Efficient Circuits for Quantum Search over 2D Square Lattice Architecture

Shaohan Hu, Dmitri Maslov, Marco Pistoia, and Jay Gambetta
IBM T. J. Watson Research Center, Yorktown Heights, NY, 10598 USA
shaohan.hu@ibm.com,dmitri.maslov@ibm.com,pistoia@us.ibm.com,jay.gambetta@us.ibm.com

ABSTRACT

Quantum computing has increasingly drawn interest and investments from the academic, industrial, and governmental research communities worldwide. Among quantum algorithms, Quantum Search is important for its quadratic speedup over its classical-computing counterpart. A key ingredient in its implementation is the Multi-Control Toffoli (MCT) gate, which creates a Boolean product of control variables and XORs it into the target. On an idealized quantum computer, all-to-all connectivity would eliminate the need to use SWAP gates to communicate information. This is, however, not affordable in the current Noisy Intermediate-Scale Quantum (NISQ) computing era. In this work, we discuss how to efficiently implement MCT gates on 2D Square Lattices (2DSL), suitable for superconducting circuits, by taking advantage of relative-phase Toffoli gates and H-tree layouts to drastically reduce resulting circuits' depths and the amount of SWAPping required.

KEYWORDS

NISQ, Quantum Computing, Quantum Search, Grover's Algorithm, Multiple-Control Toffoli Gates

1 INTRODUCTION

Today's quantum computing technology is in what is known as Noisy Intermediate-Scale Quantum (or NISQ) computing regime [6], characterized by the following two facts: (i) Gates have limited fidelity, and (ii) The qubits in the computing device can only stay coherent for a limited time, outside of which any information stored in the qubits is destroyed by noise. Therefore, when designing quantum circuits for NISQ devices, it is of crucial importance that the circuits remain small and shallow. In existing prototype quantum computers, hardware connectivity is often limited. As a result, implementations of quantum algorithms that involve arbitrary interactions among qubits require the use of quantum SWAP gates or the alike (e.g., teleportation, but teleportation can be more difficult as it requires technology mating) to bring distant qubits together to enable the desired interactions, thereby increasing the number of gates used as well as the circuit depth.

Quantum Search, also known as Grover's algorithm [2], is a well-known quantum technique for unordered searches. It introduces a quadratic speedup over the best classical strategy. In constructing circuits for Quantum Search, a key component is the Multi-Control Toffoli (MCT) gate, which takes as input n control qubits and a single target qubit, and flips the target state ($|0\rangle \mapsto |1\rangle$ and $|1\rangle \mapsto |0\rangle$) iff all controls are in the $|1\rangle$ state. MCT gates are used in Grover's iterative controlling sequence, and also to generate the oracle. The MCT implementation relies on its decomposition into physically implementable single-qubit gates and two-qubit Controlled-NOT gates, which create pairwise entangling interactions among the control qubits and the target. It is not hard to imagine that on a NISQ processor, the aforementioned pairwise entangling interactions may not be readily available unless additional operations are introduced to SWAP qubit states around. Thus, without care, the implementation of quantum search might lead to circuits with heavy gate-count overheads and high depths, rendering them impractical.

In this work, we provide a strategy for implementing MCT gates in 2D Square Lattice (2DSL), in such a way that the resulting circuits stay shallow and require little to no SWAPping. We decompose n-control MCT into traditional (also commonly denoted as *ccx*) and relative-phase Toffoli gates [4], and take advantage of H-tree [3] layout for qubit placement. Preliminary results show that, compared to the state-of-the-art baseline method [1], our proposed technique leads to drastic *reduction* in the circuit depth while relying on a small, *if any*, SWAP overhead, thus expanding the applicability of Quantum Search on NISQ devices.

2 METHODOLOGY

In this section, we discuss how the key components of our design fit together to implement shallow quantum circuits.

2.1 H-Tree Topology

The textbook method of implementing MCT is to decompose it into traditional 2-control Toffoli gates arranged in a **V** shape [5]; the left arm, with the help of ancillae, gradually builds up partial products and the right arm uncomputes the ancillae. Though straightforward, this approach produces MCT circuits with inefficient depths. In our design, we employ the H-tree layout over the divide-and-conquer approach to implement MCT gates.

