last year	Last 5 years
Nasdaq-100	20.5%	17.0%
S&P 500	16.3%	15.9%

Total return

Index	Last year	Last 5 years
Nasdaq-100	20.5%	118.8%
S&P 500	16.3%	109.0%

Figure 11: Total return of Investment of both Nasdaq, and SMP 500 over a period of 1, and 5 years

But despite smaller count of assets, and a smaller frame of qubits, the algorithm was capable of surpassing both Investment Funds with a higher Annual Expected return on average in most randomized asset selections with some examples being:

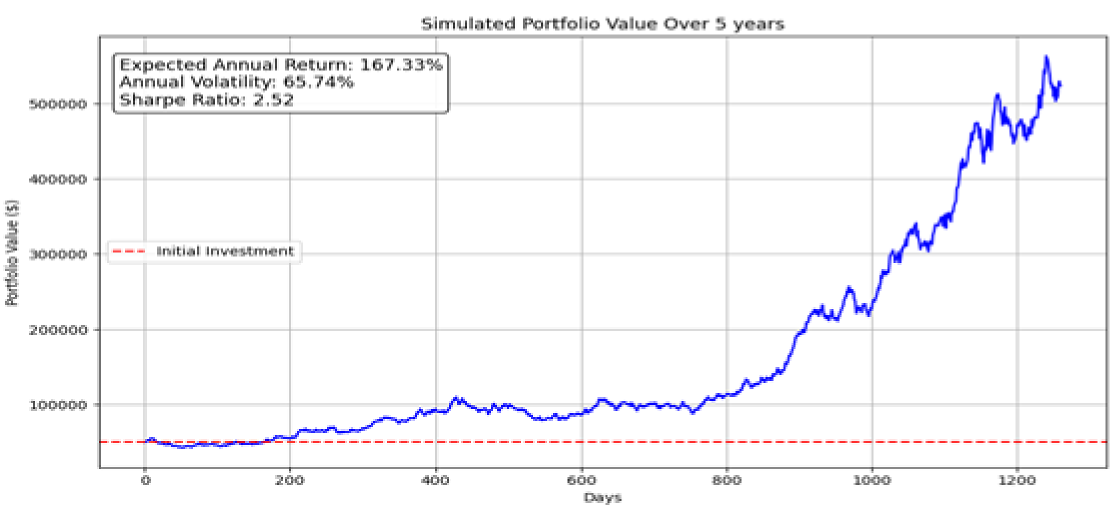
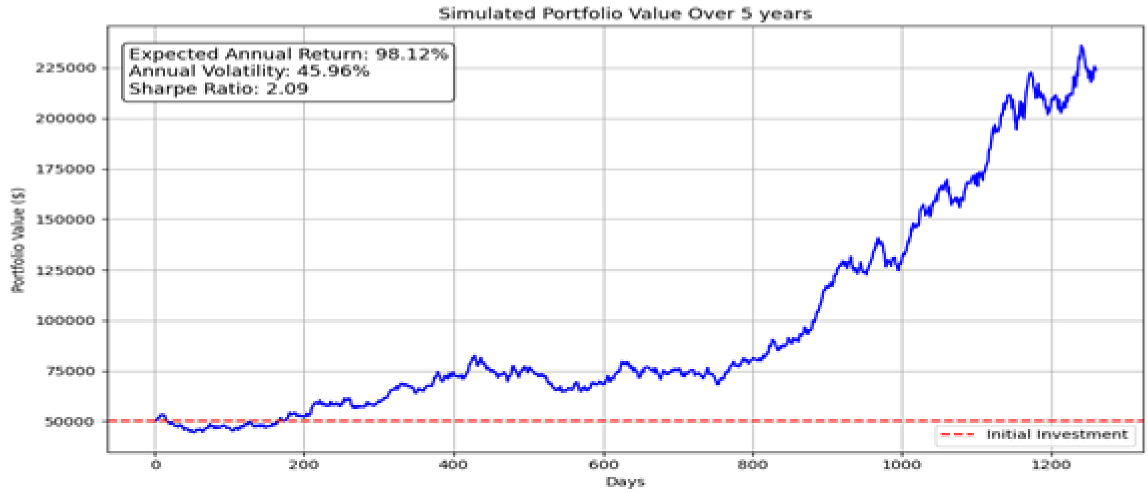


Figure 12: 2-qubits Hybrid Algorithm growth over a period of 5 years



Despite the smaller Expected return in comparison to the other previously stated strategies, the Sharpe Ratio remains above 2 showing the strategies efficiency in reducing the overall unenviable risk while also granting a high Annual Expected return that surpasses even the most advanced Portfolio Optimization Technologies of the Modern days.

