# Orchestrated Trios: Compiling for Efficient Communication in Quantum Programs with 3-Qubit Gates

**Extended Abstract**

Casey Duckering, Jonathan M. Baker, Andrew Litteken, Frederic T. Chong
*University of Chicago Department of Computer Science*

## 1. Motivation

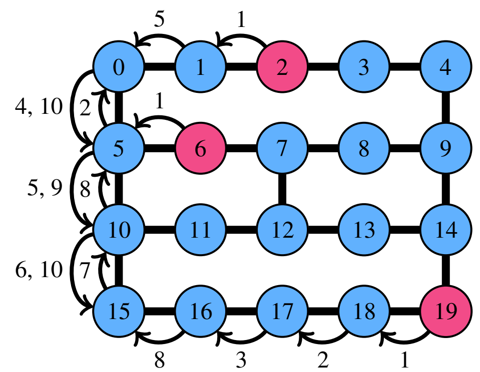
Current quantum computers are especially error prone and require high levels of optimization to reduce operation counts and maximize the probability the compiled program will succeed. These computers do not natively support the execution of complex multi-qubit operations, instead requiring them be decomposed into simpler one- and two-qubit gates. Furthermore, these computers often have limited hardware connectivity designating between which pairs of hardware qubits gates may be executed between. In order to use distant qubits then they require extra operations be added to move, or route, the qubits around the device. This routing stage of compilation has been demonstrated to add a large number of operations, often many times more than the number of operations in the input program. Because success rate is correlated with total number of operations, it is critical to minimize the total operations added in this way to maximize the probability of program success and obtain the correct answer.

## 2. Limitations of the State of the Art

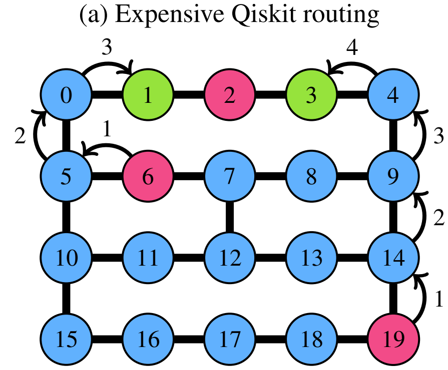
Typical compilers solve the routing problem only after lowering the input program to primitive operations. For example, Qiskit's compiler, the state of the art for compiling to IBM's hardware, first decomposes all gates to a suitable ISA before routing begins. This has one major pitfall: the compiler can no longer take advantage of known local-global structure. For example, many quantum programs use the Toffoli gate, a three-qubit gate, which will be decomposed into 6 CNOTs (a common two qubit gate). State of the art routing policies will attempt to route each of these CNOTs individually which can result in excessive SWAPs (routing operations composed of three CNOTs) as qubits are routed back-and-forth when each CNOT needs them. Figure 1a shows an extreme example of this.

## 3. Key Insights

Current quantum compilers cannot currently account for more complex structures when routing on devices with limited connectivity. The de facto strategy is to decompose all multi-qubit gates to one- and two-qubit gates and perform routing afterwards. Our approach performs some decomposition before *and after* routing in order to retain a program structure when it can be best exploited. By stopping decomposition at Toffoli gates and directly routing the Toffolis, we can avoid adding excessive SWAPs due to routing for each part of the Toffoli individually. An example is shown in 1. A second round of decomposition can then be performed to obtain hardware executable gates from each Toffoli. Because this final decomposition occurs after routing, the best decompositions can be selected given each Toffoli's location on the hardware, further improving program success probability.



(a) Expensive Qiskit routing



(b) Efficient Trios routing

**Figure 1: Example routing from Qiskit (a) vs. Trios (b) for a single Toffoli operation. Circles represent qubits and lines indicate two qubits are connected. Input qubits are highlighted in red. SWAP arrows are labeled by timestep. By reducing SWAPs and program duration, the probability of program success is greatly improved.**

## 4. Main Artifacts

We present Orchestrated Trios, a new quantum compiler structure which performs circuit decomposition, in multiple steps to better take advantage of program structure; the new pass