An example H-tree is shown in Fig. 1, which is known for efficiently laying out a balanced binary tree in 2DSL. This fits perfectly with the decomposition of MCT into Toffoli gates—for each line segment, if we place two qubits a, b at its endpoints and a single qubit c at its midpoint, we can then place a Toffoli gate $ccx(a, b, c)$ on this line segment. We make the following observations: (i) The actual controls of the MCT correspond to the leaf nodes in the H-tree; (ii) The target

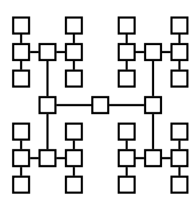

Figure 1: Example H-Tree

DAC '19, June 2–6, 2019, Las Vegas, NV, USA

ACM ISBN 978-1-4503-6725-7/19/06…$15.00
https://doi.org/10.1145/3316781.3322464

of the MCT will be held by the qubit at the center; (iii) All the other nodes are ancillae; and (iv) At higher fractal levels, where a line segment's midpoint is not immediately next to the endpoints, SWAPs or series of CNOTs (also denoted as cx) operations may be required to bring in the endpoints.

Because of the binary-tree structure, any subtree can be evaluated independently, enabling parallel execution and lowering circuit depth from linear **V**-shape to logarithmic. Although this logarithmic depth becomes square-root depth in 2DSL, the improvement from linear to square-root depth is significant.

2.2 Relative-phase Toffolis

We utilize relative-phase Toffoli gates [4] when computing intermediate products. Indeed, only the final Toffoli gate that targets the center qubit needs to be a regular *true* Toffoli; all others can be implemented up to a relative phase.

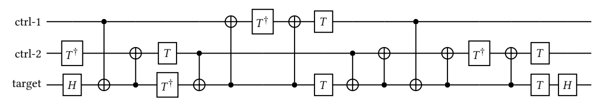

Figure 2: Regular Toffoli

Figs. 2 and 3 introduce the implementations of both the regular and relative-phase Toffoli gates, respectively. As seen, both implementations involve 2-qubit interactions only between the control and the target, which aligns with our intention of midpoint-endpoint interactions only in the H-tree. As relative-phase Toffoli gates use significantly lower number of CNOT gates (3 vs 10), they are much more efficient to implement.

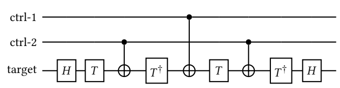

Figure 3: Relative-phase Toffoli [4]

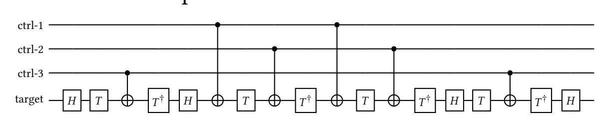

Figure 4: Relative Phase Toffoli with 3 Controls [4]

In addition to the 2-control relative-phase Toffoli (which we denote as *rccx*), a 3-control version (or *rcccx*) also exists, as shown in Fig. 4. With *rcccx*, we have the opportunity to add extra controls to existing H-tree layouts without having to grow the size of the underlying tree. To illustrate this point, Listing 1 shows the H-tree layout for implementing an MCT with 4 controls—the `C`'s, `T`, and the `A`'s correspond to the controls, target, and ancillae, respectively. The two `A`'s first serve as the targets of two parallel *rccx* gates, and then the controls for the final *ccx* that targets `T`.

Listing 1: 4 ctrls

```
C-A-C
  |
  T
  |
C-A-C
```

Listing 2: 6 ctrls

```
C-A-C     C
  |       |
  A---T---A
  |       |
C-A-C     C
```

Listing 3: 6 ctrls w/ *rcccx*

```
  C   C
  |   |
C-A-T-A-C
  |   |
  C   C
```

Given the 4-control MCT, if we wanted to create a 6-control MCT, we could either strictly follow the H-tree layout to grow the fractal structure, say, to its right, as shown in Listing 2, or turn each of the two *rccx* gates into *rcccx* and attach two additional `C`'s to the `A`'s, as illustrated in Listing 3 (note the 90° rotation from Listing 1 for ease of presentation). The benefit of doing this is twofold: we decrease the depth and reduce the footprint of the implementation.

