Beomgyu Choi, Yeongseok Lee

*2022-June.-29th*

Quantum Game Problem

1. **Odd Cycle Game**
   1. Game rule

Bob

Alice



Referee

v1

v2

v5

v3

v4

Cyclic

graph

<fig1>

Bob and Alice each receive one point from Referee. Each point satisfies the following conditions.

First, they get the same point from Referee. Second, they get the point on the edge, but Bob’s vertex number should bigger than Alice’s vertex number in mod n (If Alice get v5 vertex Bob should have v5 or v1 in <fig1>). When they got point, they must label their point +1 or -1. When they receive the same points, they must label the same or they receive the points on the edge, they must label them differently. Alice and Bob can communicate each other before starting the game. They know the shape of cyclic graph and vertex number before starting the game. So the best strategy of classical method is label differently on every edge (fig2). However, this method can no longer be applied when the number of vertices is odd. In this case, it is not possible to define different vertices in one edge, so if the number of vertices is n, the probability of winning the game is 1/2n.

In this case, In this case, the quantum strategy is superior to the classical strategy. In quantum strategy, Alice and Bob share two entangled qubit each other. These qubits have the state . After that, Alice and Bob measure their own qubits in this direction in figure 3.

<fig2, even cyclic graph>

v6

v1

v2

v5

v3

v4

Cyclic

graph

1

-1

1

1

-1

-1

|  |  |  |
| --- | --- | --- |
|  | Alice | Bob |
| +1 | cos) | cos) |
| -1 | -sin) | ) |

<fig3, measure basis>

텍스트, 시계이(가) 표시된 사진

자동 생성된 설명

<fig4, measure basis through gate>

In this condition, If Alice and Bob have the same vertex, their winning probability is . And if they have different vertex, their winning probability is = . So, in this case their winning rate is and this is bigger than 1-1/2n which is classical. comes from to make symmetry that vertex vn and v1 is same as v0 and v1. And comes from if we change that value as x, winning probability becomes and this function have maximum value when . The method of measuring in this tilted axis is a method of turning the tilted value in that direction to the actual 0 state when the value is 0 state. So, representing this method as circuit is figure 4.

|  |  |  |
| --- | --- | --- |
|  | Classical strategy | Quantum strategy |
| N=3 | 83% | 93% |
| N=5 | 90% | 97.5% |
| N=7 | 92.9% | 98.6% |
| N=9 | 94.4% | 99.2% |

<fig5, winning probability of each strategy >

However, in quantum computing, there exist noise that make winning rate decrease. So, there is a point that classical strategy is better than quantum strategy. So, we tried running these programs on Ion-Q.

|  |  |  |
| --- | --- | --- |
|  | Classical strategy | Quantum computer |
| N=3 | 83% | 90% |
| N=5 | 90% | 95% |
| N=7 | 92.9% | 97.4% |
| N=9 | 94.4% | 97.9% |
| N=11 | 95.4% | 97.2% |
| N=13 | 96.2% | 96.9% |
| N=15 | 96.7% | 98.3% |
| N=17 | 97.1% | 98.4% |
| N=19 | 97.4% | 96.8% |

<fig5, 6omparing classical strategy and Q.C> >>

As figure6, we can see that when total number of verteces is bigger than 19, Quantum computing results have no longer advantage because of noise.

1. **Magic square game**



nm rectangular

n,m > 3

…

…

1

-1

-1

-1

1

1

1

Referee

Bob

Alice

<fig6, Magic square game>

In Magic square game, there is a matrix that every row and column have the value that +1 or -1. This value means that every entry in that row or column, their product must that row or column’s value. When the game starts, Alice gets one row and then Alice decides her value at each entry that values are valid. Also, Bob gets one column and then he decides his value at each entry that values are valid. Whenever Alice and Bob get their own row and column, there is a single point that their row and column intersect. When this value is same, Alice and Bob win else, they lose.

Alice and Bob can communicate before starts the game. They know the shape of matrix, and the values of each row or column. The way to win the game in classic strategy is for Alice and Bob to assign a value to each point in advance. If there are even numbers of -1, they can fill full entries of matrix. So, in this case their winning probability is 100%. However, if there are odd numbers of -1, there is a single paradox point that they can’t fill. In this case, classical strategy no longer have perfect winning strategy and best probability is 1-1/nm.



Alice

Referee

Bob

…

…

nm rectangular

n,m > 3

1

-1

-1

-1

1

1

1

-1

-1

-1

-1

-1

1

1

1

1

<fig7, Magic square game filled full entries>

<fig8, Magic square can’t fill full entries>



Alice

Referee

Bob

-1

nm rectangular

n,m > 3

1

-1

-1

-1

1

1

1

-1

-1

-1

1

-1

1

1

?

