Solving Magic Squares, the Hard Way: a Quantum Approach

Background

Magic squares have a long and rich history, and have seen use in fields ranging from mathematics, to divination, to computer science. They are linked to a wide array of subfields such as groups, combinatorics and matrices[1].

Grover's algorithm is a quantum algorithm that can efficiently find solutions to search problems like these, by making use of quantum superposition and an ***oracle***[2].

We combine these two ideas by using Grover's algorithm to solve simplified, binary semi-magic squares. To achieve this, we construct a suitable ***oracle*** - a quantum circuit that is capable of discerning valid from invalid square solutions.

1	1	0
1	0	1
0	1	1

A binary semi-magic square. In this square all rows and columns sum to two (10 in binary).

Objectives

We aim to investigate how mathematical search problems such as the magic square problem can be adapted to run on quantum algorithms.

Although efficient construction methods exist for finding solutions to magic squares, not all solutions can be found this way[3]. Our aim is to be able to find arbitrary solutions.

The number of possible solutions for 6 by 6 magic squares is an unsolved problem, although some estimates[4] put the number of solutions at ~10^{19}. A suitable Grover's oracle could also be used in a separate algorithm, ***quantum counting***, which could determine the number of solutions with reasonable accuracy in future work.

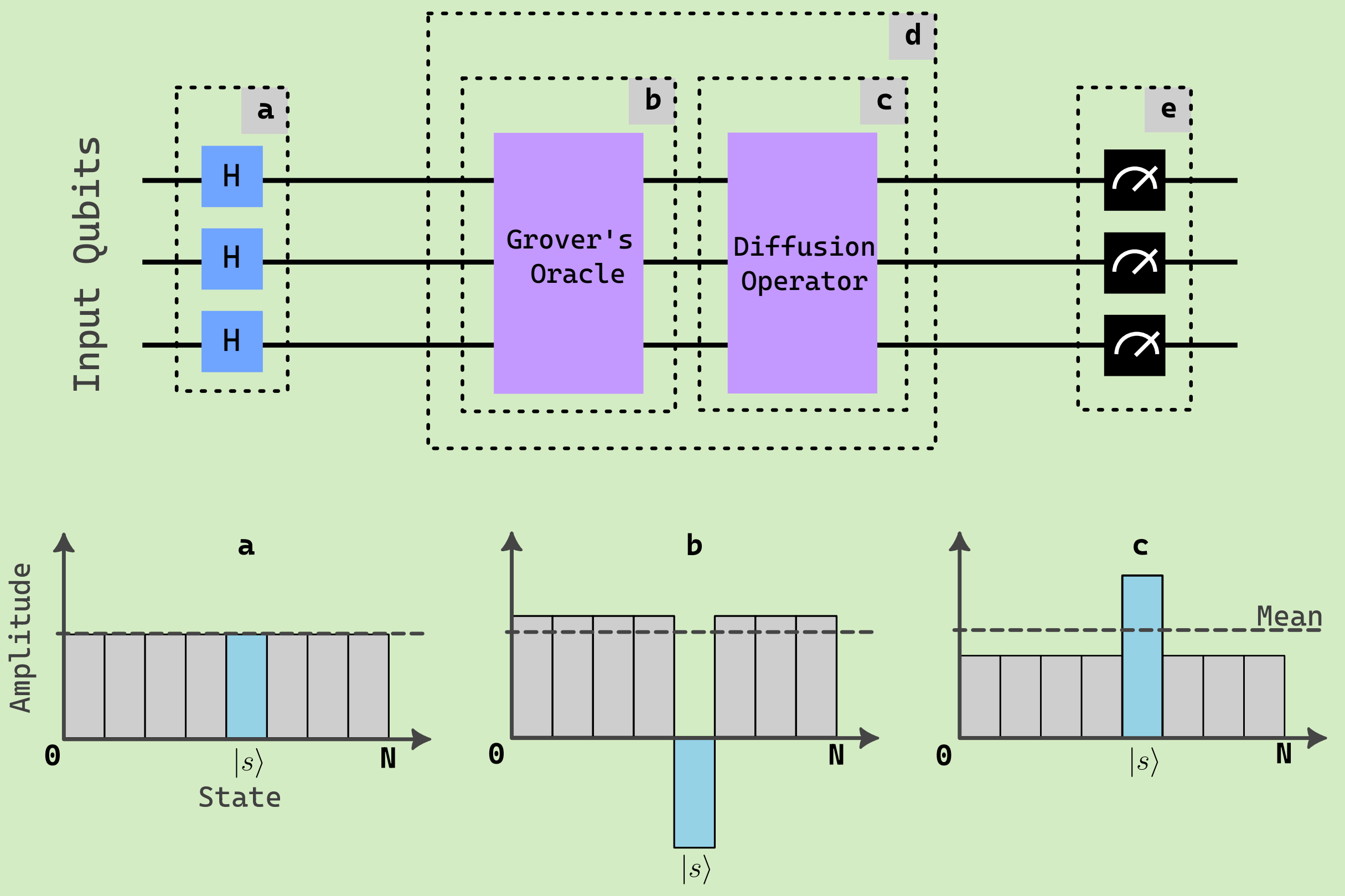
Methods

In this section we show how Grover's algorithm finds valid solutions to search problems. Two operations are needed to construct a suitable oracle. First, we need to be able to sum three (qu)bits together. Second, we need to be able to compare two of these sums and check for equality. If $r_{1/2/3}$ and $c_{1/2/3}$ represent the sums of each row and column in the 3x3 magic square, there are 5 equalities to check:

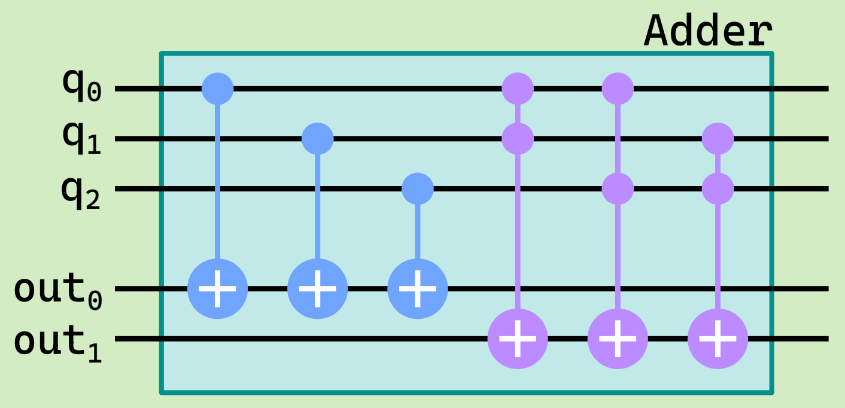
$$r_1 = r_2 = r_3 = c_1 = c_2 = c_3$$

Grover's Algorithm

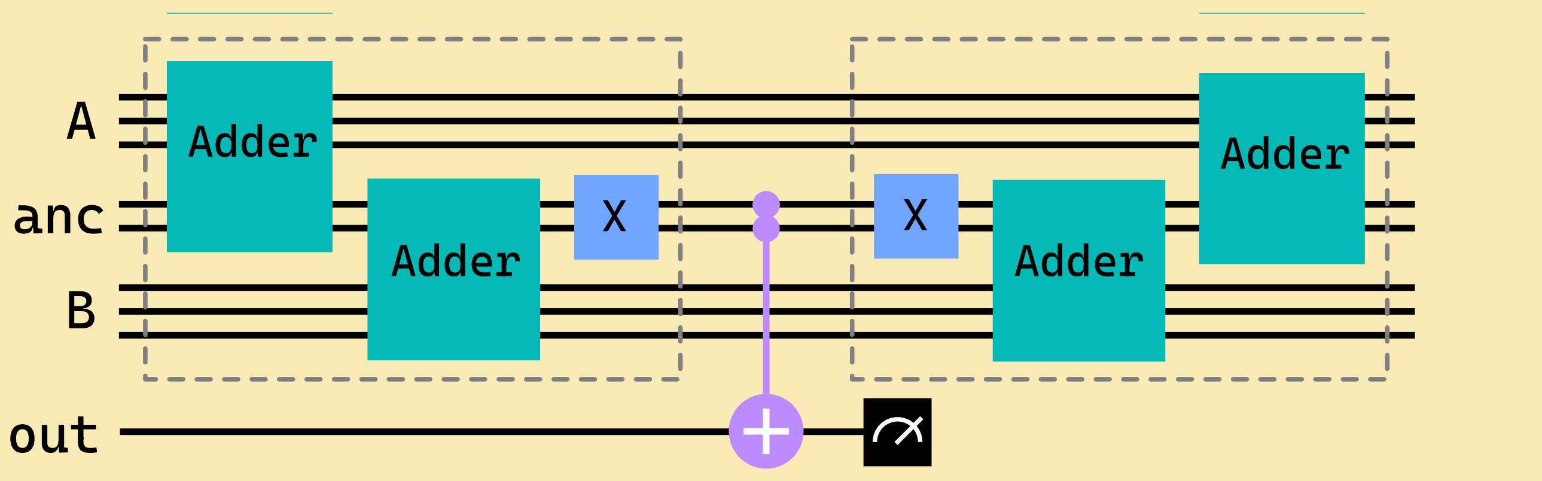
Qubits are first put in a superposition state using Hadamard gates (**a**). The oracle marks solution states $|s\rangle$ by introducing a negative phase (**b**). The diffusion operator creates an inversion about the mean (**c**), resulting in higher amplitudes for solution states. The oracle/diffusion process can occur for multiple iterations (**d**) and further increases the probability of measuring solution states (**e**). Grover's algorithm allows for fewer oracle queries than the classical expectation.



