# Arline: Benchmarks

Automated benchmarking platform for quantum compilers, quantum hardware and quantum algorithms

**Automatically generated report**

**by Arline Benchmarks**

https://github.com/ArlineQ/arline_benchmarks

June 15, 2022

# Contents

# Chapter 1

# Overview

Quantum compilation is a problem of translating quantum algorithm to the set of low-level hardware instructions to be executed on the quantum processor. Efficient compilation and circuit optimisation (finding an optimal sequence of gates for the desired quantum computation) is immense importance for practical applications and is necessary for further progress towards scalable quantum computation.

The importance of optimal quantum compilation stems from the fact that noisy intermediate-scale quantum (NISQ) devices suffer from unavoidable noise caused by individual gates. Extreme susceptibility of quantum computation to noise is the crucial problem that hinders the development of large-scale quantum computers. By the means of optimising gate count in the quantum circuit, it is possible to significantly reduce hardware errors.

Optimal (or near-optimal) circuit compilation is an extremely challenging task due to additional constraints imposed by hardware configuration, such as restricted qubit connectivity and hardware-native gate set. Finding optimal gate sequences for a given quantum circuit with all imposed constraints is an open problem. For two-qubit circuits, the KAK algorithm is an example of the efficient compilation algorithm, that is used in practical applications.

The subroutines for quantum circuit optimisation, mapping and routing for particular hardware connectivity are an integral part of existing quantum software frameworks. The complexity and diversity of various quantum compilation algorithms create a necessity of cross-benchmarking and comparison between different compiler frameworks. It worth to note, that the problem of hardware benchmarking often arises in classical computing, where hardware is compared based on their performance on a set of predefined tests.

Arline Benchmarks[1] platform is created to solve benchmarking problem in the quantum world and aims to provide a fair comparison between compilers for various quantum hardware and quantum algorithms.

## 1.1 Frameworks

Below we list compilation frameworks used in the benchmarking run:

- **IBM Qiskit**[2]v0.34.2 (open-source).**Qiskit** is an open-source framework for working with quantum computers at the level of circuits, pulses, and algorithms. Qiskit transpiler combines mapping and compression subroutines. The transpiler has three levels of optimization, in our report, we invoke the most advanced optimization level (3).
  - Compression/optimization algorithm relies on **commutative cancellations of gates, aggregation of single-qubit gates, removal of diagonal gates before measurement.**
  - Routing and mapping algorithms are based on construction of a **basic, stochastic or lookahead swap network.**
  - The output circuit gate set: $U_1$, $U_2$, $U_3$, $CNOT$, $I$.
- **Google Cirq library** v0.13.1 (open-source). **Cirq** has been developed by Google AI Quantum Team. Cirq supports mapping/routing operations and tailored towards specific grid-like qubit coupling topologies.
  - The compression subroutines include **pushing Pauli gates and phased Pauli gates to the end of the circuit, merging single-qubit gates and dropping negligible gates and empty moments**.
  - Routing and mapping methods are based on a **greedy swap insertion strategy.**
  - The output circuit gate set: $R_x$, $R_z$, $CZ$.

[1]**Disclaimer.** This report was prepared using open source program Arline Benchmarks and is provided "AS IS", "AS AVAILABLE", and all warranties, express or implied, are disclaimed.

Neither the authors of the Arline Benchmarks program, nor any of their employees, nor any of their contractors, subcontractors or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or any third party's use or the results of such use of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

Readers are cautioned that information contained in this report is only current as of the respective dates of such information. The technical condition, analyzed program code, comparative results may have changed since those dates. Readers are advised to review only the most recent document for the most current information.

[2]IBM and Qiskit are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

## 1.2 Definitions

We define compression factor ($CF$) for a particular class of gates ($G$) (e.g. single-qubit gates, two-qubit gates) averaged over circuits ($C$) as:

- Compression factor greater then unity ($CF(G) > 1$) corresponds to a successful compression (the gate count of the gate ($G$) in the output circuit is less compared to the input circuit). Similarly, compression factor less then unity ($CF(G) < 1$) corresponds to unsuccessful compression (the output circuit contains more gates of type ($G$) compared to the input circuit).

$$CF(G) = \left\langle \frac{\text{gate count}(G, C_i)_{\text{before}}}{\text{gate count}(G, C_i)_{\text{after}}} \right\rangle_C .$$

- Circuits **before** and **after** correspond to initial and final compilation stages. In the present report, we usually assume that the initial stage is target generation, the final stage is typically either **compression** or **rebase** to the output hardware gate set.

As an additional metric that defines a circuit cost we use the following circuit cost function:

$$C = -\log\left(K^d \prod_i F_i^{1q} \prod_j F_j^{2q}\right),$$

where $C$ - circuit cost, $K$ - factor that penalises deep circuits, $d$ - circuit depth, $F_i^{1q}$ - fidelity of single qubit gates, $F_j^{2q}$ - fidelity of two qubit gates.