3 RESULTS

We next apply the techniques developed in Section 2 to construct circuits implementing MCT gates in 2DSL for various numbers of controls. Listings 4 through 9 report the layouts. Interestingly, and perhaps unexpectedly, *MCTs with up to 12 controls can be implemented in 2DSL with minimized depth and without needing SWAPs.*

Listing 4: 7 ctrls

```
C
|
A-C C
|   |
A-T-A-C
|   |
A-C C
|
C
```

Listing 5: 8 ctrls

```
  C
  |
C-A   C
  |   |
C-A-T-A-C
  |   |
C-A   C
  |
  C
```

Listing 6: 9 ctrls

```
  C
  |
C-A-C C
  |   |
  A-T-A-C
  |   |
C-A-C C
  |
  C
```

Listing 7: 10 ctrls

```
  C   C
  |   |
C-A-C A-C
  |   |
  A-T-A
  |   |
C-A C-A-C
  |   |
  C   C
```

Listing 8: 11 ctrls

```
  C   C
  |   |
C-A-C A-C
  |   |
  A-T-A-C
  |   |
C-A C-A-C
  |   |
  C   C
```

Listing 9: 12 ctrls

```
  C   C
  |   |
C-A-C A-C
  |   |
C-A-T-A-C
  |   |
C-A C-A-C
  |   |
  C   C
```

Below, we report the results for MCT circuit-depth comparisons on a 5×5 2DSL. For baseline, we employ the textbook linear **V**-shape structure and the 6-CNOT Toffoli implementation [5]. All qubits are placed on the 2DSL in a top-to-bottom and left-to-right manner. All MCT circuits are generated and transpiled using Qiskit [1]. All circuits are unrolled to the basis $u1$, $u2$, $u3$, and cx gates, and depths are counted with cx gates only. We use Qiskit's `StochasticSwap` mapper to map each unrolled circuit to the 2DSL topology, repeated for 100 trials. As follows from Fig. 5, our method leads to circuit depths that are only a fraction of those in the baseline.

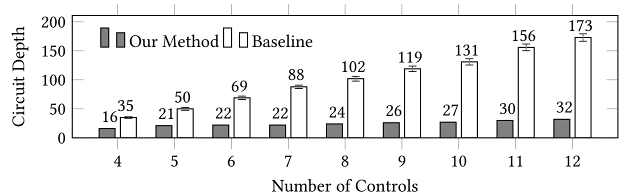

Figure 5: MCT Circuit Depths Comparisons

4 CONCLUSION & DISCUSSION

In this work, we discussed how to combine relative-phase Toffoli gates with H-tree layout to construct efficient MCT circuits for NISQ devices with 2DSL architecture. The results show drastic improvement over the baseline.

Since Quantum Search heavily relies on the MCT gates and admits further opportunities for optimization—for example, the oracle structure can be exploited such that MCTs' partial products can be reused before fully uncomputing them—our proposed technique shows promise for additional reductions.

REFERENCES

[1] Gadi Aleksandrowicz, Thomas Alexander, et al. 2019. Qiskit: An Open-source Framework for Quantum Computing. https://doi.org/10.5281/zenodo.2562110
[2] Lov K Grover. 1996. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing.* ACM, 212–219.
[3] Charles E Leiserson. 1980. Area-efficient graph layouts. In *21st Annual Symposium on Foundations of Computer Science (sfcs 1980).* IEEE.
[4] Dmitri Maslov. 2016. Advantages of using relative-phase Toffoli gates with an application to multiple control Toffoli optimization. *Physical Review A* 93, 2 (2016), 022311.
[5] Michael A Nielsen and Isaac Chuang. 2002. Quantum computation and quantum information.
[6] John Preskill. 2018. Quantum Computing in the NISQ era and beyond. *Quantum* 2 (2018), 79.