Quantum BlackJack

James Liu, Sidharth Srinivas

October 2022

1 Quantum Blackjack

In a normal game of BlackJack, players are presented with two cards, after which they may decide to draw more (hit) or stop (stand) in order to get as close as possible or equal to 21 without passing it. Additionally, there also exists a dealer whose card total the player must beat. In summary, the player must draw cards to get a card total higher than the dealers, while trying to get as close to 21 without passing it.

Now what is Quantum Blackjack? Quantum Blackjack is a unique card game that mostly follows the logic of normal blackjack, but with a quantum twist. Instead having cards be confined to one of two states (played or not played), the cards can exist within the superposition of states in-between. This way, both the dealer and the player are not sure if a card will count towards the card total until it has been measured - they can manipulate a card towards a favorable outcome, but they cannot ensure it. Essentially, Quantum Blackjack is a version of BlackJack where players are dealt a pre-determined number of cards, and must use quantum gates to manipulate them into being measured favorably. This is done using the following gates: H, X, Y, Z, RY(pi/2), RX(pi/2), RZ(pi/2), SWAP, and CNOT.

2 Gameplay

As a player, you have a few items. These include the 5 cards that you are dealt at the beginning and tokens (20 in easy mode and 40 in hard mode) that can be used to apply quantum gates (each operation costs different amounts).

At the start of the game, our program generates 5 random cards that are ”dealt” to the player. For each card you are dealt, there also exists a corresponding qubit is used to represent the superposition of that card. If this qubit measures a $|1\rangle$ at the end of the game, then its value is added to the players card total. Otherwise, it is not counted. At the start of the game, each card is set to a random state. In easy mode, the computer applies a random I, X, or H gate to each qubit to generate the random state. In hard mode, the computer applies a unitary with random values of lambda, theta, and phi. Because of this, it is possible to guarantee a card is counted in easy mode, while hard mode makes it much more challenging and requires the user to have a better understanding of bloch sphere rotations to win. Additionally, just like in normal BlackJack, the dealer is dealt one face up and one face down card that will be used to determine the dealer's card total.

Once the setup is complete, each round consists of the user spending tokens to apply a quantum gate to a specific qubit in order to manipulate its state to as close to $|1\rangle$ as possible. Another strategy involves moving unfavorable cards as close to $|0\rangle$ as possible. To add to the complexity, the player also has the option to use multi-qubit gates to entangle multiple cards. For example, the CNOT gate can be used to ensure that two or more cards are counted together, or that one is only counted when the other isn't. The SWAP gate can also be used to swap two cards' states if one card that is unfavorable is easier to manipulate into the $|1\rangle$ state than a more favorable card. Each token costs a different amount, so the player must decide on what manipulations are more important than others in order to create the most desirable outcome.

Once the player is either out of tokens, or applies the measurement gate, the qubits' states are collapsed as they are measured. If the qubit measure a $|1\rangle$, then the corresponding card is added to the player's card total. Additionally, the dealer randomly draws cards until its card total is above 16. Thus, the user only wins if their card total is less than or equal to 21 and higher than the dealers card total.

We also implemented a two-player mode, where one player's card total would the cards measuring $|1\rangle$, and the other's would be the cards measuring $|0\rangle$. They still share the same 5 cards (but have their own tokens), and just take turns applying gates to the circuit. This way the two players are in direct opposition of each other, so every action they take directly hinders the opponent. The goal is to get a higher card total then the opponent with passing 21. This modifies the gameplay by introducing ways of winning such as purposely sending the opponent over 21, or entangling unfavorable cards with favorable ones to obstruct the opponent.

3 Design Choices

After every round, we display the quantum circuit that represents the state of the cards and the bloch sphere of every qubit. Because reversible single qubit gates act as rotations around the bloch sphere, we decided to show the bloch spheres after every gate application in order for the user to visually understand how they are affecting the state of the system. We believe that this will help players better understand the fundamentals of how quantum gates function and affect qubits.

4 Qiskit Details

In terms of the technical details, each of the 5 qubits are on a QuantumRegister that is stored in a QuantumCircuit. This allows us to use quantum gates to directly affect individual qubits and create entanglement with multi-qubit gates. Every action the user takes simply runs the corresponding method on the QuantumCircuit in order to apply it to the qubit at the specified index. Once the measurement gate is played, it is applied to every qubit in the circuit and its result is saved to a corresponding classical bit in a ClassicalRegister. After this, we run a simulation on the qasm simulator with 1 shot in order to determine the state that the system collapsed into. This bitstring is then parsed in order to determine which indexes contain a 1, and then the corresponding card values at those indexes are summed to create the final card total of the player.

5 Issues Encountered and Further Ideas

One feature that we wanted to implement was the ability to display the individual states of each qubit; however, we realized that there was no easy way to display this due to our multi-qubit gates. The only way to give users a good glimpse into the numerical state of the game would be to display the statevector of every possible outcome using the statevector library, but we thought this would be an overabundance of displayed data, and would also take away from the goal of having understand gate operations from a rotations viewpoint. Thus, we decided to limit the users information access to only the bloch spheres to create more elements of luck and cement a solid understanding of how gates create rotation. We also considered adding an extra hard mode where each turn the computer would also add a random gate to a random card. This could either help or hinder the player, but its main goal would be to add another element of randomness to the gameplay.

6 Why Quantum BlackJack?

One of the best gateways into learning a new subject is to take something fun and familiar and add a twist to it. Seeing how popular BlackJack was, we decided that it could be turned into a great teaching tool as it was a very accessible game and had game play that could easily be modified to highlight a very important concept to quantum computing.