## 1.3 Metrics

- Depth
- Total Gate Count
- Single-Qubit Gate Count
- Two-Qubit Gate Count
- Single-Qubit Gate Depth
- Two-Qubit Gate Depth
- Circuit Cost Function
- Execution Time

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance. This is because single-qubit gate count is very sensitive to the choice single-qubit basisgates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

## 1.4 Hardware Backends

Hardware backends are defined in open-source **Arline Quantum** library https://github.com/ArlineQ/arline_quantum. Names of hardware backends reflect gate set and connectivity. E.g. IbmAll2All16Q corresponds to a 16-qubit fully-connected mock backend with the native gate set of the IBM quantum hardware.

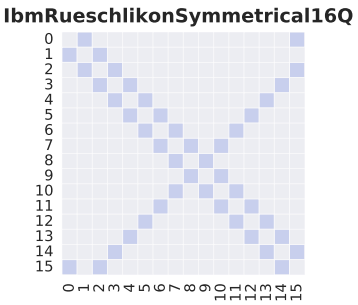


Figure 1.1: IbmRueschlikonSymmetrical16Q hardware coupling map: adjacency matrix.

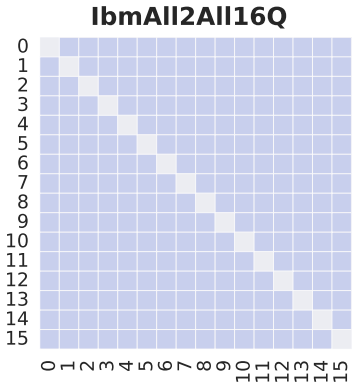


Figure 1.2: IbmAll2All16Q hardware coupling map: adjacency matrix.

## 1.5 Compilation Pipelines

**Compilation pipeline** is a sequence of compilation routines, which typically consist of three main stages:

1. **Unroll** (optional) for the three-qubit gates;
2. **Mapping** and **routing** of the original circuit to a hardware topology, further **mapping**;
3. **Compression** of the circuit that reduces the number of gates involved;
4. **Rebase** (optional) of the final gate sequence into a particular gate set;

Summary of compilation pipelines settings used in the current report is presented below.
Number of pipelines for benchmarking: **2**

- Pipeline name: **QiskitPl**
    1. Stage name: **mapping_compression**
- Pipeline name: **CirqPl**
    1. Stage name: **mapping_compression**

## 1.6 System Info

- Platform: Linux-5.13.0-1029-azure-x86_64-with-glibc2.29
- Processor: AMD EPYC 7763 64-Core Processor
- Memory: 15.6 Gb

## 1.7 Targets

Target is a quantum circuit subject to compilation. List of target generators:

- Target generator type: random circuits from [Clifford + T] gate set:
    - Number of circuits: 10
    - Gates frequencies: [Cnot: 0.152, H: 0.17, S: 0.161, Sd: 0.168, T: 0.172, Td: 0.177]
    - Mean number of qubits: 2.0
    - Mean circuit depth per circuit: 84.7
    - Mean single-qubit gate count per circuit: 101.7

  - Mean two-qubit gate count per circuit: 18.3
  - Mean total gate count per circuit: 120.0

- Target generator type: random circuits from [CNOT, $U_3$] gate set:
  - Number of circuits: 10
  - Gates frequencies: [Cnot: 0.515, U3: 0.485]
  - Mean number of qubits: 2.0
  - Mean circuit depth per circuit: 108.0
  - Mean single-qubit gate count per circuit: 58.2
  - Mean two-qubit gate count per circuit: 61.8
  - Mean total gate count per circuit: 120.0

- Target generator type: random circuits from [Clifford + T] gate set:
  - Number of circuits: 10
  - Gates frequencies: [Cnot: 0.743, H: 0.061, S: 0.045, Sd: 0.049, T: 0.054, Td: 0.048]
  - Mean number of qubits: 16.0
  - Mean circuit depth per circuit: 33.8
  - Mean single-qubit gate count per circuit: 30.9
  - Mean two-qubit gate count per circuit: 89.1
  - Mean total gate count per circuit: 120.0

- Target generator type: random circuits from [CNOT, $U_3$] gate set:
  - Number of circuits: 10
  - Gates frequencies: [Cnot: 0.926, U3: 0.074]
  - Mean number of qubits: 16.0
  - Mean circuit depth per circuit: 36.9
  - Mean single-qubit gate count per circuit: 8.9
  - Mean two-qubit gate count per circuit: 111.1
  - Mean total gate count per circuit: 120.0

- Target generator type: QASM circuits for arithmetic blocks:
  - Number of circuits: 5

- Gates frequencies: [Cnot: 0.445, H: 0.123, T: 0.246, Td: 0.185, X: 0.001]
- Mean number of qubits: 16.0
- Mean circuit depth per circuit: 165.2
- Mean single-qubit gate count per circuit: 184.7
- Mean two-qubit gate count per circuit: 148.2
- Mean total gate count per circuit: 332.8

# Chapter 2

# Comparison with KAK Decomposition: Two-Qubit Circuits

## 2.1 Cartan decomposition

Cartan decomposition (KAK decomposition) provides theoretical $CNOT$-optimal schemes for two-qubit circuits (Figure **??**).

