12-2019

# Decomposing the Hamiltonian of Quantum Circuits Using Machine Learning

Jordan Burns
*Utah State University*

Yih Sung
*Utah State University*

Colby Wight
*Utah State University*

# Decomposing the Hamiltonian of Quantum Circuits using Machine Learning

Jordan Burns, Yih Sung, Colby Wight

December 2019

## 1 Introduction

Quantum computing is one of the most promising techniques for simulating physical systems that cannot be simulated on classical computers[1]. A significant drawback of this approach is the inherent difficulty in designing circuits that can represent these systems on quantum computers. Every quantum circuit is built out of small components called quantum gates. Each of these gates manipulate the quantum system in a specific way. When used in combination, a finite subset of these gates, the set of universal gates, can be used to construct any possible quantum circuit[2].

The Schrodinger equation states that every quantum system evolves in time according to

$$e^{-iHt}|\psi>$$

where $H$ is the Hamiltonian of the system and $|\psi>$ is the initial state of the system. Because of this every finite quantum system can be described as a finite unitary matrix. Designing a quantum circuit that represents this system is equivalent to decomposing this matrix in terms of the the universal gates (which can also be represented by unitary matrices).

Let $E_1, E_2, ..., E_m$ be the set of universal quantum gates, where $m$ is the total number of gates in the set. Every Hamiltonian can be decomposed by some combination of these gates

$$H = E_{j_1} E_{j_2} ... E_{j_t}$$

Since there is a finite number of options for each $E_j$ in this decomposition the task of finding this combination can be viewed as a tree search problem. Due to the rate at which the size of this tree grows with both the complexity and size of the quantum circuit it is not possible to explore the entire tree using current computers, similarly to how computers are unable to explore the entire search space of other popular tree search problems such as the games of chess and go.

In 2017 DeepMind's AlphaGo [3] program beat the current top ranked human player in a 3 game match of go, a task which seemed impossible using classical techniques. This was achieved by using a combination of a Monte Carlo tree search and a neural network trained off of moves from human go matches.

# 2 Methodology

The task of decomposing the Hamiltonian matrix of a quantum circuit is arranged as follows.

## 2.1 Tree Search

The Hamiltonian matrix that represents the entire system is defined as the target matrix of the search. The root node of the tree is the identity matrix of size $2^n$ where n is the number of qubits in the quantum system. From there every node branches off using each of the possible gates from the set of universal quantum gates.

At each node of the tree the appropriate gate is multiplied with its parent gate to create a new matrix. The relative distance between this matrix and the target matrix by computing the matrix norm of the difference of these matrices

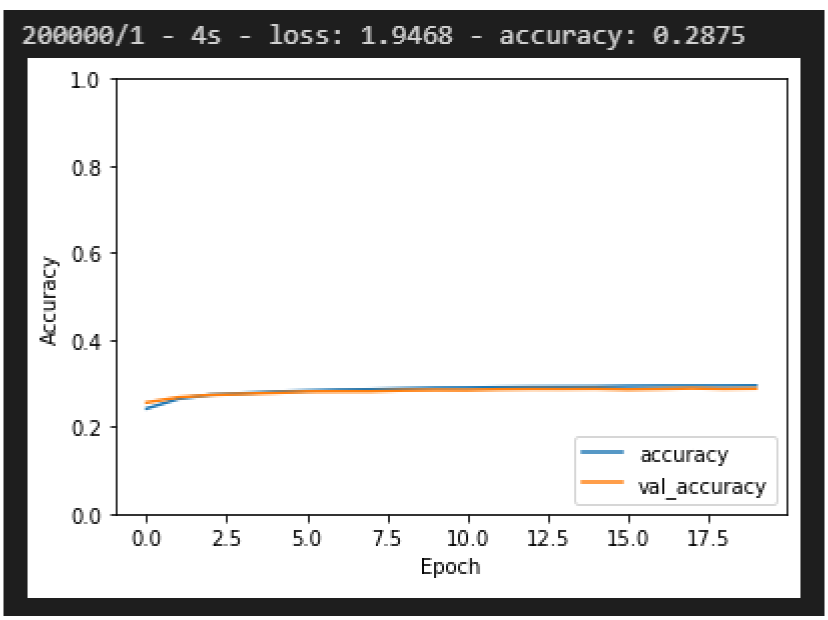
$$||T - C||$$

where $T$ is the target matrix and $C$ is the current matrix. Finding a node where this norm is smaller than some acceptable error is defined as a success for the tree search.

Different portions of this tree are then traversed many times and statistics are collected about whether or not a given path resulted in a success or failure. A standard Monte Carlo tree search [4] would randomly pick a new branch to explore when statistics were not known for the current node. This approach instead uses a neural network trained to pick the correct gate to apply to achieve the target matrix to make decisions when statistics are not known.

