Quantum approximate optimization algorithm

* Variational algorithm that solves "Combinatorial optimization" problem

What does it do?

* Uses a parametarized circuit (ansatz) where
  * each quantum state can output a probable solution (each input has a certain parameter where the probability for each input can differ)
  * Parameters adjusted to minimize cost hamiltonian

Rule for adiabatic theorm

Adiabatic theorm in quantum mechanics

Base theorm

* if a quantum system starts from a specific eigen state and gradually, the time step increases. The time step will continue to increase while it remains in the ground state

∴ eg: if it started from ground state & evolved over the period of time (gradually) it will remain in the ground state

Long story short

- if the time evolves slowly while the eigen state still in ground state. It will keep track of every optimal solution for every time step while holding the optimal solution (still staying in ground state)

hamiltonian interpretation

equation for hamiltonian interpolation

- $H(t) = \left[(1-t) H_m + t H_c \right]$

- H_m : mixer hamiltonian : the starting point and we know the solution of our starting point (ground state)

- H_c : cost hamiltonian : the final mark but as the energy level for every time step evolves, we get parameterized solutions but we still keep track of the ground state solution

$$H(t) = (1-t) H_m + t H_c \qquad t \in [0,1]$$

our main aim here is that our time range will only be from 0 to 1 but the time step from 0 to 1 is what matters why?

So $t \in [0,1]$ is like energy level $n=0$ to energy level $n=1$ and we need to stay within the ground state region

General QAOA step by step!!!

Step 1: Define cost hamiltonian

def the cost hamiltonian H_C

- H_C: cost hamiltonian or the problem you are trying to solve

What do I want to minimize?

1) crew inefficiencies
2) flight delay
3) flight cost

What do we do to minimize it?

- we can add penalty function to each problem