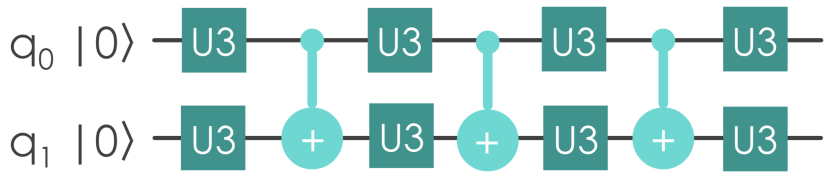


Figure 2.1: KAK decomposition scheme for two-qubit circuits.

Arbitrary two-qubit unitary can be represented using $U_3$ and $CNOT$ gates with no more than:

| **Metrics** | **Gate Count** | **Depth** |
|---|---|---|
| $CNOT$ | 3 | 3 |
| $U_3$ | 8 | 4 |
| Total | 11 | 7 |

# Chapter 3

# Target: Random Circuits from [Clifford + T] Gate Set

## 3.1 Hardware: IbmAll2All2Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

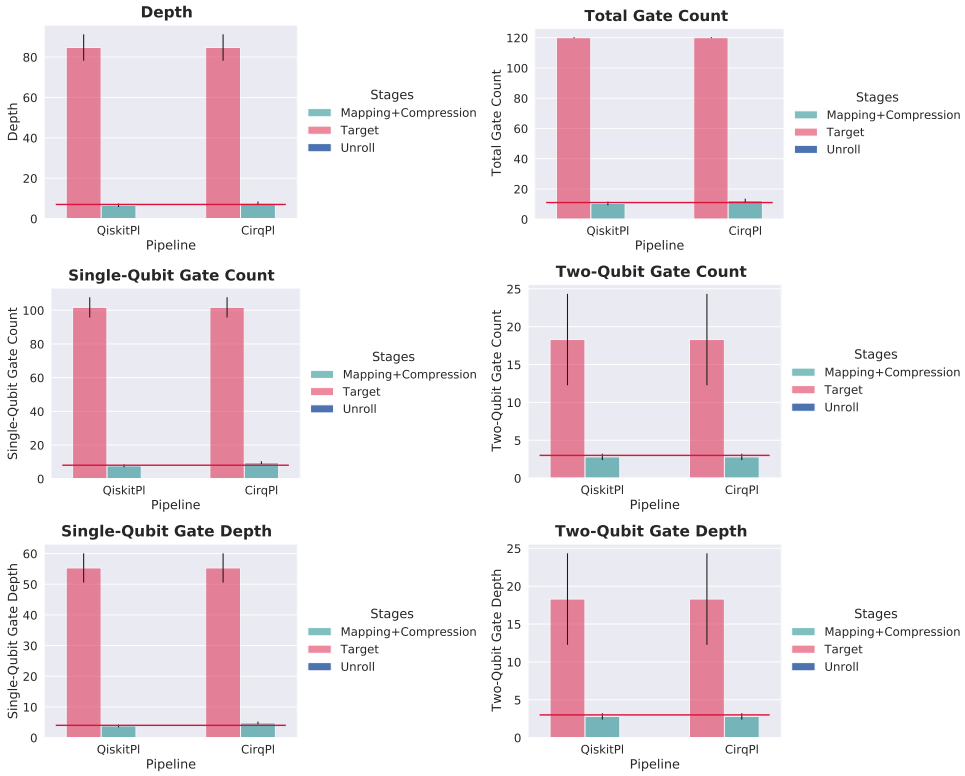


Figure 3.1: Circuits metrics for each compilation pipeline stage for IbmAll2All2Q.

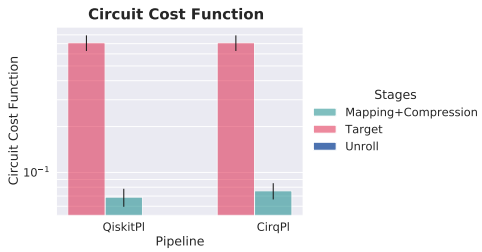


Figure 3.2: Circuit cost function for IbmAll2All2Q.

