Quantum Finance - an Overview

C H Renumadhavi, Archit Srivastava, Ashutosh Kumar Singh and Anushka Mittal

July 13, 2021

Quantum Finance - An Overview

Dr. C.H. Renumadhavi*, Archit Srivastava*, Ashutosh Kumar Singh*, Anushka Mittal*
*Department of Electronics and Instrumentation Engineering, RV College of Engineering, Bengaluru, Karnataka, India

***Abstract*—With the advent of emerging technologies, it has become possible to have a road-map which tackles really difficult real world challenges. From the past decade, there has been significant development in the field of Quantum Computing. The advantages that it provides can take an industry decades ahead of where it stands right now. One of the fields with rigorous implementation of the same technology is, Financial Markets. Quantum Finance has been one of the most promising field of application and development with significantly improved results. The implementation of various algorithms such as Variational Quantum Eigensolver(VQE) and Quantum Approximate Optimization Algorithm(QAOA) to map and solve problem statements pertaining to Financial use cases have proven to be immensely beneficial for the industry. The paper intends to scratch the tip of the field and have an overlook on how the industry giants like Goldman Sachs, JP Morgan Chase & Co. and Consulting giants like McKinsey & Co. are working towards the development of solutions to tackle such problems. The paper tries to give an overview of the financial market history and specifically look into a Mean Variance Portfolio Optimization Problem using existing Cloud Quantum Computational Frameworks and analyze the results.**

I. Introduction

Stock market, well known as equity or share market, is an amalgamation of dynamic sellers and buyers of stocks and shares. The market includes securities that are numbered on stock exchange market as well as shares of private companies which are only traded privately through crowdfunding platforms based off of equity.[1] Investments in the stock market are made usually with an investment strategy and often done via stock brokerages and electronic trading platforms.[2] The price of these shares usually depends on the supply and demand. Stocks are categorised based on the geographical location of the company's domicile.

Being able to drive a countries Financial landscape, Stock Markets are one of the most integral part of everyone on planet. There are a lot of factors which drives a stock price on a daily basis thus influencing the portfolios of companies and Multi-Billionaires. There are various factors affecting the Financial Markets:

1) *Basic Factors:* These are the factors that are external in nature but are influential enough to change the behaviour of the Financial Market as whole. These are the factors that do not rise inside the Market but they are affected by various levels of economic activity as a whole.
2) *Technical Factors:* These are the factors that arise within the financial market and they are generated because of the decision made by an investor. There are more technical factors that affect the Financial Markets, but will be discussed in the upcoming chapters.

A. History of Stock Market

The place of giants or Stock Market has been a significant part of History. In the late 1400s, Antwerp or Belgium was the international trade center where traders bought goods hoping a rise in price and hence, profits. Following the advancements in ideologies and trading, in 1611, in Amsterdam, first modern stock trading was observed. In the late 1700s Buttonwood Tree Agreement was made by a group of merchants which ultimately forms the New York Stock Exchange(NYSE).[2] The formation of NYSE was a pivotal moment in the history of Financial Markets. In 1790, was formed the Philadelphia Stock Exchange which helped the development of finance sectors of the US immensely.

With the advent of 20th century we saw various events such as the early version of S&P 500 created by Henry Barnum Poor's company, the crash of the U.S. stock market, Trading on U.S. Stock exchange NASDAQ. These events helped shape today's economies, companies that we see today and helped a lot of Fortune 500 firms to learn and grow. In 2008, the crash of stock market was observed after the rise and fall of the housing market and mortgage-backed securities in the finance sectors proliferated.

Figure 1. NYSE in Early 1900's (Source: Wikipedia)

Entering the new decade, Markets were going to see a long haul of rise and fall. The World Trade Organization on April 08, 2020 predicted that the market may fall between the volumes of 13% to 32% in the upcoming months. This unrest in the market was caused due to the COVID-19 Pandemic.[3] It has become necessary now than it was ever before to predict and optimize the risks related to this sudden rise and fall of markets across globe. From decades there has been rigorous research and development in the area to improve the predictability of stocks by using classical computational methods, use of Machine Learning has been the tool of choice for many researchers.

Classical Computers have limitations which then reduces the span of problem statements to be tackled. The limitations and the rise of Quantum Computers in the last few decade has lead to massive shift in the way modern computation is moving forward. Practical development and implementation of Quantum Algorithms on Real Quantum Devices has lead to the start of a promising field which can shape how the world of Financial markets will develop.

B. Classical View

The classical view to financial markets majorly depends upon the use of various classical methods of computation to tackle different scenarios and use of these technologies in the field of Financial Market Analysis has exponentially increased the versatility of portfolios and reduced the risks related to them when compared to early stages of the markets.[4] Today a lot of companies try to solve these problems being faced on a daily basis and they make the use of programming to tackle them.