1

-1

1

How about Quantum strategy? Surprisingly, there is a way that maintain 100% winning probability.

We can reduce matrix to matrix. We can fill entries of row or column until the matrix reduce to matrix. By doing this, we can consider matrix that have odd -1. Also, this matrix has similarity that if there are 5 odd numbers, that can be reduced to 3 odd numbers or 1 odd number if we properly choose the number of another entry that is not inside the matrix.

That means only we must do is solve 1 matrix that have one paradox point. We use matrix that value of rows is 1 and value of columns are -1. As figure 9, We can put Pauli matrix to decide entry’s measure method. For example, means measure the second value in z axis and put that number if then +1 else then -1.

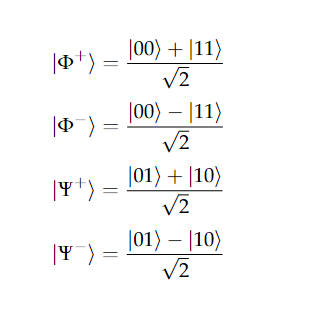
Alice and Bob share 4qubits each other and those qubits have state

Alice get 1,2 qubits and Bob get 3,4 qubits. By doing this, Alice and Bob can have same result when they measure in same direction. When Alice or Bob assign 1 or 2 column or row, it can be easy to measure the value because they only measure their qubits

to following direction as in table. However, if Alice

gets 3rd row, or Bob gets 3rd column this method

can’t be answer because according to Bob, Bob must measure his two qubits in z direction and because he

already measured his all qubits, he can’t measure his second entry. In this case we must use entangled basis. Entangled basis is basis that two or more values in linear representations. For example, Bell states are entangled basis that have two values each basis. When we use entangled basis, we must use eigen states that every measurement has. That means that if we decide entangled basis and measure that, three entries should be filled with value not probability.

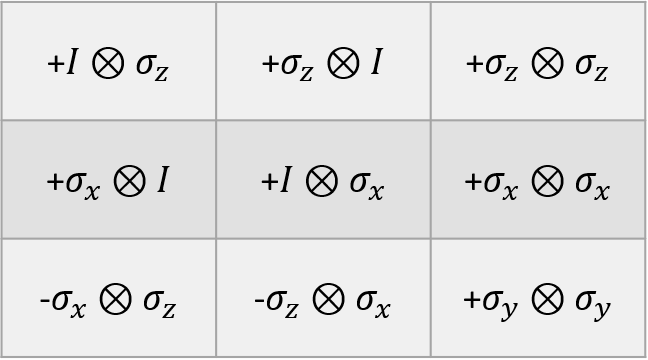
<fig9, Quantum strategy matrix>

(Math metical formula)

The method to find that eigen states and value is using linear algebra. If we assume is eigen state of these three measurements, we can write

and each ,,have eigen value +1 or -1, if we put ,

<fig10, Bell basis>



we can figure out which value should be in that entry. For example, if eigen value is 10109, we should put

-1, 1, 1 each entry. To solve the 3rd row or column problem, we must find eigen state by using that. When calculating that value, the result is

e-state: , , e-state: ,

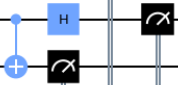
e-state: , and e-state: , (-1,1, 1)

and Alice’s eigen-states of row 3 is

e-state: , , e-state: ,

e-state: , and e-state: , (1,1, 1)

Now we know the eigen states, we should design circuit. The way to design circuit is that find the value that first and second value are same and initialize the state to that’s eigen state and find gates by using trans pile function and inverse function. Finally, we get the inverse gate to find entangled basis. We can get the first value of entry by measuring the second qubit and second value of entry by measuring the first qubit.

 텍스트, 시계, 표지판이(가) 표시된 사진

자동 생성된 설명

<fig12, 3rd row>

<fig11, 3rd column>

By using this quantum method, winning probability becomes 100%. However, because quantum computer has noise, there should be losing probability. Figure 12 is losing probability of each entry.

|  |  |  |
| --- | --- | --- |
| 2.73% | 2.25% | 11.43% |
| 3.02% | 3.51% | 11.43% |
| 8.20% | 9.37% | 10.02% |

<fig12, Result of quantum computer>

Quantum computer’s winning probability is 93.11% that means the possible lowest matrix is and classical probability is 93.75%, quantum computer don’t have advantages compare to classical strategy.

1. **Magic pentagram game**

‘



Alice

Referee

Bob

(red edge)

(black edge)

<fig13, Magic pentagram game>

Magic pentagram game is like magic square game. In this case Alice gets 1 edge of 5 edges and fill in entries properly then Referee give Bob a single vertex and Bob can label that entry at will.