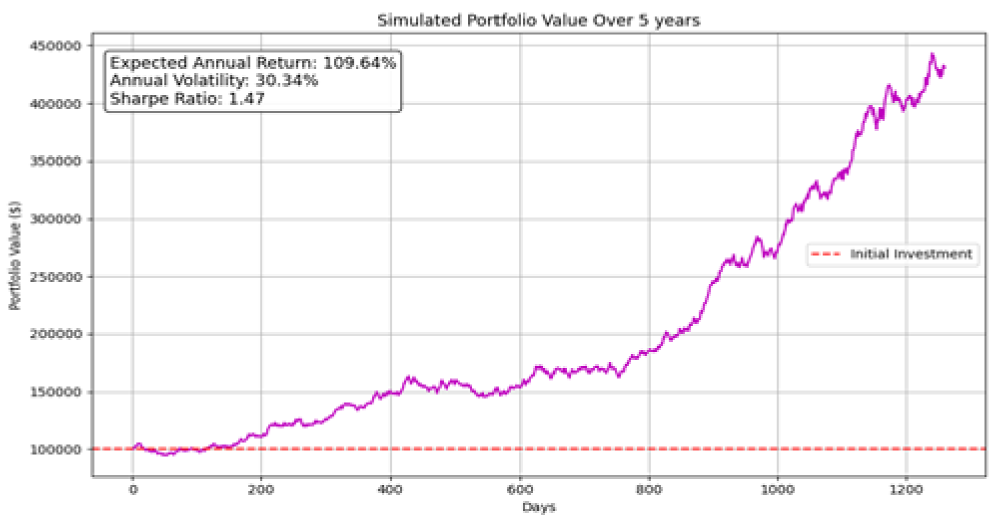


Figure 14: 1 qubit Hybrid Algorithm growth over a period of 2 years

Additionally, while the 1-qubit strategies may have had a higher expected return on some occasions due to its simplistic investment strategy it tends to output a lower average Sharpe Ratio across multiple randomize tests such as in Figure 15, showing its increased risk upon a much simpler approach, once more consolidating the correlation between the addition of qubits, and the Algorithm's efficiency.

3.6 Future Prospects

The future of quantum computers is undoubtedly promising, and intriguing. With advancements in quantum hardware with the creation of Quantum computers with less noise, quantum software programming languages, and quantum algorithms, quantum computers are set to change the way we approach problems across most of the known modern industries,

and as seen in this paper despite the small number of asset choices, and qubits, the algorithm was still able to output results comparable or better than modern fund benchmarks such as the NASDAQ-100 and S&P 500 (Curvo 2025; Marques and Witman 2025) where, despite those same firms achieving an annual return of 118% and 109% over a period of 5 years (Curvo 2025; Marques and Witman 2025) where, this algorithm was able to output similar or even higher results than those firms in a significant amount of randomize choices, despite its limited number of assets and qubits. Suggesting that, in a near future an enhanced less noisy with hundreds or even thousands of qubits, with a higher number of assets would be able to change output Annual Expected Returns never even seen in modern history changing the Portfolio Optimization world as we know and eventually causing a revolution across the Finance world similar to other key events in investment history such as The Modern Portfolio Theory. Therefore, any future research in regard to the Quantum Finance world should consider Hybrid Quantum-Classical frameworks, comparative risk modeling, and real-time market responsiveness as despite, this papers researching limitations combined with the modern Quantum Computing limitations,

References

Wang, Haoshu. *Advantages and Applications of Quantum Game Theory.* REU paper, University of Chicago, Summer 2022. https://math.uchicago.edu/~may/REU2022/REUPapers/Wang,Haoshu.pdf.