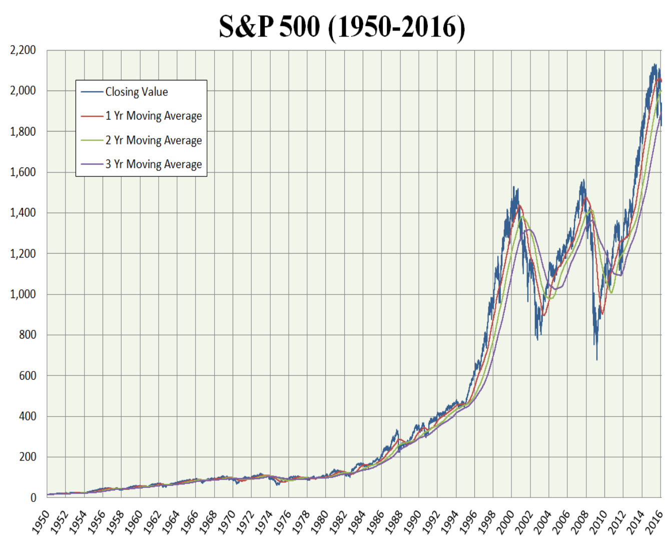

Figure 2. The S&P 500 stock trend from 1950-2016 (Source: Wikipedia)

Python one of the most popular programming languages has revolutionized the field of Stock Market Trading, by improving the ways in which current financial giants operate and have significantly changed their way of operation from their early days. There are majorly two ways for Stock Market Analysis:

1) *Fundamental Analysis:*This is a study which analyzes the current business and related trends and tries to predict the various effects that the portfolio might suffer.
2) *Technical Analysis:* This is a more scientific way of analyzing a portfolio and majorly it revolves around the use of statistics and graphical analysis of portfolios to derive some meaningful insight on the same.

But there are a lot of disadvantages when the size of parameters rise, more practical results are required, a faster time of operation is required from the technical end and a lot of other unaccounted difficulties rise due to the use of Traditional Computers. These can be tackled by using Quantum Computers.

C. Quantum View

Quantum Computing is more often considered as an exotic computational model that can break all encryption in the world, but it had a very humble beginning when it comes to the origin of this field.[5] The field of Quantum Computing has its origins from modern Quantum Mechanics, which was proposed by eminent physicists like Werner Heisenberg, Max Born, and Pascual Jordan in 1925. Quantum mechanics deals with the theory behind interactions of elementary particles with each other giving rise to mind-bending phenomena like superposition, entanglement, and interference.

The idea of quantum computing came into existence when physicist Paul Benioff put forth a quantum mechanical model of the famous Turing Machine. Since then, many quantum computational models have come into existence like the quantum circuit model, quantum Turing[16] machines, adiabatic quantum computers, one-way quantum computing, and various types of quantum cellular automata.[6] With time, all these mathematical computational models refined and developed the basis of modern quantum computational science that is quantum logic gates.

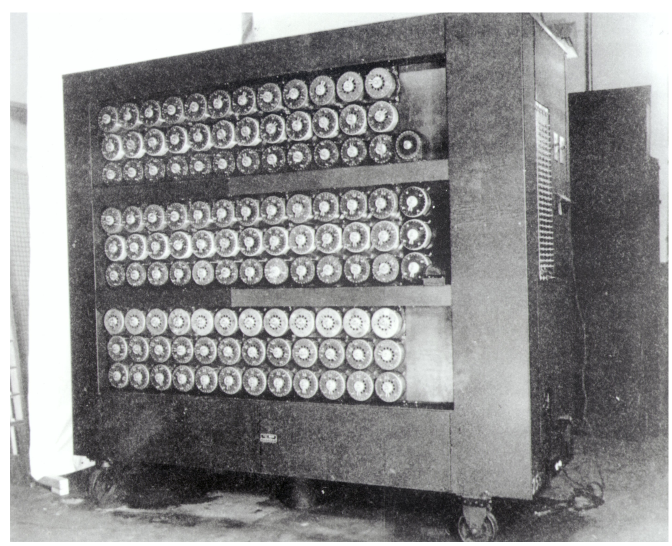

Figure 3. Alan Turing's Turing Machine (Source: Wikipedia)

Quantum logic gates helped one to discretize the parts of a quantum circuit to group similar operations together and also made it possible to define them as functional blocks rather than physical quantities. For several years it was thought that it is impossible to build a real-life quantum computer due to the technology and infrastructure needed to maintain and sustain its proper functioning.[17] Although the math was developed half a century ago, the technology was not ready to support such a sensitive device whose state can be altered even with the slightest of the interactions that it undergoes with its surrounding environment.