Oracle Construction

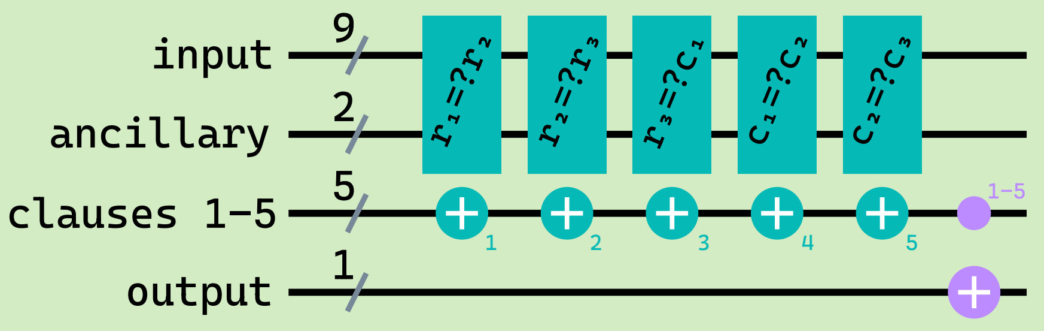


The adder circuit contains two gate types. The CNOT gate flips the value of the output qubit ⊕ if the control qubit ● is in the **on** state. The Toffoli gate flips the output qubit ⊕ only if both control qubits ● ● are **on**. The binary addition is stored in $out_{0/1}$.



By computing the addition of two sums A, B in series with the same output anc, we can check if the two sums are equal since they will cancel each other out.

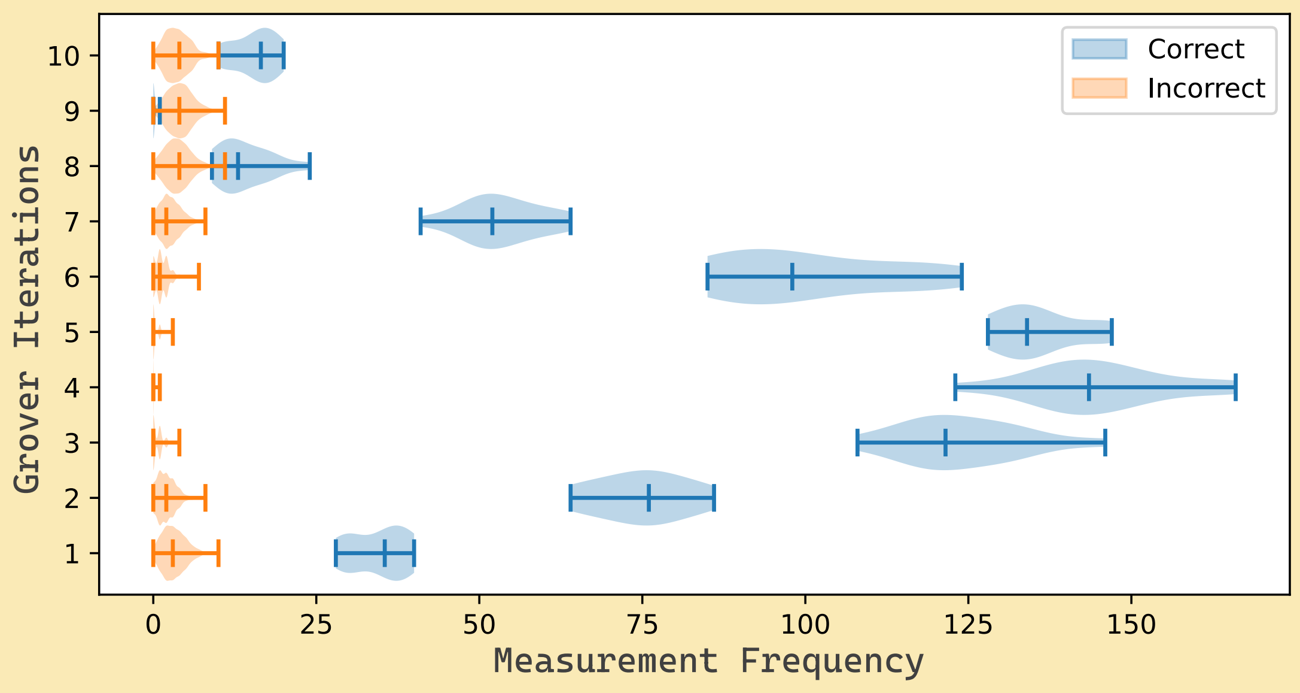
To check if the output has cancelled out, we negate the output X and use a Tofolli gate. We ***uncompute*** the anc qubits by repeating the calculation again.