## 2.2 Neural Network

The neural network used is built of several convolutional layers using the tensorflow framework[5]. It takes as input a 4x4 complex matrix and outputs the probability that each of the matrices in the set of universal quantum gates is to be the correct gate to be applied next. For 2 qubits there are 9 possible output matrices.




Accuracy of Neural Net

# 3 Results

This technique was applied to several different quantum circuits including randomly generated circuits and the Grover Diffusion Operator from Grover's Algorithm for 2 qubits. The results are given below.

## 3.1 Randomly Generated Gates

This model uses a randomly generated quantum circuit composed of basic quantum gates to test the limits of a standard monte carlo tree search on this problem and possible improvements gained from using a neural network approach.

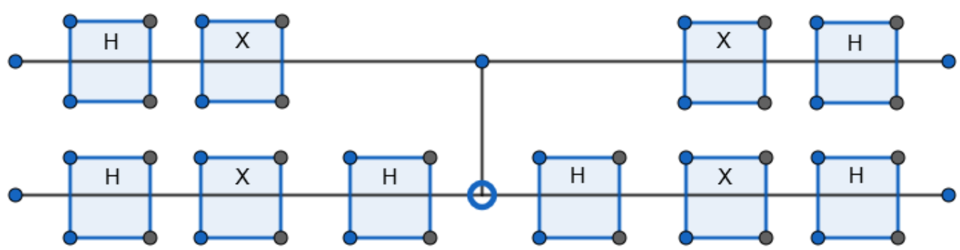
Using standard monte carlo techniques, a desktop computer is able to produce the correct composition a randomly generated circuit made up of up to 4 gates. The addition of another randomly generated gate greatly reduces the accuracy of this approach in decomposing the matrix given the same amount of processing time.

| | Monte Carlo 1 Gate | Neural Net 1 Gate | Monte Carlo 2 Gate | Neural Net 2 Gate | Monte Carlo 3 Gate | Neural Net 3 Gate |
|---|---|---|---|---|---|---|
| Successes | 95 | 100 | 82 | 100 | 75 | 74 |
| Failures | 5 | 0 | 18 | 0 | 25 | 26 |

Using a neural network trained on sample input and output pairs of quantum gates produces the results seen in the above table.

## 3.2 Grover Diffusion Operator

The Grover Search Algorithm [6] is one of the few quantum algorithms that is known to have a computational advantage to comparable classical algorithms. The Grover Diffusion operator is the main section of this algorithm and in a 2 qubit system is built of the following gates.




Applying the standard monte carlo approach to this known circuit had a success rate of 1%, while the neural network assisted monte carlo has a success rate of 4%.

Application of this approach to various section of this operator gave the results in the following table:

| | Monte Carlo 1 Gate | Neural Net 1 Gate | Monte Carlo 2 Gate | Neural Net 2 Gate | Monte Carlo Full Circuit | Neural Net Full Circuit |
|---|---|---|---|---|---|---|
| Successes | 18 | 25 | 11 | 25 | 1 | 4 |
| Failures | 7 | 0 | 14 | 0 | 99 | 96 |

## 4 Summary and Future Goals

This technique can be used to design a circuit based on elementary quantum gates using only a target hamiltonian.. Application of this technique to several small scale systems yielded promising results. Using a trained neural network in combination with a monte carlo tree search decomposed the target hamiltonian more often both in small randomly generated systems as well as the Grover Diffusion Operator.

In future research additional systems could be studied such as Shor's Algorithm and small scale atomic system who's decomposition is not explicitly known. Only one type of neural network was used in this work. A variety of different types of layers as well as additional inputs could be of interest in a farther study. The unsupervised learning approach used in DeepMind's AlphaZero may also be an area of future research [7].

## References

[1] Nielsen, M. A., and Chuang, I. L. (2010). Quantum computation and quantum information. Cambridge: Cambridge University Press.

[2] Barenco, A., Bennett, C. H., Cleve, R., DiVincenzo, D. P., Margolus, N., Shor, P., ... and Weinfurter, H. (1995). Elementary gates for quantum computation. Physical review A, 52(5), 3457.

[3] Silver, D., Huang, A., Maddison, C. et al. Mastering the game of Go with deep neural networks and tree search. Nature 529, 484–489 (2016) doi:10.1038/nature16961

[4] Bradberry, J. (n.d.). Jeff Bradberry. Retrieved from https://jeffbradberry.com/posts/2015/09/intro-to-monte-carlo-tree-search/.

[5] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[6] Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. arXiv preprint quant-ph/9605043.

[7] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... and Lillicrap, T. (2017). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. arXiv preprint arXiv:1712.01815.