Hence, it was very difficult to achieve practical results with the technology in the '70s or '80s, when a major portion of the mathematics of quantum computation was being developed.[18] But, as technology progressed and classical systems become more robust and error-resistant, the possibility of developing precise control systems for actual quantum hardware increased, and the first experimental quantum computer was established using Nuclear Magnetic Resonance (NMR) in atoms and molecules at the Technical University of Munich, München, Germany. This feat opened new avenues for the development of quantum technologies.[19]

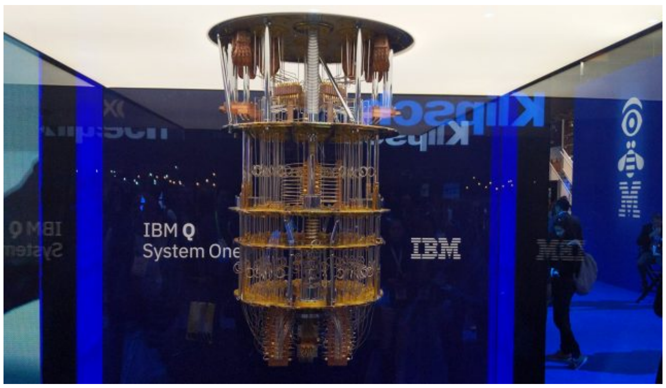

Figure 4. IBM Quantum Computer (Source: IBM Quantum)

Fast forward to today, most of the universities are working on various aspects of quantum computing. Not only academia but in the private sector, big companies like IBM, Google, Microsoft, Bosch, and even start-ups like Rigetti, DWave, IonQ, and many more are working on various types of technologies that have a great potential for developing the future quantum computers and quantum networking technologies.[7]

A few different types of designs include electron spin qubits, superconducting qubits, ionic qubits in RF/optical ion traps, nitrogen-vacancy center qubits, topological qubits with exotic Majorana fermions, and also photonic qumode systems.[20]

This paper tries to scratch the surface of the field of Quantum Finance leveraging the benefits of the Quantum Paradigm to solve real world scenarios in Stock Market Analysis.

II. Classical Approach

This section will try to give an in-depth overview to the classical approach that have been implemented to tackle financial uses cases. When talking about tools that help to determine the state of market and perform important analysis of the same, we arrive at the following:

1) *Moving Average Convergence Divergence(MACD):* These are specific type of indicators which signals after the appearance of Trading Condition.
2) *Accumulation and Distribution Line(A/D):* They are used to measure the volume of security over a range of trade. They are an essential part when it comes to Financial Analysis.
3) *Aroon Indicator:*The measurement of new highs and lows is done using Aroon Indicators. These specifically are used in price movement of market trends.
4) *Average Directional Index(ADI):* They are used to measure the momentum and the strength of the price moving trend. Any value above 40 corresponding to the ADI score indicates a High Trending and ADI below 20 indicates None Trending.
5) *On Balance Volume(OBV):* An indicator with measures of volume of security over time with positive and negative flows.
6) *Relative Strength Index(RSI):* These are a counter of MACD and signals before the trading condition appears.

These were some of the major tools which are used when it comes to stock market analysis. They are extensively used along with several algorithms. Looking into the major challenges that current companies are trying to tackle leads us to majorly three fields:

1) *Portfolio Optimization:* As the name suggests, these type of problem statements focus more on the ways to optimize the portfolio of an individual or a firm to minimize the losses incurred.
2) *Stock Price Prediction:* These are the problem statements that are dependent upon the prediction of trends in the stocks price specifically. These prediction thus helps the owner of stock, to make choices more efficiently and reduce the risk of incurring huge losses.
3) *Risk Analysis:* It is the analysis of measure of an investors willingness to take risk and invest in a particular asset. It is comprises of two components *Risk Capacity* and *Risk Aversion*.

In the paper we will be focusing more on the part of Portfolio Optimization.

A. Portfolio Optimization

This is an area which deals with increasing the profitability of the portfolio of owners and simultaneously reducing the risk incurred when trading is being made. Users like to have the portfolio which is clean, gives them high returns and involves less risk.[8] They want to have the selection of investment which fully fills their long term financial goals and are risk free at the same time.[9] Also The focus is on how to maximize the earnings.

B. Methods to achieve an Optimal Portfolio

1) *Equal Weight:* All components are assigned equal weights and this method has proven to be the most useful as there are no views when interested all assets are random.[10] This is the most basic form of optimization technique that is used across the industry.[11]
   Equal Weight Optimized portfolios are beneficial but still have a lot of drawbacks like:
   - Sudden changes in the portfolio can cause drastic change.
   - It is not suitable for sectors with High Catastrophic Loss Rates.

- Because of more volatility, they tend to fall more sharply in recession.