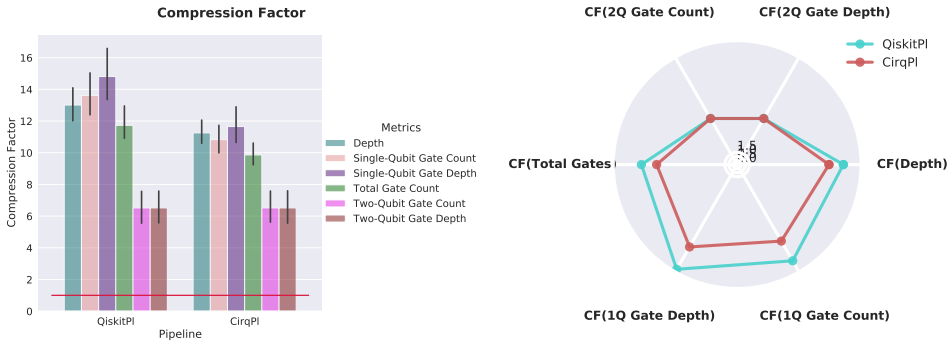


Figure 3.3: Compression factor ($CF$) between target and final compilation stage for IbmAll2All2Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

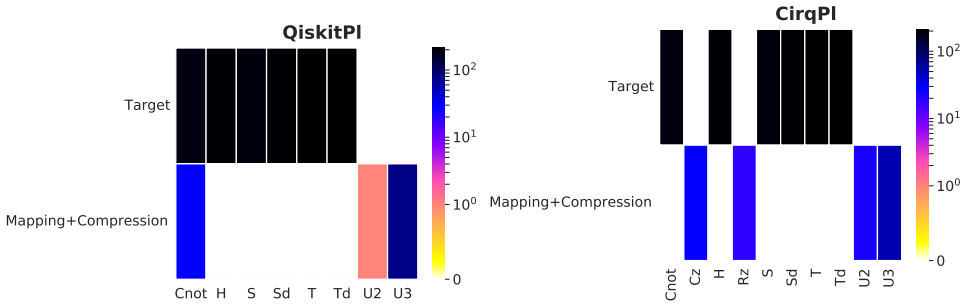


Figure 3.4: Gate frequencies in each pipeline stage for IbmAll2All2Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

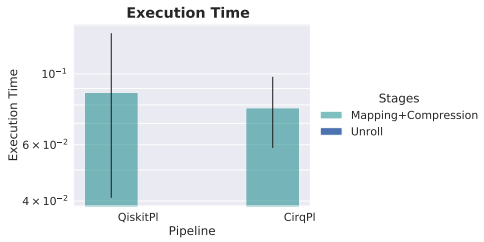


Figure 3.5: Mean execution time of each compilation stage for IbmAll2All2Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | CirqPl | CirqPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| Metrics | Min | Max |
|---|---|---|
| Two-Qubit Gate Depth | CirqPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | CirqPl | QiskitPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