Qiskit Community. *Portfolio Optimization*. *Qiskit Finance*. Accessed October 14, 2025. https://qiskit-community.github.io/qiskit-finance/tutorials/01_portfolio_optimization.html.

Symons, B. C. B., S. Tornow, V. Valls, S. Woerner, M. L. Wolf-Bauwens, J. Yard, S. Yarkoni, D. Zechiel, S. Zhuk, and C. Zoufal. *Challenges and Opportunities in Quantum Optimization*. *Nature Reviews Physics* 6 (2024): 718–735. https://doi.org/10.1038/s42254-024-00770-9.

Hidary, Jack D. *Quantum Computing: An Applied Approach*. Cham: Springer, 2019. https://doi.org/10.1007/978-3-030-23922-0.

Ceroni, Jack. *Intro to QAOA*. *PennyLane Demos*, November 18, 2020. https://pennylane.ai/qml/demos/tutorial_qaoa_intro/.

IBM Quantum. *Quantum Approximate Optimization Algorithm*. *IBM Quantum Platform*. Accessed October 14, 2025. https://quantum.cloud.ibm.com/docs/en/tutorials/quantum-approximate-optimization-algorithm.

Curvo. *Nasdaq-100 vs. S&P 500: Annual Returns Comparison*. Accessed October 14, 2025. https://curvo.eu/backtest/en/compare-indexes/nasdaq-100-vs-sp-500?currency=usd.

Marques, Rodrigo, and Shoshanna Witman. *Nasdaq 100 or S&P 500? The Better Long-Term Investment Might Surprise You*. *Investopedia*, August 22, 2025. https://www.investopedia.com/nasdaq-100-or-s-and-p-500-the-better-long-term-investment-might-surprise-you-11796258.

Xu, Tongyang, and Ming Zhong. *Analysis of the Quantum Approximate Optimization Algorithm*. *Heliyon* (2018). https://doi.org/10.1016/j.heliyon.2018.e00770.

Pierre, Samuel. *An Introduction to Portfolio Optimization in Python*. *Built In*, April 3, 2025. https://builtin.com/data-science/portfolio-optimization-python.

Author(s). *Title of the Paper*. In *Proceedings of the Conference Name*, pages xx–xx. IEEE. Accessed October 14, 2025. https://ieeexplore.ieee.org/document/9222275.
(Replace with full citation details once identified.)

Microsoft Research. *What Problems Will We Solve with a Quantum Computer? Microsoft Research Blog*, July 5, 2017. https://www.microsoft.com/en-us/research/blog/problems-will-solve-quantum-computer/.

Jawade, Bhavesh. *Quantum Computing ?/! Towards Data Science*, July 8, 2018. https://towardsdatascience.com/quantum-computing-5b715976e61d/.

NerdWallet. *What Is Diversification? How It Works, Benefits*. *NerdWallet*. Accessed October 14, 2025. https://www.nerdwallet.com/investing/learn/diversification.

Appendix

Optimal Result Table

The QAOA solver outputs a set of sampled strategies, each represented by a bitstring, with corresponding values and probabilities. Outputting all the possible selections considered by the algorithm in its original 1s, and 0s form, showing all the probabilities and values each possible strategies had, while displaying the Optimal Selection Strategies (The previous strategy outputted on the Investment Table) that grants the highest Expected Return in the long-run, actively showing the Multitask testing of all possible results, capacity of Quantum computers.

_---------------- Optimal Result ----------------

Optimal: selection [0. 1. 0. 1. 0. 1. 0. 1.], value 0.0213

--- Full result ---

selection	value	probability

[0. 1. 0. 1. 0. 1. 0. 1.]	0.0213	0.0196
[0. 1. 1. 0. 0. 1. 0. 1.]	0.0200	0.0198
[0. 1. 0. 0. 0. 1. 0. 1.]	0.0189	0.0120
[1. 0. 0. 1. 0. 1. 0. 1.]	0.0183	0.0200
[1. 0. 1. 0. 0. 1. 0. 1.]	0.0171	0.0202
[0. 1. 0. 1. 1. 0. 0. 1.]	0.0162	0.0203
[1. 0. 0. 0. 0. 1. 0. 1.]	0.0160	0.0122
[0. 1. 0. 1. 0. 1. 1. 0.]	0.0159	0.0203
[0. 0. 0. 1. 0. 1. 0. 1.]	0.0157	0.0125
[0. 1. 1. 0. 1. 0. 0. 1.]	0.0150	0.0205
[0. 1. 1. 0. 0. 1. 1. 0.]	0.0147	0.0205