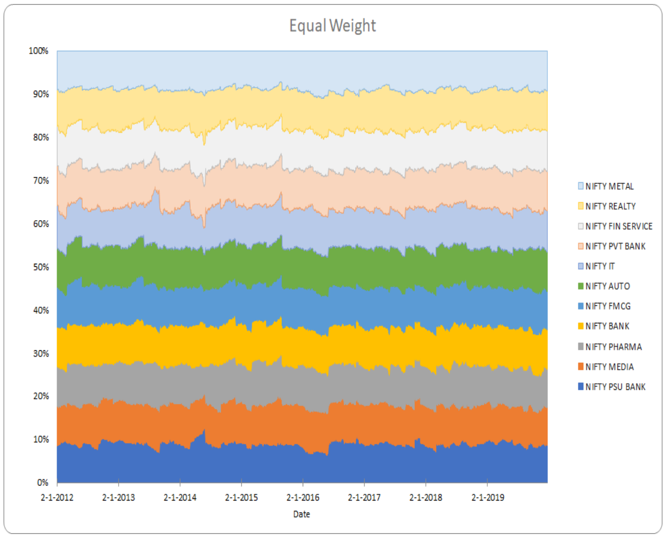

Figure 5. Sectoral Indicies of Equal Weighted Portfolio (Source: QuantInsti)

2) *Risk Parity:* This has the same function as Equal Weights but the weights of Risk Contribution of each asset is equal[12].
The functional formula for Risk Parity is as follows: The

$$w^{optimal} = arg\ min\ \frac{1}{2}\ w^T \Sigma w - \frac{1}{n} sum(ln(w_i))$$

sectoral indecies for Risk Parity looks like:

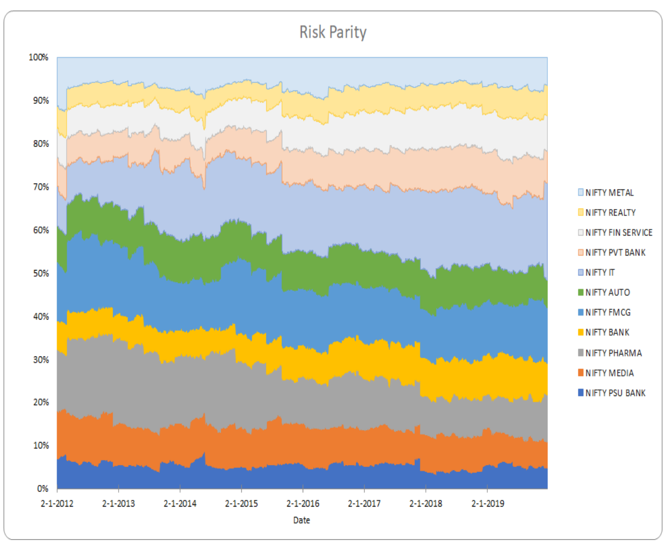

Figure 6. Sectoral Indicies of Risk Parity Portfolio (Source: QuantInsti)

3) *Minimum Variance Optimization:* This is an optimal algorithm when the volatility of the portfolio is to be reduced. This method works the best when the risks are not proportional to returns.[13]

$$w^{optimal} = arg\ min\ w^T \Sigma w$$

Formula for this Optimization method is:
The Sectoral Index representation of this method is representad as:

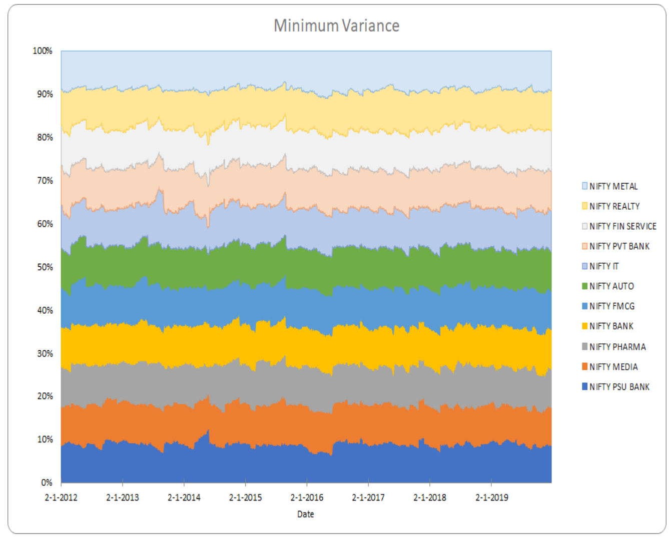

Figure 7. Sectoral Indicies of Minimum Variance Portfolio (Source: QuantInsti)

4) *Mean Variance Portfolio Optimization:* This is a type of portfolio optimization which provides maximum return for a given risk associated to the asset[14].
The formula for expected returns is[15]:

$$w^{optimal} = arg\ min\ w^T \Sigma w - qRw$$

The corresponding sectoral indecies for the following method is:

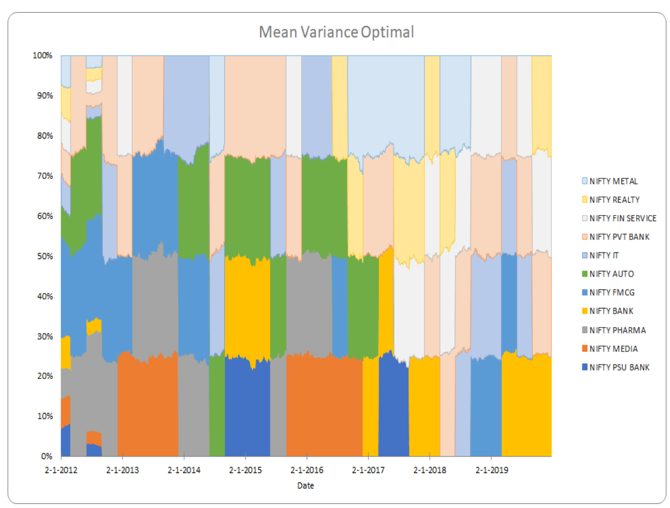

Figure 8. Sectoral Indicies of Mean Variance Portfolio (Source: QuantInsti)

III. Quantum Approach

In this section we will explicitly look into the specific case of Mean Variance Optimization using Quantum Computational Methods. We use Qiskit and Pennylane, python libraries from IBM and Xanadu to run the necessary code and get the desired result.[21]

A. Solution With Qiskit

The problem statement is to solve portfolio optimization of n assets: $\min_{x\in\{0,1\}^n} qx^T\Sigma x - \mu^T x$
$subject\ to: 1^T x = B$
The problem statement is mapped to a hamiltonian after the constraint is mapped to a penalty term $(1^T x - B)^2$. There are certain steps to be followed they are:

1) Defining The Problem Statement: Initially a Hamiltonian is created based on an Operator Instance and this generates an automatically mapped Paulis for Ising Hamiltonian and then is extended to the Time Series Data[22].

```

# set number of assets (= number of qubits)
num_assets = 4

# Generate expected return and covariance matrix from (random) time-series
stocks = [("TICKER%s" % i) for i in range(num_assets)]
data = RandomDataProvider(tickers=stocks,
                 start=datetime.datetime(2016,1,1),
                 end=datetime.datetime(2016,1,30))
data.run()
mu = data.get_period_return_mean_vector()
sigma = data.get_period_return_covariance_matrix()

# plot sigma
plt.imshow(sigma, interpolation='nearest')
plt.show()
```

The output of the following operations is as follows:

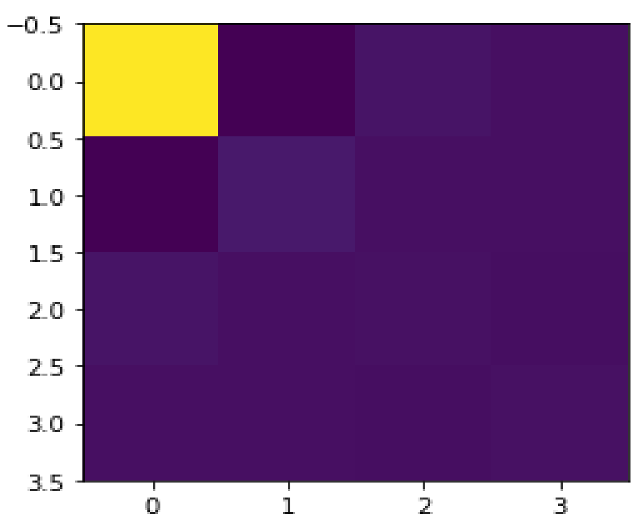

Figure 9. Sigma Plot of the Portfolio

2) Defining Utility Methods: For the output to look optimal, we define some methods.

```
def index_to_selection(i, num_assets):
    s = "{0:b}".format(i).rjust(num_assets)
    x = np.array([1 if s[i]=='1' else 0 for i in reversed(range(num_assets))])
    return x

def print_result(result):
    selection = sample_most_likely(result.eigenstate)
    value = portfolio.portfolio_value(selection, mu, sigma, q, budget, penalty)
    print('Optimal: selection {}, value {:.4f}'.format(selection, value))

    eigenvector = result.eigenstate if isinstance(result.eigenstate, np.ndarray) else result.eigenstate.to_matrix()
    probabilities = np.abs(eigenvector)**2
    i_sorted = reversed(np.argsort(probabilities))
    print('\n----------------- Full result ---------------------')
    print('selection\tvalue\t\tprobability')
    print('---------------------------------------------------')
    for i in i_sorted:
        x = index_to_selection(i, num_assets)
        value = portfolio.portfolio_value(x, mu, sigma, q, budget, penalty)
        probability = probabilities[i]
        print('%10s\t%.4f\t\t%.4f' %(x, value, probability))
```

