

**Solving the Unit commitment problem using Quantum Computing research internship report**

AIS390

# Internship Details

This internship was conducted last summer at Nile University under the supervision of Dr. Marwa Sorour and Dr. Ahmed El-Mahdy. The purpose of the internship was to grasp the basics of quantum mechanics and quantum computing. I was part of a group, and together, we spent an average of 10 hours daily researching and studying the unit commitment problem and how to solve it using a quantum approach. At the end of the internship, we presented the results of our study to a group of doctors, TAs, and representatives from Japan.

# Introduction

Optimizing energy production has become more challenging with the increasing of variable renewable energy sources like wind and solar. Accounting for the uncertainty and variability of renewable generation is crucial for maximizing their utilization while maintaining system balance. This is where The Unit Commitment problem (UCP) comes in. UCP is fundamental in the electric power industry. The objective of UC is to determine an optimal schedule for each generating unit to meet the demand for power with the minimum cost. This includes determining which generators work at what time and how many generators work simultaneously. While there are many classical approaches to this problem, we aim for a quantum approach. QUBO and QAOA will be utilized to create a functional quantum circuit. We first solve the problem classically using Cplex which gives the value of each variable and the minimum cost. Then, to implement QUBO we need to turn the classical constraints of the UCP into penalties as the QUBO model is incapable of handling constraints using a qiskit-optimizer. In this paper, the project “Explore the use of quantum computing for unit commitment to balance the electricity grid” from the “The UK Quantum Hackathon – July 2023” was used as a basis.

The team from UK Quantum Hackathon has formulated an objective function for the unit commitment problem, however they didn’t have time to formulate it in Quadratic Unconstrained Binary Optimization (QUBO) form. There are many algorithms which can solve optimization problems efficiently and easily if the problem is in QUBO form. In this project we converted the objective function into QUBO form and solving it using qiskit-optimizer on a simulator and on real quantum computer. As mentioned before there is classical solutions for the optimization problem which lead to a comparison between the classical solution and the quantum solution.

## Objective Function:

Goal (Min or Max): Minimize the cost of the objective function.

## List of used variables:

bj,t : A binary variable indicating the operational status of generator j at time t, where bj,t =1, if the generator is in use, and bj,t =0 otherwise.

uj, t: A binary variable indicating the altering status of generator j at time t, where uj,t =1, if the generator is starting up, and uj,t =0 otherwise.

dj,t : A binary variable indicating the altering status of generator j at time t, where dj,t =1, if the generator is shutting down, and dj,t =0 otherwise.

Dt: A fixed variable representing the demand (amount of energy needed) at time t.

Cj: A fixed variable representing the cost of running a certain generator j for one unit of time.

Pj: A fixed variable representing the amount of energy produced by a certain generator j for one unit of time.

Cj u: A fixed variable representing the cost of starting up unit j.

MNZT: the minimum time a generator must be active before shutting down.

MZT: the minimum time a generator must be down before turning on again.

## Constraints

### Positive Margin

This constraint ensures that, at every time unit t, the sum of the maximum possible production capacities of all active generators is greater than or equal to the energy demand . The purpose of this constraint is to guarantee that the energy demand is met at each time t.

### Minimum Non-Zero Time

This constraint ensures that for each generator j and every time unit t within the set , the generator can be turned on only once if it is currently active , or it has not been turned on at all within this interval. The purpose of this constraint is to guarantee that if the generator is active , it cannot be turned on more than once within the interval , as this would imply it was shut down at least once within this period, violating the requirement of maintaining a minimum nonzero time (MNZT). Conversely, if the generator is not active , it cannot be turned on within this interval, as this would indicate a shutdown before satisfying the MNZT condition.

### Minimum Zero Time

This constraint ensures that for each generator j and every time unit t within the set , if the generator is currently active , then it must have remained off for the preceding MZT time units. Conversely, if the generator w). Turned off during any of these preceding MZT time units , it cannot be active at time t . The purpose of this constraint is to enforce a minimum zero time (MZT), ensuring that once a generator is turned off, it remains off for at least MZT consecutive time units before it can be turned on again.