Classical strategy is fill in vertices with out 1 paradox point. So, probability of winning is 1/20.

Quantum strategy is like magic square. In this case, Alice and Bob share 6 qubits that are entangled in this way.

Alice gets 1,2,3 qubits and Bob gets 4,5,6 qubits. If they measure in same direction, they will see the same value because qubits are entangled. In figure 14, we can find that there is a need to use entangled basis which is edge 1. Other edges can use single qubit basis. So, calculate the edge 1,

e-state: , *,* e-state: ,

e-state: , *,* e-state: ,

IIX

XZX

XXZ

ZXX

ZZZ

ZII

IIZ

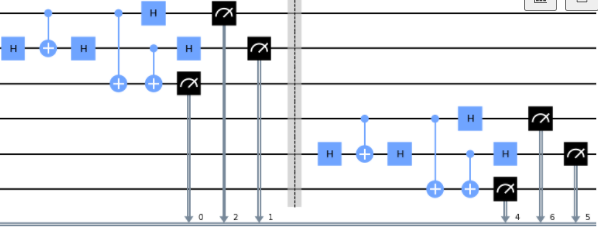
IXI

IIZ

IZI

XII

<fig14, Pauli matrix of each entry>

And gates to confirm that basis is figure 15. Quantum strategy’s winning probability is 100%. However, noise occur and there is some losing probability. Lossing probability by using ion-q computer is figure 16. We can figure out entanglement makes the winning probability worse. Also, quantum computer’s average winning

probability 94.31% is lower than classical strategy’s winning probability 95%.

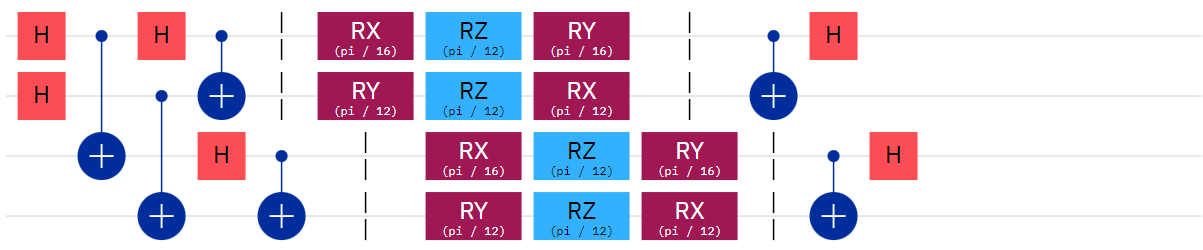
<fig15, gate of edge 1>

|  |  |
| --- | --- |
| edge | 10 iteration mean error value |
| 1 | 0.125 |
| 2 | 0.0391 |
| 3 | 0.0439 |
| 4 | 0.0394 |
| 5 | 0.0371 |

1. **Further**

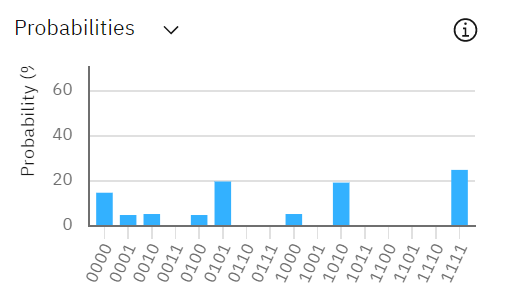
Why entangled state has many noises? We thought that entangled basis is more sensitive than single one.