3) Solution using VQE: We use a Variational Quantum Eigensolver(VQE) to achieve the most optimal portfolio state of the defined experiment.

```
backend = Aer.get_backend('statevector_simulator')
seed = 50

cobyla = COBYLA()
cobyla.set_options(maxiter=500)
ry = TwoLocal(qubitOp.num_qubits, 'ry', 'cz', reps=3, entanglement='full')
vqe = VQE(qubitOp, ry, cobyla)
vqe.random_seed = seed

quantum_instance = QuantumInstance(backend=backend, seed_simulator=seed, seed_transpiler=seed)

result = vqe.run(quantum_instance)

print_result(result)
```

B. Solution With Pennylane

The same code is tried to be implemented using pennylane and is analyzed for the results that it generates. The code included herewith is inspired by an article published by Ms. Rochisha Agarwal. [23]

There are various steps to achieve the results desired, and they are:

1) Quadratic Program Generation: This steps helps in the development of Hamiltonian and this thus forms the first step of any VQE algorithm.

```
#Calculate mean returns
mu = data.get_period_return_mean_vector()

#Calculate mean covariance
sigma = data.get_period_return_covariance_matrix()q = 0.5                    # set risk factor
budget = 2           # set budget (B)
penalty = num_assets      # set parameter to scale the budget penalty termmdl = Model('docplex model')
x = mdl.binary_var_list(num_assets)# set objective function:
#
#     maximize { mu^T * x - q * x^T * sigma * x }
#objective = mdl.sum([mu[i] * x[i] for i in range(num_assets)])
# mu^T * x
objective -= q * mdl.sum([sigma[i][j]*x[i]*x[j] for i in range(num_assets) for j in range(num_assets)])
mdl.maximize(objective)# add budget constraint:
#
#     1^T * x == budget
#
cost = mdl.sum([x[i] for i in range(num_assets)])
mdl.add_constraint(cost == budget, ctname='budget')# converting to Quadratic Program
mod = QuadraticProgram()
mod.from_docplex(mdl)#removing the constraint to create the QUBO
lineq2penalty = LinearEqualityToPenalty(penalty)
qubo = lineq2penalty.convert(mod)#converting QUBO to an Ising Hamiltonian
H, offset = qubo.to_ising()H = H.to_legacy_op()  # coverting it to a legacy operator
H.print_details()
```

Next comes the change of Hamiltonians into Pauli Operator Dictionaries:

```
def pauli_operator_to_dict(pauli_operator):

    d = pauli_operator.to_dict()
    paulis = d['paulis']
    paulis_dict = {}for x in paulis:
        label = x['label']
        coeff = x['coeff']['real']
        paulis_dict[label] = coeffreturn paulis_dictpauli_dict = pauli_operator_to_dict(H)print(pauli_dict)
```

2) Generation of Ansatz: By defining ansatz, we help define the circuit that is used to optimize the parameters to achieve the most optimal portfolio. The code for the same is:

```
def ansatz(theta):

    qml.RX(theta[0], wires=0)
    qml.RX(theta[1], wires=1)
    qml.RX(theta[2], wires=2)
    qml.CNOT(wires=[0,1])
    qml.CNOT(wires=[0,2])
    qml.CNOT(wires=[1,2])
    qml.RX(theta[3], wires=0)
    qml.RX(theta[4], wires=1)
    qml.RX(theta[5], wires=2)
    qml.CNOT(wires=[0,1])
    qml.CNOT(wires=[0,2])
    qml.CNOT(wires=[1,2])
    qml.RX(theta[6], wires=0)
    qml.RX(theta[7], wires=1)
    qml.RX(theta[8], wires=2)

    return ansatz

    ################################

    @qml.qnode(dev1)
def circuit_IIZ(params):
    ansatz(params)
    return qml.expval(qml.Identity(0)@qml.Identity(1)@qml.PauliZ(2))@qml.qnode(dev1)
def circuit_IIZ(params):
    ansatz(params)
    return qml.expval(qml.Identity(0)@qml.Identity(1)@qml.PauliZ(2))@qml.qnode(dev1)
def circuit_IZI(params):
    ansatz(params)
    return qml.expval(qml.Identity(0)@qml.PauliZ(1)@qml.Identity(2))@qml.qnode(dev1)
def circuit_ZII(params):
    ansatz(params)
    return qml.expval(qml.PauliZ(0)@qml.Identity(1)@qml.Identity(2))@qml.qnode(dev1)
def circuit_IZZ(params):
    ansatz(params)
    return qml.expval(qml.Identity(0)@qml.PauliZ(1)@qml.PauliZ(2))@qml.qnode(dev1)
def circuit_ZIZ(params):
    ansatz(params)
    return qml.expval(qml.PauliZ(0)@qml.Identity(1)@qml.PauliZ(2))@qml.qnode(dev1)
def circuit_ZZI(params):
    ansatz(params)
    return qml.expval(qml.PauliZ(0)@qml.PauliZ(1)@qml.Identity(2))@qml.qnode(dev1)
def circuit_IZI(params):
    ansatz(params)
    return qml.expval(qml.Identity(0)@qml.PauliZ(1)@qml.Identity(2))@qml.qnode(dev1)
def circuit_ZII(params):
    ansatz(params)
    return qml.expval(qml.PauliZ(0)@qml.Identity(1)@qml.Identity(2))@qml.qnode(dev1)
def circuit_IZZ(params):
    ansatz(params)
    return qml.expval(qml.Identity(0)@qml.PauliZ(1)@qml.PauliZ(2))@qml.qnode(dev1)
def circuit_ZIZ(params):
    ansatz(params)
    return qml.expval(qml.PauliZ(0)@qml.Identity(1)@qml.PauliZ(2))@qml.qnode(dev1)
def circuit_ZZI(params):
    ansatz(params)
    return qml.expval(qml.PauliZ(0)@qml.PauliZ(1)@qml.Identity(2))
```