### Binary Consistency:

This constraint ensures the consistency of the binary variables representing the state transitions of each generator. The constraint ensures that any change in the state of the generator from t−1 to t is accurately captured by the turn-on and turn-off variables. If a generator was on at t−1 and is off at t ( then it must have been turned off at t Conversely, if a generator was off at t−1 and is on at t (, then it must have been turned on at t−1 ( Further, if there was no change, t( or ( it means the generator must haven’t been turned on or turned off t( t This constraint ensures the logical consistency of the generator's operational states over time.

### QUBO formulation:

First challenge of converting the objective function into QUBO is the constrains as it is unconstrained formulation, so we have to add a slack variable to each inequality constrain and square the whole constrain then add it to the objective function. If the constrain is an equality then we square it directly without adding a slack variable and finally adding it to the objective function.

A table with math equations

Description automatically generated

This is the table that was used as the basis for the constraint into penalty transformation. 

Figures down show the conversion of each constrain.

Conversion of constrain number 1 (Positive Margin):

Conversion of constrain number 2 (Minimum non-zero time (MNZT)):

Conversion of constrain number 3 (Minimum zero time (MZT)):

Conversion of constrain number 4 (Binary consistency):

### Solving QUBO using Quantum Computing:

We have used Quantum Approximate Optimization Algorithm (QAOA) to solve the QUBO function. It is an optimization algorithm which solve the problem by constructing the Hamiltonian of the solution state through a ground state which its Hamiltonian is known then slowly reaching the approximate solution state.

The purpose of quantum approximate optimization algorithms is to is that it encodes a QUBO problem to Hamiltonian to find the ground state for the problem, which represents the optimal result. The ansatz of the QAOA problem is what keeps alternating between the problem Hamiltonian and the mixer state, which is the initial state that we know it's ground state It uses a classical Optimizer to initialize the parameters of the circuits.

A computer screen shot of a circuit

Description automatically generated

Qiskit Optimization:

Qiskit open source were used to run the algorithm.

Results:

A graph with numbers and lines

Description automatically generated

A graph with a line

Description automatically generated

A graph with a line

Description automatically generated

A graph with a line

Description automatically generated

Difference between simulators and real quantum computer:

A graph of a number of blue squares

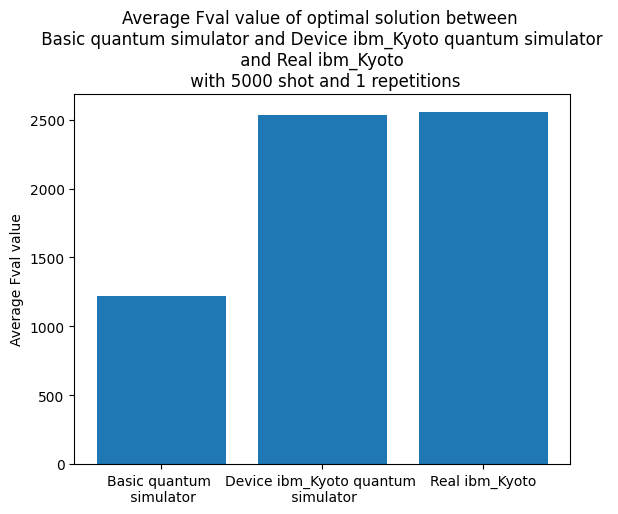
Description automatically generated

A graph of a number of blue squares

Description automatically generated

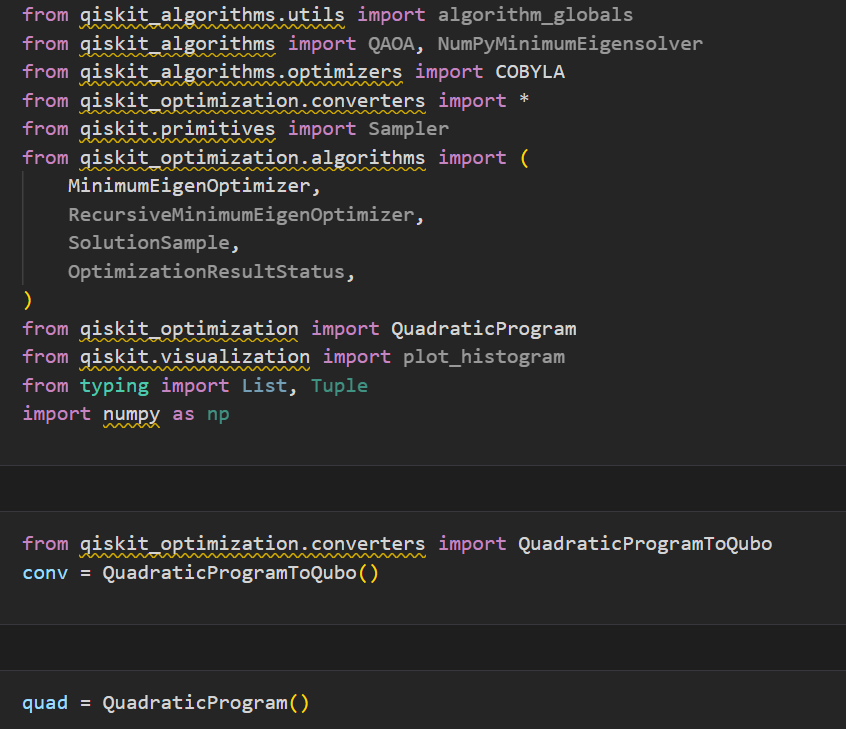
A graph with blue rectangular bars

Description automatically generated

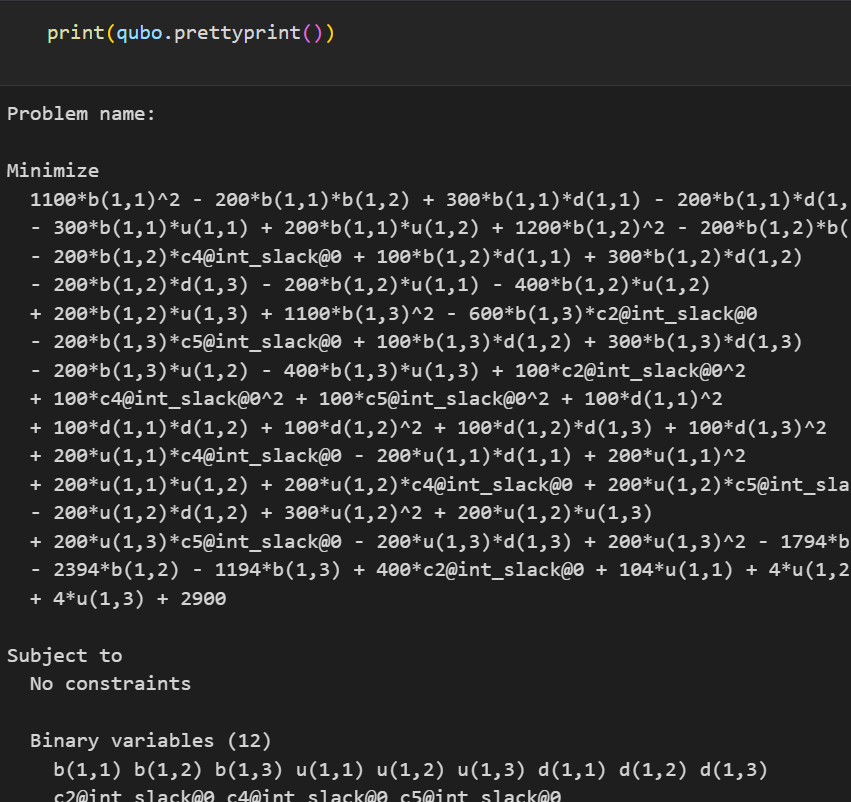


Technical part:

Importing necessary libraries:



Objective function after converting it in QUBO form and adding slack variables:



Solving problem using simulator:



Solving problem using noisy simulator:

A screenshot of a computer

Description automatically generated

Solving problem using real quantum computing:

A screenshot of a computer

Description automatically generated

Conclusion:

This research internship has provided invaluable insights into the potential of quantum computing to address complex optimization problems in the energy sector, particularly the Unit Commitment Problem (UCP). By successfully formulating the UCP into a Quadratic Unconstrained Binary Optimization (QUBO) problem and implementing it on quantum simulators and real quantum computers using Qiskit, the study demonstrated the analysis and challenges of applying quantum techniques to real-world problems. The comparison between classical and quantum solutions highlighted the current limitations of quantum computing but also underscored its promise for future advancements. The experience has increased my understanding of both the theoretical and practical concepts of quantum computing and its application in optimizing energy production.