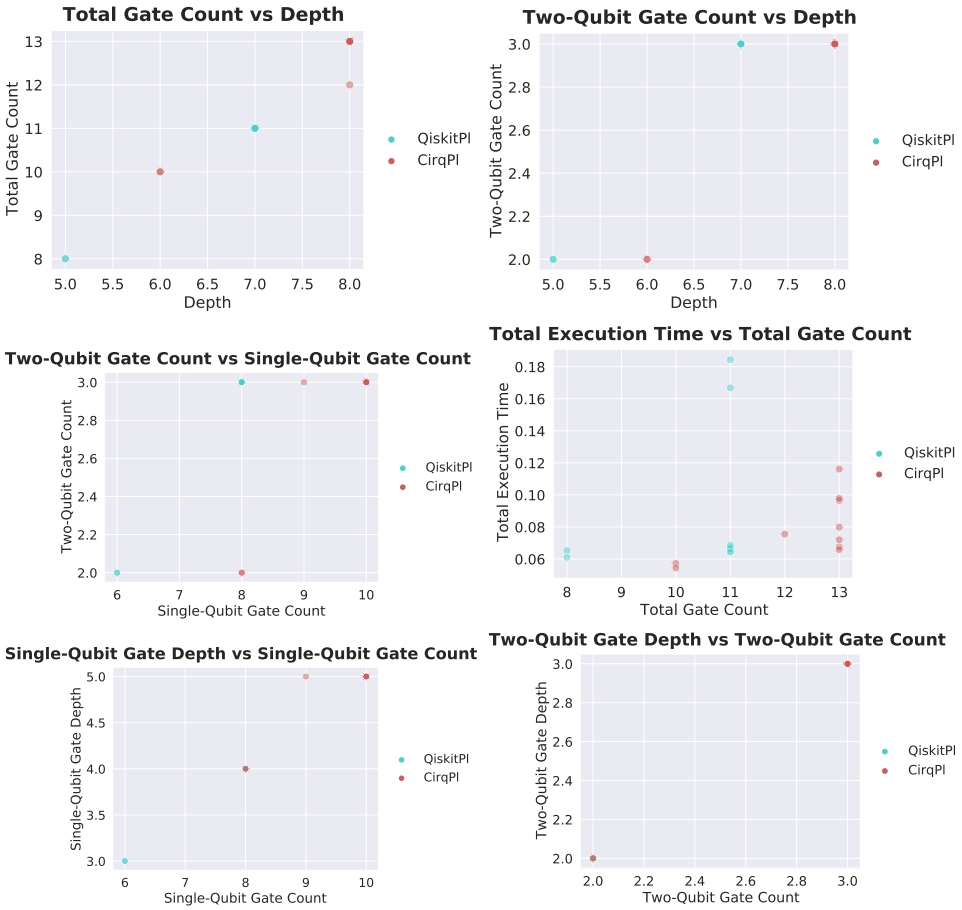


Figure 3.6: Cluster analytics for IbmAll2All2Q. Each point corresponds to an individual target quantum circuit from the target generator.