[0. 0. 1. 0. 0. 1. 0. 1.]	0.0146	0.0126
[0. 1. 0. 0. 1. 0. 0. 1.]	0.0140	0.0124
[0. 1. 0. 0. 0. 1. 1. 0.]	0.0137	0.0124
[0. 0. 0. 0. 0. 1. 0. 1.]	0.0136	0.0076
[1. 0. 0. 1. 1. 0. 0. 1.]	0.0134	0.0207
[1. 0. 0. 1. 0. 1. 1. 0.]	0.0132	0.0207
[0. 1. 0. 1. 0. 0. 0. 1.]	0.0126	0.0131
[1. 0. 1. 0. 1. 0. 0. 1.]	0.0123	0.0209
[1. 0. 1. 0. 0. 1. 1. 0.]	0.0121	0.0209
[0. 1. 0. 1. 0. 1. 0. 0.]	0.0117	0.0132
[0. 1. 1. 0. 0. 0. 0. 1.]	0.0115	0.0132
[1. 0. 0. 0. 1. 0. 0. 1.]	0.0114	0.0127
[0. 1. 0. 1. 1. 0. 1. 0.]	0.0113	0.0210
[0. 0. 0. 1. 1. 0. 0. 1.]	0.0111	0.0129
[1. 0. 0. 0. 0. 1. 1. 0.]	0.0111	0.0127
[0. 0. 0. 1. 0. 1. 1. 0.]	0.0110	0.0129
[0. 1. 0. 0. 0. 0. 0. 1.]	0.0107	0.0080
[0. 1. 1. 0. 0. 1. 0. 0.]	0.0105	0.0133
[0. 1. 1. 0. 1. 0. 1. 0.]	0.0102	0.0212
[0. 0. 1. 0. 1. 0. 0. 1.]	0.0101	0.0130
[1. 0. 0. 1. 0. 0. 0. 1.]	0.0100	0.0134
[0. 0. 1. 0. 0. 1. 1. 0.]	0.0099	0.0130
[0. 1. 0. 0. 0. 1. 0. 0.]	0.0095	0.0081
[0. 1. 0. 0. 1. 0. 1. 0.]	0.0093	0.0129
[1. 0. 0. 1. 0. 1. 0. 0.]	0.0093	0.0134

[0. 0. 0. 0. 1. 0. 0. 1.]	0.0092	0.0079
[1. 0. 1. 0. 0. 0. 0. 1.]	0.0091	0.0135
[0. 0. 0. 0. 0. 1. 1. 0.]	0.0089	0.0079
[1. 0. 0. 1. 1. 0. 1. 0.]	0.0088	0.0215
[1. 0. 0. 0. 0. 0. 0. 1.]	0.0083	0.0082
[0. 1. 0. 1. 0. 0. 1. 0.]	0.0082	0.0136
[1. 0. 1. 0. 0. 1. 0. 0.]	0.0082	0.0136
[0. 0. 0. 1. 0. 0. 0. 1.]	0.0080	0.0084
[1. 0. 1. 0. 1. 0. 1. 0.]	0.0078	0.0216
[0. 1. 0. 1. 1. 0. 0. 0.]	0.0076	0.0136
[0. 0. 0. 1. 0. 1. 0. 0.]	0.0073	0.0084
[0. 1. 1. 0. 0. 0. 1. 0.]	0.0072	0.0137
[1. 0. 0. 0. 0. 1. 0. 0.]	0.0072	0.0082
[0. 0. 1. 0. 0. 0. 0. 1.]	0.0071	0.0084
[1. 0. 0. 0. 1. 0. 1. 0.]	0.0069	0.0131
[0. 0. 0. 1. 1. 0. 1. 0.]	0.0068	0.0134
[0. 1. 1. 0. 1. 0. 0. 0.]	0.0065	0.0137
[0. 1. 0. 0. 0. 0. 1. 0.]	0.0064	0.0083
[0. 0. 0. 0. 0. 0. 0. 1.]	0.0063	0.0051
[0. 0. 1. 0. 0. 1. 0. 0.]	0.0062	0.0085
[1. 0. 0. 1. 0. 0. 1. 0.]	0.0060	0.0139
[0. 0. 1. 0. 1. 0. 1. 0.]	0.0059	0.0135
[0. 1. 0. 0. 1. 0. 0. 0.]	0.0056	0.0083
[1. 0. 0. 1. 1. 0. 0. 0.]	0.0054	0.0139
[0. 0. 0. 0. 0. 1. 0. 0.]	0.0053	0.0051