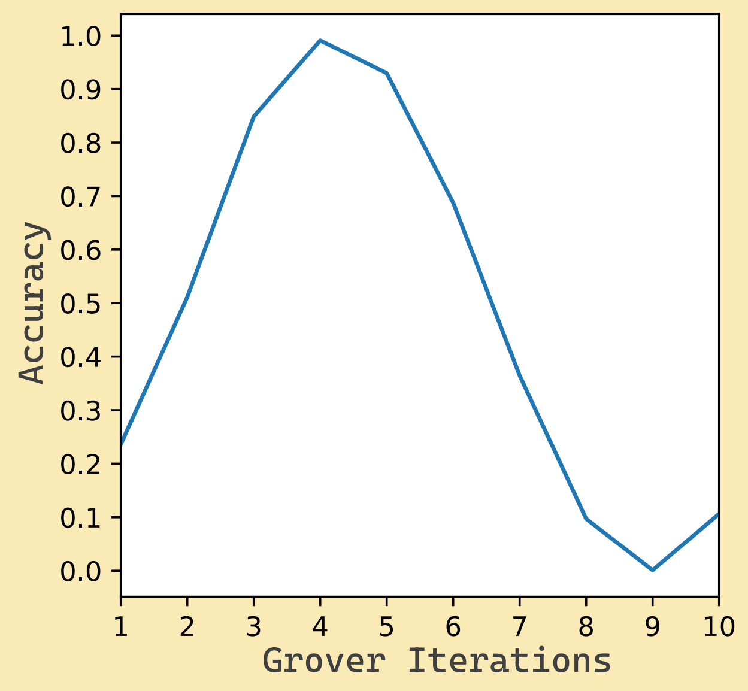


We repeat this equality check for each of the 5 clauses. With uncomputation we are able to reuse the ancillary (anc) qubits. Finally, we need one output qubit, acting as the target of a multi controlled Tofolli gate that verifies the **on** state of all 5 clauses.

Results

We evaluate the suitability of this oracle by running it on a simulator for a variable number of Grover iterations. For each experiment, we run the algorithm for 2048 shots. The solution state is measured at each shot, and it's validity is checked on a classic computer.





It's clear the optimal amount of iterations for our problem is ***4***; for this value we measure correct solution states with an accuracy of ***99.3%***. This agrees with the expected upper bound according to Nielsen and Chuang [5]:

$$\left\lceil \frac{\pi}{4}\sqrt{N/M} \right\rceil = \left\lceil \frac{\pi}{4}\sqrt{2^9/14} \right\rceil = 5$$

where N is the number of possible states and M is the number of valid solution states (which can be calculated through brute force in this case).

Discussion

We find an efficient oracle for the binary semi-magic square problem, and evaluate its effectiveness experimentally.

The ideas presented in this poster could easily be expanded to solve classic magic squares, using more advanced quantum arithmetic such as the Draper adder[6] to compute more complex sums. This would allow finding arbitrary solutions to N by N magic squares. Expanding the circuit like this, we start to reach the limits of current simulators.

This instils confidence in the suitability of this problem for quantum counting, and hence could potentially provide arbitrary precise estimations for the number of solutions for N by N magic squares.

References

[1] Gardner, Martin. "Mathematical games." Scientific American 234.4 (1976): 126-130.

[2] Grover, Lov K. "A fast quantum mechanical algorithm for database search." *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing.* 1996.

[3] Duan, Zhenhua, et al. "Improved even order magic square construction algorithms and their applications in multi-user shared electronic accounts." Theoretical Computer Science 607 (2015): 391-410.

[4] Pinn, Klaus, and Christian Wieczerkowski. "Number of magic squares from parallel tempering Monte Carlo." *International Journal of Modern Physics C* 9.04 (1998): 541-546.

[5] Nielsen, Michael A., and Isaac Chuang. "Quantum computation and quantum information." (2002).

[6] Ruiz-Perez, Lidia, and Juan Carlos Garcia-Escartin. "Quantum arithmetic with the quantum Fourier transform." Quantum Information Processing 16 (2017): 1-14.

University of Birmingham

Lucas Van Mol