### Breakdown by individual circuits

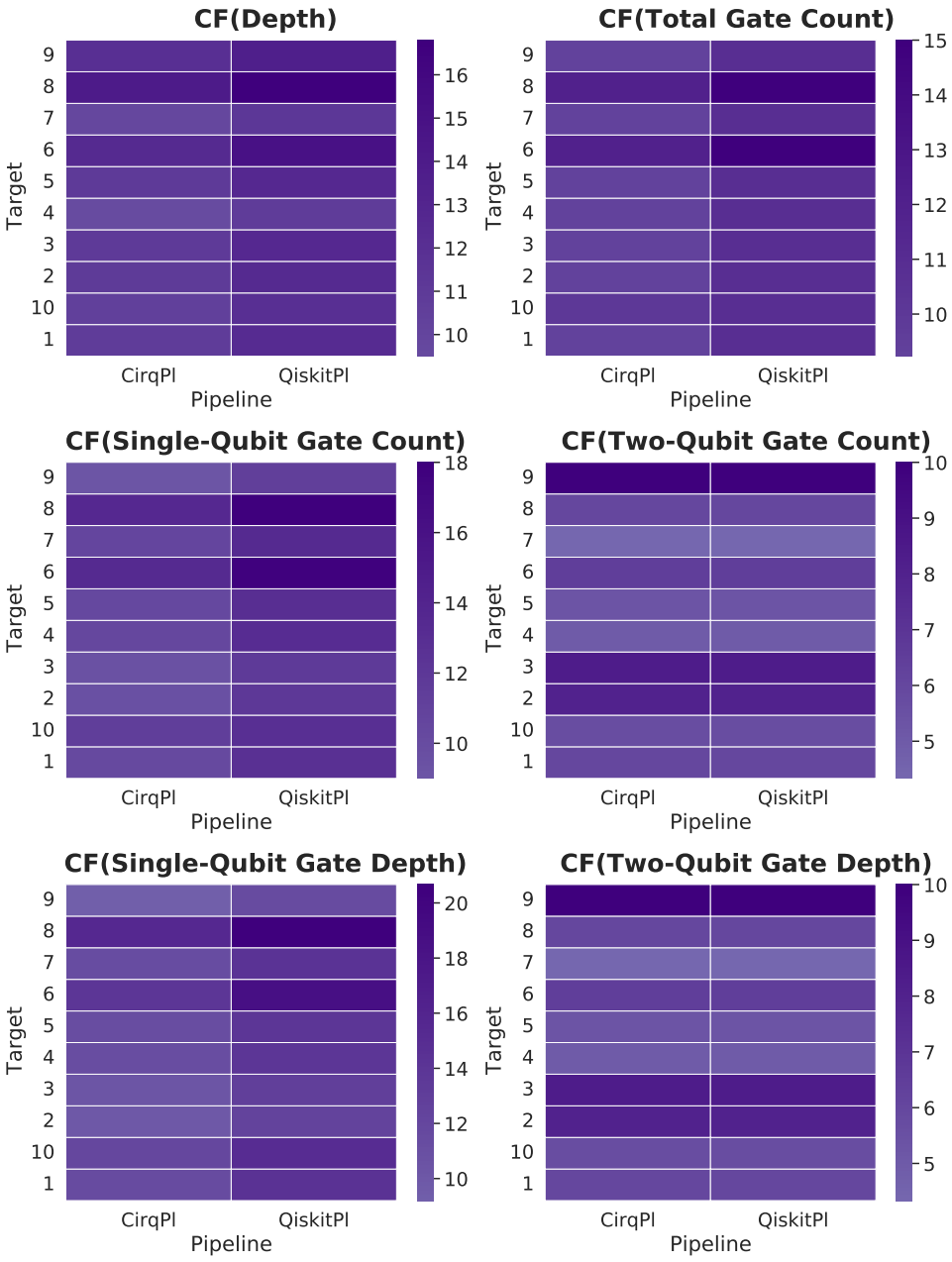


Figure 3.7: Compression factor ($CF$) vs circuit for IbmAll2All2Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | 6 | 5 | CirqPl | 1 | 8 |
| Total Gate Count | QiskitPl | 6 | 8 | CirqPl | 1 | 13 |
| Single-Qubit Gate Count | QiskitPl | 6 | 6 | CirqPl | 1 | 10 |
| Two-Qubit Gate Count | QiskitPl | 6 | 2 | QiskitPl | 1 | 3 |
| Single-Qubit Gate Depth | QiskitPl | 6 | 3 | CirqPl | 1 | 5 |
| Two-Qubit Gate Depth | QiskitPl | 6 | 2 | QiskitPl | 1 | 3 |
| Circuit Cost Function | QiskitPl | 6 | 0.051 | CirqPl | 1 | 0.08 |
| Execution Time | CirqPl | 8 | 0.054 | QiskitPl | 10 | 0.184 |

## 3.2 Hardware Comparison

This analysis is performed across all classes of target circuit.

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Depth | IbmAll2All2Q | IbmAll2All2Q |
| Total Gate Count | IbmAll2All2Q | IbmAll2All2Q |
| Single-Qubit Gate Count | IbmAll2All2Q | IbmAll2All2Q |
| Two-Qubit Gate Count | IbmAll2All2Q | IbmAll2All2Q |
| Single-Qubit Gate Depth | IbmAll2All2Q | IbmAll2All2Q |
| Two-Qubit Gate Depth | IbmAll2All2Q | IbmAll2All2Q |
| Circuit Cost Function | IbmAll2All2Q | IbmAll2All2Q |
| Execution Time | IbmAll2All2Q | IbmAll2All2Q |

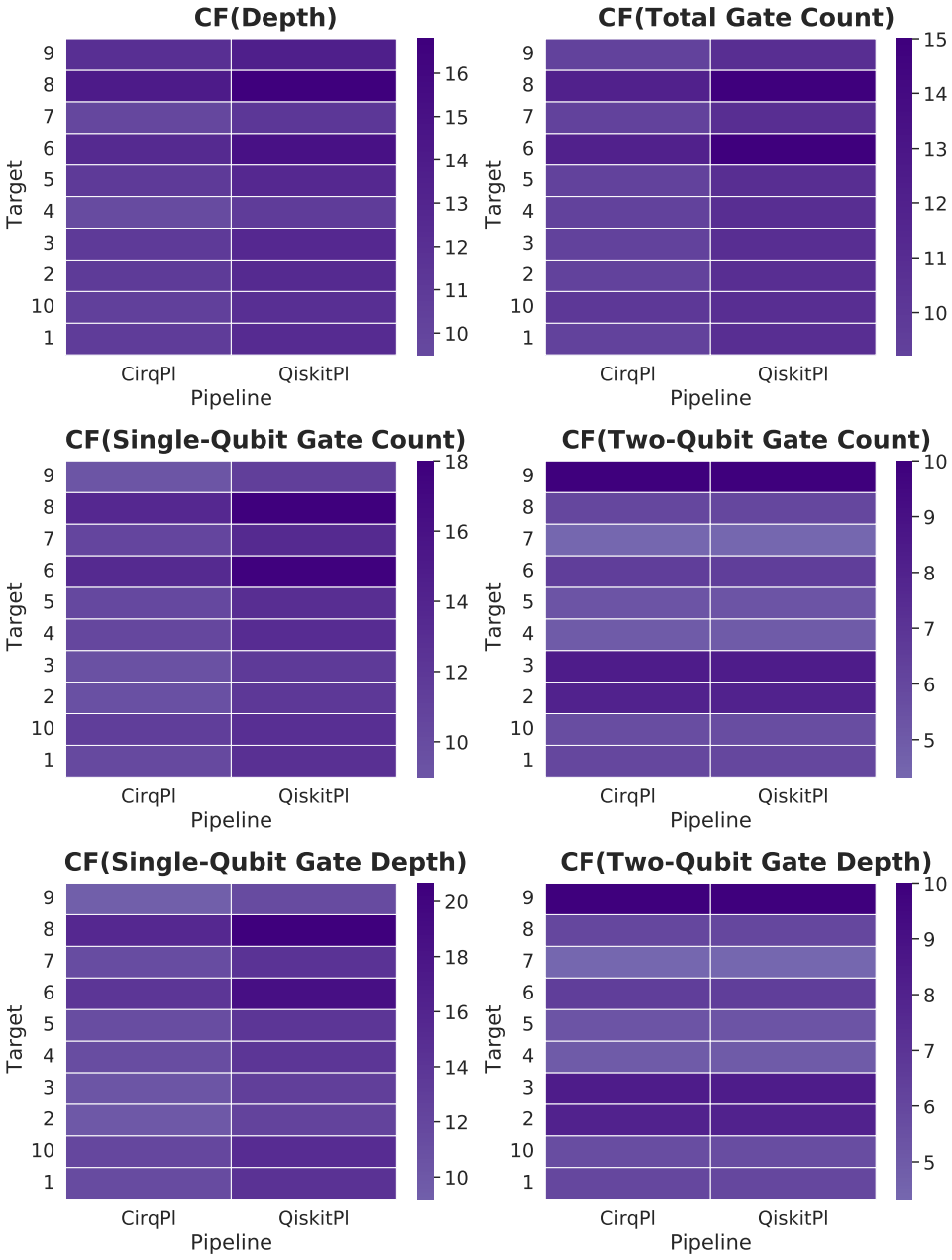


Figure 3.8: Final circuit metrics after compilation vs hardware backend.