[1. 0. 1. 0. 0. 0. 1. 0.]	0.0051	0.0140
[0. 0. 0. 0. 1. 0. 1. 0.]	0.0050	0.0082
[0. 1. 0. 1. 0. 0. 0. 0.]	0.0049	0.0088
[1. 0. 1. 0. 1. 0. 0. 0.]	0.0044	0.0140
[1. 0. 0. 0. 0. 0. 1. 0.]	0.0043	0.0085
[0. 0. 0. 1. 0. 0. 1. 0.]	0.0042	0.0087
[0. 1. 1. 0. 0. 0. 0. 0.]	0.0040	0.0089
[0. 0. 0. 1. 1. 0. 0. 0.]	0.0036	0.0087
[1. 0. 0. 0. 1. 0. 0. 0.]	0.0035	0.0085
[0. 0. 1. 0. 0. 0. 1. 0.]	0.0033	0.0087
[0. 1. 0. 0. 0. 0. 0. 0.]	0.0032	0.0054
[1. 0. 0. 1. 0. 0. 0. 0.]	0.0030	0.0090
[0. 0. 1. 0. 1. 0. 0. 0.]	0.0027	0.0087
[0. 0. 0. 0. 0. 0. 1. 0.]	0.0026	0.0053
[1. 0. 1. 0. 0. 0. 0. 0.]	0.0021	0.0091
[0. 0. 0. 0. 1. 0. 0. 0.]	0.0019	0.0053
[0. 0. 0. 1. 0. 0. 0. 0.]	0.0015	0.0056
[1. 0. 0. 0. 0. 0. 0. 0.]	0.0014	0.0055
[0. 0. 1. 0. 0. 0. 0. 0.]	0.0007	0.0057
[0. 0. 0. 0. 0. 0. 0. 0.]	0.0000	0.0034
[0. 1. 0. 1. 1. 1. 1. 1.]	0.0347	0.0000
[0. 1. 1. 0. 1. 1. 1. 1.]	0.0333	0.0000
[0. 1. 0. 0. 1. 1. 1. 1.]	0.0321	0.0000
[1. 1. 0. 1. 0. 1. 1. 1.]	0.0314	0.0000
[1. 0. 0. 1. 1. 1. 1. 1.]	0.0312	0.0000