3) Development of VQE circuit: The Variational Quantum Eigensolver(VQE) takes in the prepared experiment and then performs operations to obtain the most optimal state of the system. VQE majorly comprises of two parts:

- *Classical Part:* Performs optimization operation on the fed in Quantum State and provides the result.
- *Quantum Part:* It uses the prepared Experiment as the input and a feedback from the classical part to prepare the most optimal *ansatz circuit.*

The choice of VQE to perform the optimization, greatly enhances the result and the optimization of the portfolio.

The corresponding code for the same development of the VQE Circuit is as follows:

```
def vqe(parameters):

    expval_ZII = pauli_dict['ZII'] * circuit_ZII(parameters)
```

```
    expval_IIZ = pauli_dict[’IIZ’] * circuit_IIZ(
    parameters)
    expval_IZI = pauli_dict[’IZI’] * circuit_IZI(
    parameters)
    expval_ZZI = pauli_dict[’ZZI’] * circuit_ZZI(
    parameters)
    expval_ZIZ = pauli_dict[’ZIZ’] * circuit_ZIZ(
    parameters)
    expval_IZZ = pauli_dict[’IZZ’] * circuit_IZZ(
    parameters)

    # summing the measurement results
    total_expval= expval_ZII + expval_IIZ +
    expval_ZIZ +  expval_ZZI + expval_ZIZ +
    expval_IZZ

    return total_expval
```

Introducing the Gradient Descent for optimization of the ansatz used:

```
opt = qml.GradientDescentOptimizer(stepsize=0.5)
value = []
    # optimize parameters in objective
params = np.random.rand(9)steps = 500
for i in range(steps):
    params = opt.step(vqe, params)
    value.append(vqe(params))
plt.plot(value)
```

Generating the final Circuit the code we have is as follows:

```
@qml.qnode(dev1)
def final_circ(params):
    ansatz(params)
    return qml.probs(wires=[0,1,2])
```

and the resulting eigen state for the following code that we have are plotted as:

```
plt.bar([’000’, ’001’, ’010’, ’011’, ’100’, ’101’,
    ’110’, ’111’], final_circ(params[0]))
```

This ends the section briefing about the development of codes for Portfolio Optimization using Pennylane and Qiskit by Xanadu and IBM.

IV. Results

After successful implementation of the codes, we have the following generated results:

A. *Results using Qiskit*

The results provided by qiskit indicate the probability of having the state of [0 1 0 1] as the **most optimal portfolio.**

The results obtained are listed below:

```
Optimal: selection [0. 1. 0. 1.], value -0.0011

----------------- Full result ---------------------
selection        value           probability
---------------------------------------------------
 [0 1 0 1]      -0.0011         0.9434
 [0 0 1 1]      -0.0095         0.0364
 [1 0 1 0]      -0.0079         0.0096
 [1 1 0 0]      0.0012          0.0053
 [0 1 1 0]      -0.0067         0.0049
 [1 0 0 1]      -0.0018         0.0004
 [1 0 0 0]      4.0002          0.0000
 [1 0 1 1]      3.9905          0.0000
 [0 1 1 1]      3.9917          0.0000
 [1 1 1 0]      3.9933          0.0000
 [0 0 1 0]      3.9922          0.0000
 [0 0 0 1]      3.9980          0.0000
 [1 1 0 1]      3.9992          0.0000
 [0 1 0 0]      4.0009          0.0000
 [0 0 0 0]      16.0000         0.0000
 [1 1 1 1]      15.9917         0.0000
```

B. *Results using Pennylane*

The results obrtained by Pennnylane using the VQE method, has helped to get a good optimized result at the state of [1 0 1]. The plotted results of the optimally generated ansatz is as follows:

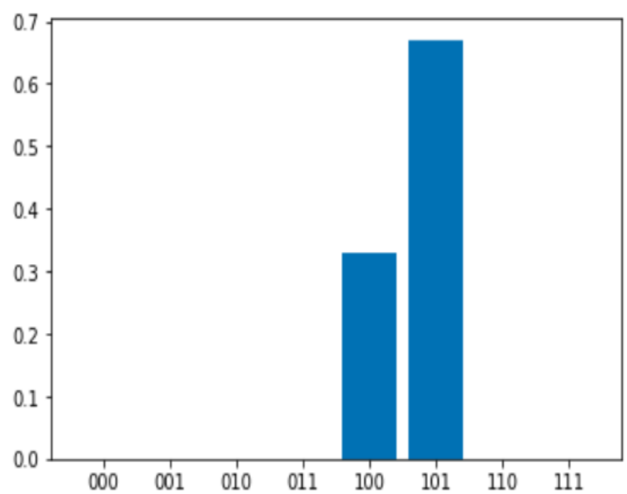

Figure 10. Optimal Portfolio State

V. Conclusion

Thus we conclude that we were able to get a brief overview about Quantum Finance and how we can perform Portfolio Optimization on real Quantum Hardware.

References

[1] NYSE Group. ”New York Stock Exchange/Archipelago Holdings Merger Complete.” Accessed Oct. 10, 2020.

[2] Intercontinental Exchange. ”IntercontinentalExchange to Acquire NYSE Euronext For $33.12 Per Share in Stock and Cash, Creating Premier Global Market Operator.” Accessed Sep. 6, 2020.

[3] National Park Service. ”New York Stock Exchange.” Accessed Sep. 6, 2020.

[4] B. Aouni, Multi-Attribute Portfolio Selection: New Perspectives,Information Systems and Operational Research, 47 (2009), no.1, 1-4.

[5] M.A. Nielsen, I.L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2010.

[6] P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM J. Sci. Stat. Comput. 26 (1997) 1484, https://doi.org/10.1137/S0097539795293172

[7] Nada Elsokkary, Faisal Shah Khan, Davide La Torre, Travis S. Humble,Joel Gottlieb. Financial Portfolio Management using D-Wave’s QuantumOptimizer: The Case of Abu Dhabi Securities Exchange. IEEE High Per-formance Extreme Computing Conference,HPEC, Sept. 2017 (edited)

[8] F. Ben Abdelaziz, B. Aouni, R. El Fayedh, Multi-objectivestochastic programming for portfolio selection, European Journalof Operational Research, 177, (2007), no. 3, 18111823

[9] H. Markowitz, Portfolio Selection, The Journal of Finance 7(1952), 77-91.

[10] Ors, Romn, Samuel Mugel, and Enrique Lizaso. Quantum computing for finance: overview and prospects. Reviews in Physics 4 (2019)(arXiv:1807.03890 [quant-ph])
[11] French, Craig W., Another Look at the Determinants of Portfolio Perfor-mance: Return Attribution for the Individual Investor (July 29, 2003).https://ssrn.com/abstract=706603orhttp://dx.doi.org/10.2139/ssrn.706603
[12] https://blog.quantinsti.com/portfolio-optimization-methods/
[13] Dynamic Portfolio Optimization with Real Datasets Using Quantum Pro-cessors and Quantum-Inspired Tensor Networks, Samuel Mugel, CarlosKuchkovsky, et.al., arXiv 2007.00017, 2020
[14] Option Pricing using Quantum Computers, Stamatopoulos, Nikitas, Egger,Daniel J., et.al., Quantum issn 2521-327X, vol 4, page 291, July 2020., Pub-lisher: Verein zur F orderung des Open Access Publizierens in den Quan-tenwissenschaften
[15] De Looze, M.P., et al.: Cost effectiveness of ergonomic interventions in production. Hum.Factors Ergon. Manuf. Serv. Ind.20(4), 316–323 (2010
[16] Deutsch David 1985Quantum theory, the Church–Turing principle and the universal quantum computerProc. R. Soc. Lond. A40097–117
[17] Noson S. Yanofsky and Mirco A. Mannucci. 2008. Quantum Computing for Computer Scientists (1st. ed.). Cambridge University Press, USA.
[18] Chris Bernhardt. 2019. Quantum Computing for Everyone. The MIT Press.
[19] Warren WS (1997). ”The usefulness of NMR quantum computing”. Science.
[20] ”A new kind of quantum”. spie.org. Retrieved 2021-03-26.
[21] https://www.ibm.com/quantum-computing/technology/quantum-in-finance/
[22] “Qiskit/ibmqx-user-guides.” GitHub. November 13, 2018.
[23] Picking optimal portfolios using Quantum Computing, Agarwal Rochisha, https://medium.com/@rochishaagarwal/picking-optimal-portfolios-using-quantum-computing-7a8e89c24689