# Chapter 4

# Target: Random Circuits from [CNOT, $U_3$] Gate Set

## 4.1 Hardware: IbmAll2All2Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

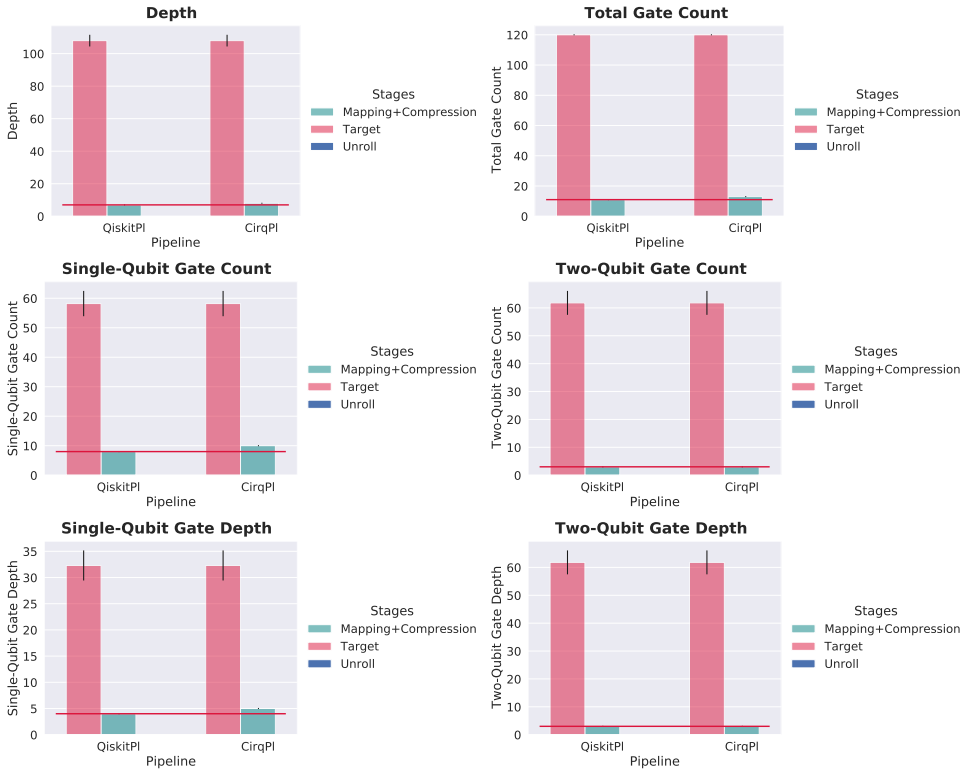


Figure 4.1: Circuits metrics for each compilation pipeline stage for IbmAll2All2Q.

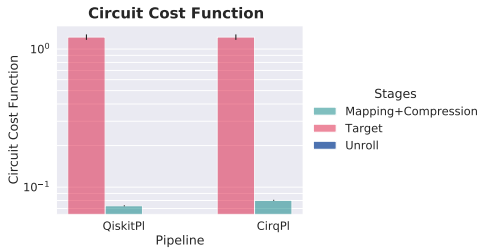


Figure 4.2: Circuit cost function for IbmAll2All2Q.

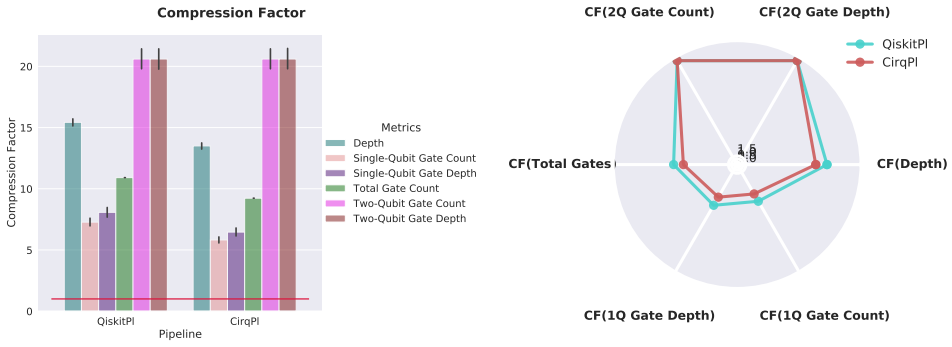


Figure 4.3: Compression factor ($CF$) between target and final compilation stage for IbmAll2All2Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