[1. 1. 1. 0. 0. 1. 1. 1.]	0.0300	0.0000
[1. 0. 1. 0. 1. 1. 1. 1.]	0.0299	0.0000
[1. 0. 0. 0. 1. 1. 1. 1.]	0.0287	0.0000
[0. 0. 0. 1. 1. 1. 1. 1.]	0.0282	0.0000
[0. 1. 0. 1. 1. 1. 0. 1.]	0.0279	0.0000
[0. 1. 0. 1. 0. 1. 1. 1.]	0.0276	0.0000
[0. 0. 1. 0. 1. 1. 1. 1.]	0.0269	0.0000
[0. 1. 1. 0. 1. 1. 0. 1.]	0.0265	0.0000
[0. 1. 1. 0. 0. 1. 1. 1.]	0.0263	0.0000
[0. 0. 0. 0. 1. 1. 1. 1.]	0.0257	0.0000
[1. 1. 0. 1. 1. 0. 1. 1.]	0.0256	0.0000
[0. 1. 0. 0. 1. 1. 0. 1.]	0.0253	0.0000
[0. 1. 0. 0. 0. 1. 1. 1.]	0.0252	0.0000
[1. 1. 0. 1. 0. 1. 0. 1.]	0.0247	0.0000
[1. 0. 0. 1. 1. 1. 0. 1.]	0.0246	0.0000
[1. 0. 0. 1. 0. 1. 1. 1.]	0.0244	0.0000
[1. 1. 1. 0. 1. 0. 1. 1.]	0.0243	0.0000
[1. 0. 1. 0. 1. 1. 0. 1.]	0.0233	0.0000
[1. 0. 1. 0. 0. 1. 1. 1.]	0.0231	0.0000
[0. 1. 1. 1. 0. 1. 0. 1.]	0.0227	0.0000
[1. 0. 0. 0. 1. 1. 0. 1.]	0.0222	0.0000
[0. 1. 0. 1. 1. 1. 1. 0.]	0.0221	0.0000
[0. 1. 0. 1. 1. 0. 1. 1.]	0.0221	0.0000
[1. 0. 0. 0. 0. 1. 1. 1.]	0.0220	0.0000
[0. 0. 0. 1. 1. 1. 0. 1.]	0.0219	0.0000

[0. 0. 0. 1. 0. 1. 1. 1.]	0.0215	0.0000
[0. 1. 1. 0. 1. 0. 1. 1.]	0.0209	0.0000
[0. 1. 1. 0. 1. 1. 1. 0.]	0.0207	0.0000
[0. 0. 1. 0. 1. 1. 0. 1.]	0.0206	0.0000
[0. 0. 1. 0. 0. 1. 1. 1.]	0.0204	0.0000
[0. 1. 0. 0. 1. 0. 1. 1.]	0.0199	0.0000
[1. 0. 1. 1. 0. 1. 0. 1.]	0.0196	0.0000
[0. 1. 0. 0. 1. 1. 1. 0.]	0.0196	0.0000
[0. 0. 0. 0. 0. 1. 1. 1.]	0.0193	0.0000
[1. 0. 0. 1. 1. 1. 1. 0.]	0.0191	0.0000
[1. 0. 0. 1. 1. 0. 1. 1.]	0.0190	0.0000
[0. 1. 0. 1. 0. 0. 1. 1.]	0.0180	0.0000
[1. 0. 1. 0. 1. 0. 1. 1.]	0.0179	0.0000
[1. 0. 1. 0. 1. 1. 1. 0.]	0.0178	0.0000
[0. 1. 1. 1. 1. 0. 0. 1.]	0.0174	0.0000
[0. 1. 0. 1. 1. 1. 0. 0.]	0.0173	0.0000
[0. 1. 1. 1. 0. 1. 1. 0.]	0.0173	0.0000
[0. 0. 1. 1. 0. 1. 0. 1.]	0.0170	0.0000
[1. 0. 0. 0. 1. 0. 1. 1.]	0.0169	0.0000
[0. 1. 1. 0. 0. 0. 1. 1.]	0.0169	0.0000
[1. 0. 0. 0. 1. 1. 1. 0.]	0.0167	0.0000
[0. 0. 0. 1. 1. 1. 1. 0.]	0.0166	0.0000
[0. 0. 0. 1. 1. 0. 1. 1.]	0.0164	0.0000
[0. 1. 1. 0. 1. 1. 0. 0.]	0.0161	0.0000
[0. 1. 0. 0. 0. 0. 1. 1.]	0.0160	0.0000