<fig16, Losing probability>

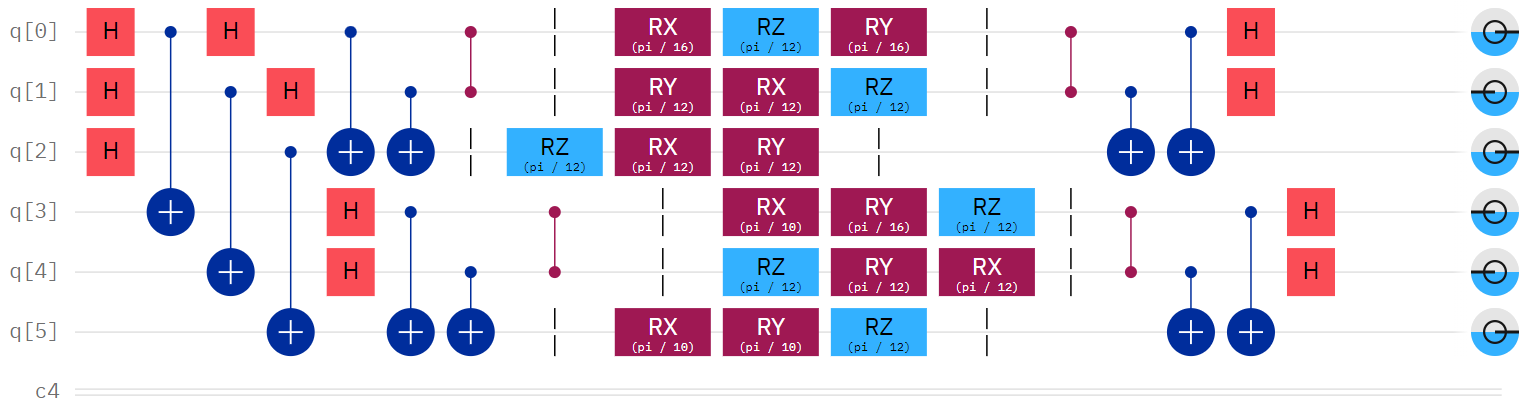


<fig17, Noise modeling 4qubits>

We made noise modeling algorithms left side is entangle and initialize the qubit and middle is noise and right are measure of that state. If there are no noise, the final state should be combination of 0000,0101,1010,1111. However, when we see figure 18, noise make some distortions. In this case probability decrease from 100% to 78.84%. However, in single basis probability decrease from 100% to 88.31%. How about 6 qubits case? In single qubit basis, probability decrease from 100% to 76.8% and entangled basis decreases from 100% to 66.36%. This tells us that entangled basis is more sensitive to noise, and this is the one of the reasons why entangled basis have more losing probability. When 4qubits single basis probability is 96%, entangled basis probability becomes 93.6% and when 6qubits single basis probability is 93.68%, entangled basis probability is 89.36%



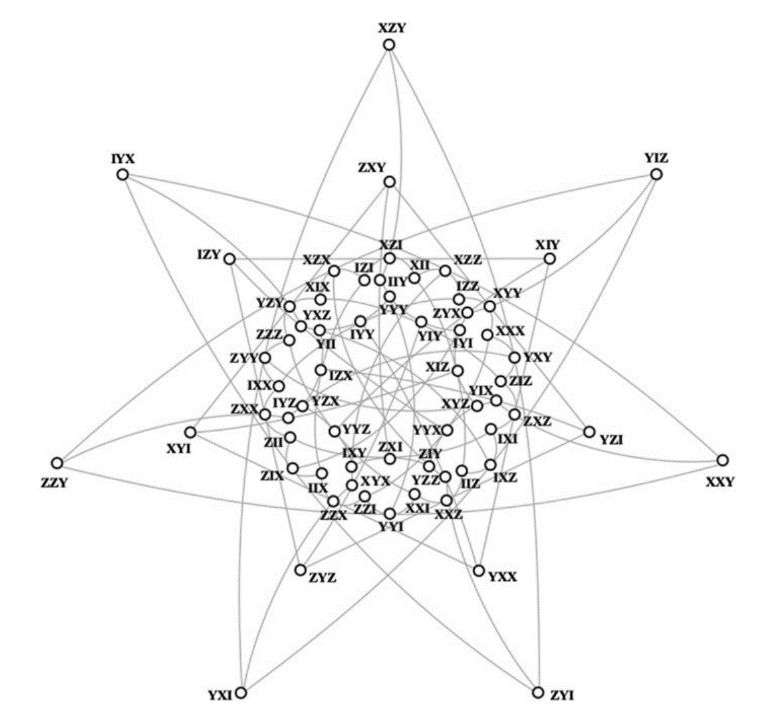
<fig18, Losing probability>



<fig18, Noise modeling 6qubits >

More complicate questions

We know that we have to Pauli matrix properly to solve the problem. However, what is the method to locate them properly? The answer is they must commute each other. Commute means they have same value even they change their sequence and share same eigen state. For example, and is commute because [, *- = - =0* and and is not commute each other because [=- in this case there is no eigen state they share. We can simply know if we want to design 3qubits entangled or single basis measurement, it is needed to find whether they commute or not. If we follow that rule, we can match all 63 basis

( to one vertex. That means we can develop quantum strategy that if there is a shape that every vertex shares 3 edges, we can make quantum strategy to win 100%. Figure 19 is the case that every 63basis used. If Alice and Bob know how to find eigen states of their qubits and find the inverse gate of that eigen states, they can win the game every time. If this pseudo communication trimmed and well defined to another cases, we can optimize some problems without communicate each other.

<fig19, 3qubits, 63 different basis>

<fig18, Noise modeling 6qubits >