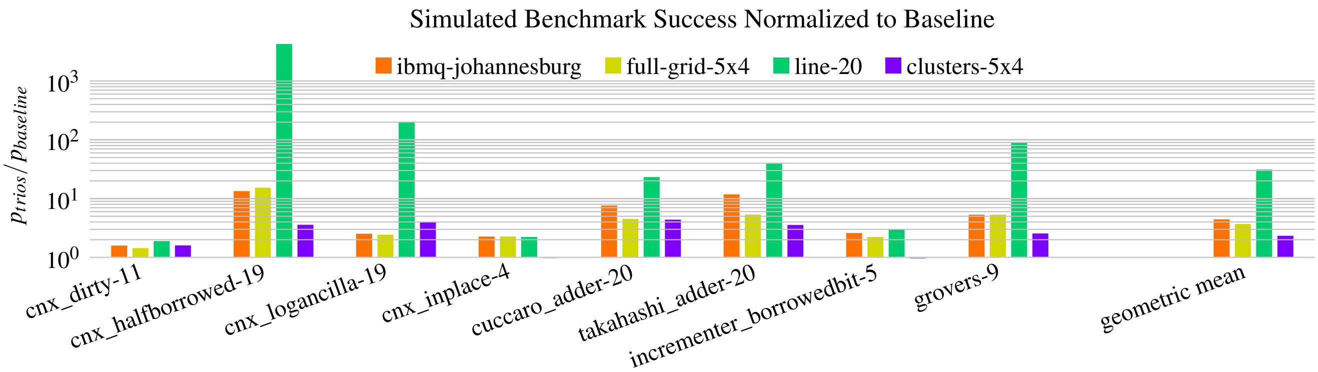



**Figure 2: Normalized success rate on simulated benchmarks with Trios on various hardware topologies. Above $10^0$ indicates benefit. Some improvement factors are huge due to near-zero baseline success rates. The geometric mean increases in success rate are 4.4x, 3.7x, 31x, and 2.3x respectively.**

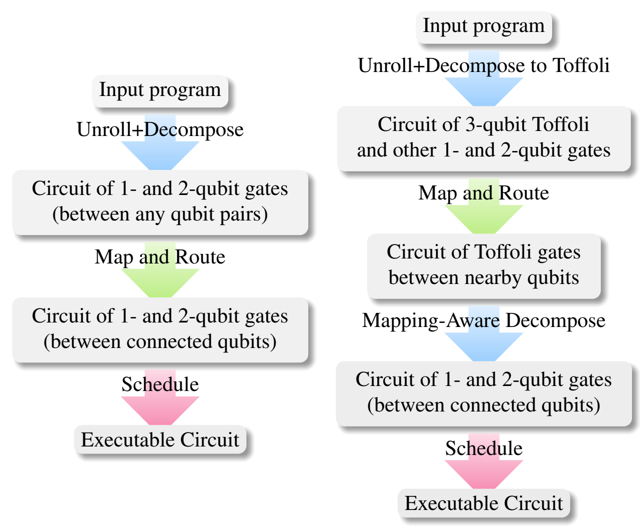
structure is found in Figure 3. Trios targets routing three qubit gates like the Toffoli, a ubiquitous mid-level operation used in many quantum benchmarks. We also show how this new structure enables the dynamic selection of Toffoli decompositions based on the qubit connectivity known after mapping, for example choosing between the standard decomposition requiring all three qubits to be pairwise connected and a decomposition requiring only qubits connected in a chain. We build a compiler to evaluate realistic near-term quantum benchmarks in simulation and also build on the Qiskit compiler to show the benefit of our method in real hardware experiments on IBM Johannesburg, a 20 qubit superconducting transmon device.

## 5. Key Results and Contributions

We make the following contributions:

- A new compiler structure, Trios, with two passes for decomposition with a modified routing pass in between which greatly improves qubit routing.
- A simple method for architecture-tuned Toffoli decompositions during the second decompose pass.
- On Toffoli-only experiments, Trios reduces the total number of gates by 35% geomean (geometric mean) resulting in 23% geomean increase in success rate when run on real IBM hardware as compared to Qiskit.
- On near-term algorithms shown in Figure 2 (4 to 20 qubit benchmarks), Trios reduces total gate count by 37% geomean resulting in 344% geomean increase in (or 4.44x) simulated success rate on IBM Johannesburg with noise rates of near-future hardware as compared to programs compiled without Trios. A sensitivity analysis over four architecture types shows the benefit range from 133% to 3020% increase in success rate.

These massive advantages derive from better routing of complex multi-qubit gates on connectivity-limited devices. By retaining information about global program structure, Trios is able to reduce overhead substantially when compared to the state of the art.




(a) Conventional compilation (b) Trios compilation

**Figure 3: Typical compilation passes (simplified).**

## 6. Why ASPLOS

This work sits at the intersection of architecture and compilers, restructuring the quantum compiler passes to take advantage of an architectural optimization involving 3-qubit operations. The work fits well into the interdisciplinary emphasis of ASPLOS, evaluating the proposed systems in terms of error models and applications in the quantum domain. We hope to inspire those with a systems perspective to help address similar problems in the effort to realize practical quantum computation.

## 7. Citation for Most Influential Paper Award

For demonstrating a multi-layered compiler paradigm that enabled optimization for higher-level quantum operations, with specific application to substantially reducing communication for ubiquitous 3-qubit Toffoli gates. This led to a change in how quantum compilers were structured and helped bridge the gap between practical applications and noisy quantum hardware.