[0. 0. 1. 0. 1. 1. 1. 0.]	0.0154	0.0000
[0. 0. 1. 0. 1. 0. 1. 1.]	0.0154	0.0000
[1. 0. 0. 1. 0. 0. 1. 1.]	0.0152	0.0000
[0. 1. 0. 0. 1. 1. 0. 0.]	0.0149	0.0000
[1. 0. 0. 1. 1. 1. 0. 0.]	0.0147	0.0000
[1. 0. 1. 1. 1. 0. 0. 1.]	0.0146	0.0000
[1. 0. 1. 1. 0. 1. 1. 0.]	0.0145	0.0000
[0. 0. 0. 0. 1. 0. 1. 1.]	0.0145	0.0000
[0. 0. 0. 0. 1. 1. 1. 0.]	0.0144	0.0000
[1. 0. 1. 0. 0. 0. 1. 1.]	0.0142	0.0000
[0. 1. 1. 1. 0. 0. 0. 1.]	0.0137	0.0000
[1. 0. 1. 0. 1. 1. 0. 0.]	0.0134	0.0000
[1. 0. 0. 0. 0. 0. 1. 1.]	0.0133	0.0000
[0. 1. 1. 1. 0. 1. 0. 0.]	0.0130	0.0000
[0. 0. 0. 1. 0. 0. 1. 1.]	0.0128	0.0000
[0. 1. 1. 1. 1. 0. 1. 0.]	0.0126	0.0000
[0. 0. 0. 1. 1. 1. 0. 0.]	0.0124	0.0000
[1. 0. 0. 0. 1. 1. 0. 0.]	0.0124	0.0000
[0. 0. 1. 1. 1. 0. 0. 1.]	0.0123	0.0000
[0. 0. 1. 1. 0. 1. 1. 0.]	0.0122	0.0000
[0. 0. 1. 0. 0. 0. 1. 1.]	0.0119	0.0000
[0. 0. 1. 0. 1. 1. 0. 0.]	0.0113	0.0000
[1. 0. 1. 1. 0. 0. 0. 1.]	0.0112	0.0000
[0. 0. 0. 0. 0. 0. 1. 1.]	0.0111	0.0000
[1. 0. 1. 1. 0. 1. 0. 0.]	0.0105	0.0000

[0. 0. 0. 0. 1. 1. 0. 0.]	0.0103	0.0000
[1. 0. 1. 1. 1. 0. 1. 0.]	0.0100	0.0000
[0. 1. 1. 1. 0. 0. 1. 0.]	0.0094	0.0000
[0. 0. 1. 1. 0. 0. 0. 1.]	0.0090	0.0000
[0. 1. 1. 1. 1. 0. 0. 0.]	0.0088	0.0000
[0. 0. 1. 1. 0. 1. 0. 0.]	0.0085	0.0000
[0. 0. 1. 1. 1. 0. 1. 0.]	0.0080	0.0000
[1. 1. 0. 1. 0. 0. 0. 0.]	0.0074	0.0000
[1. 0. 1. 1. 0. 0. 1. 0.]	0.0070	0.0000
[1. 0. 1. 1. 1. 0. 0. 0.]	0.0065	0.0000
[1. 1. 1. 0. 0. 0. 0. 0.]	0.0064	0.0000
[0. 1. 1. 1. 0. 0. 0. 0.]	0.0060	0.0000
[0. 0. 1. 1. 0. 0. 1. 0.]	0.0052	0.0000
[0. 0. 1. 1. 1. 0. 0. 0.]	0.0047	0.0000
[1. 0. 1. 1. 0. 0. 0. 0.]	0.0040	0.0000
[0. 0. 1. 1. 0. 0. 0. 0.]	0.0025	0.0000

Example of optimal result table for 1 qubit approach:

---------------- Optimal Result ----------------

Optimal: selection [0. 1. 1. 1.], value 0.0018

--- Full result ---

selection value probability

[0. 1. 1. 1.]	0.0018	0.0639
[1. 1. 1. 1.]	0.0018	0.0639
[0. 1. 0. 1.]	0.0013	0.0632
[1. 1. 0. 1.]	0.0013	0.0631
[0. 0. 1. 1.]	0.0013	0.0631
[1. 0. 1. 1.]	0.0012	0.0630
[0. 1. 1. 0.]	0.0010	0.0627
[1. 1. 1. 0.]	0.0010	0.0627
[0. 0. 0. 1.]	0.0008	0.0623
[1. 0. 0. 1.]	0.0007	0.0623
[0. 1. 0. 0.]	0.0005	0.0620
[1. 1. 0. 0.]	0.0005	0.0619
[0. 0. 1. 0.]	0.0005	0.0619
[1. 0. 1. 0.]	0.0005	0.0618
[0. 0. 0. 0.]	0.0000	0.0612
[1. 0. 0. 0.]	-0.0000	0.0611