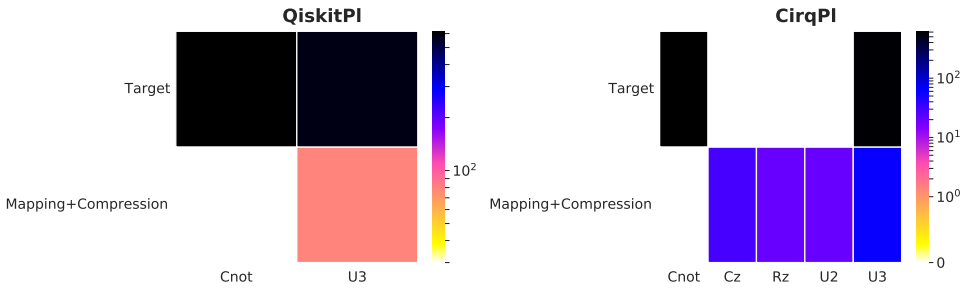


Figure 4.4: Gate frequencies in each pipeline stage for IbmAll2All2Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

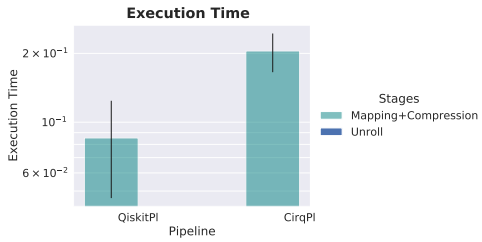


Figure 4.5: Mean execution time of each compilation stage for IbmAll2All2Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | CirqPl | CirqPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Two-Qubit Gate Depth | CirqPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

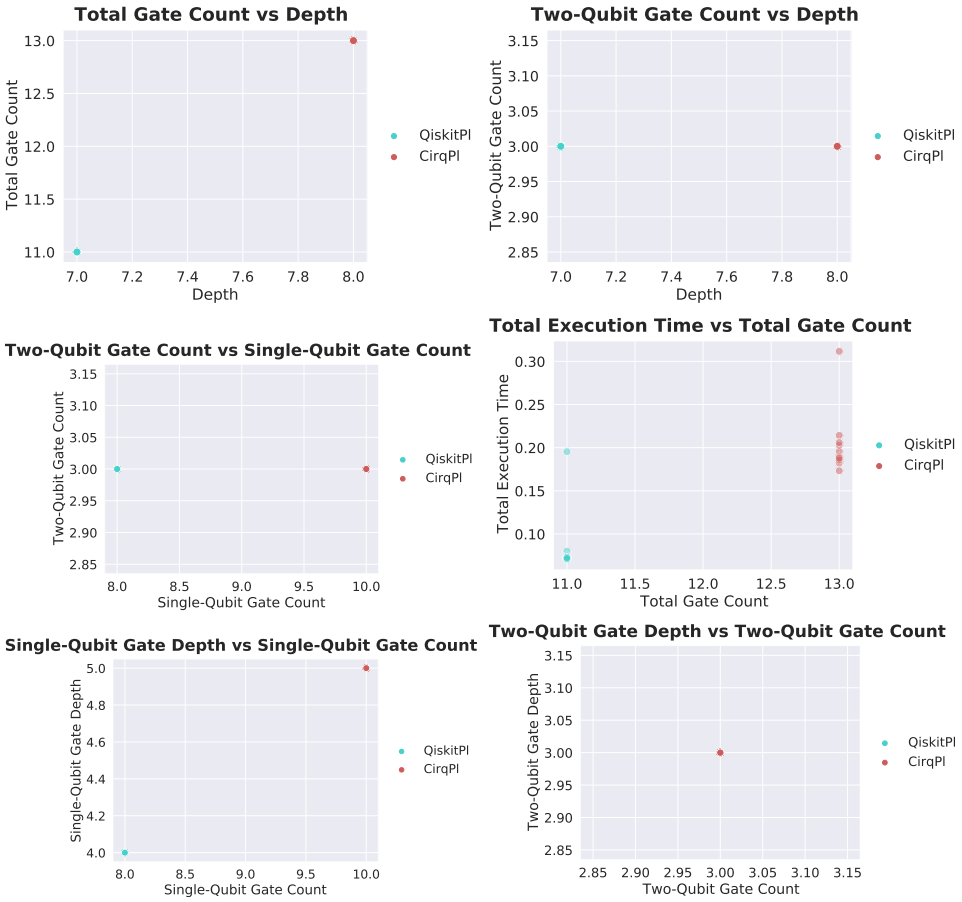


Figure 4.6: Cluster analytics for IbmAll2All2Q. Each point corresponds to an individual target quantum circuit from the target generator.

## Breakdown by individual circuits

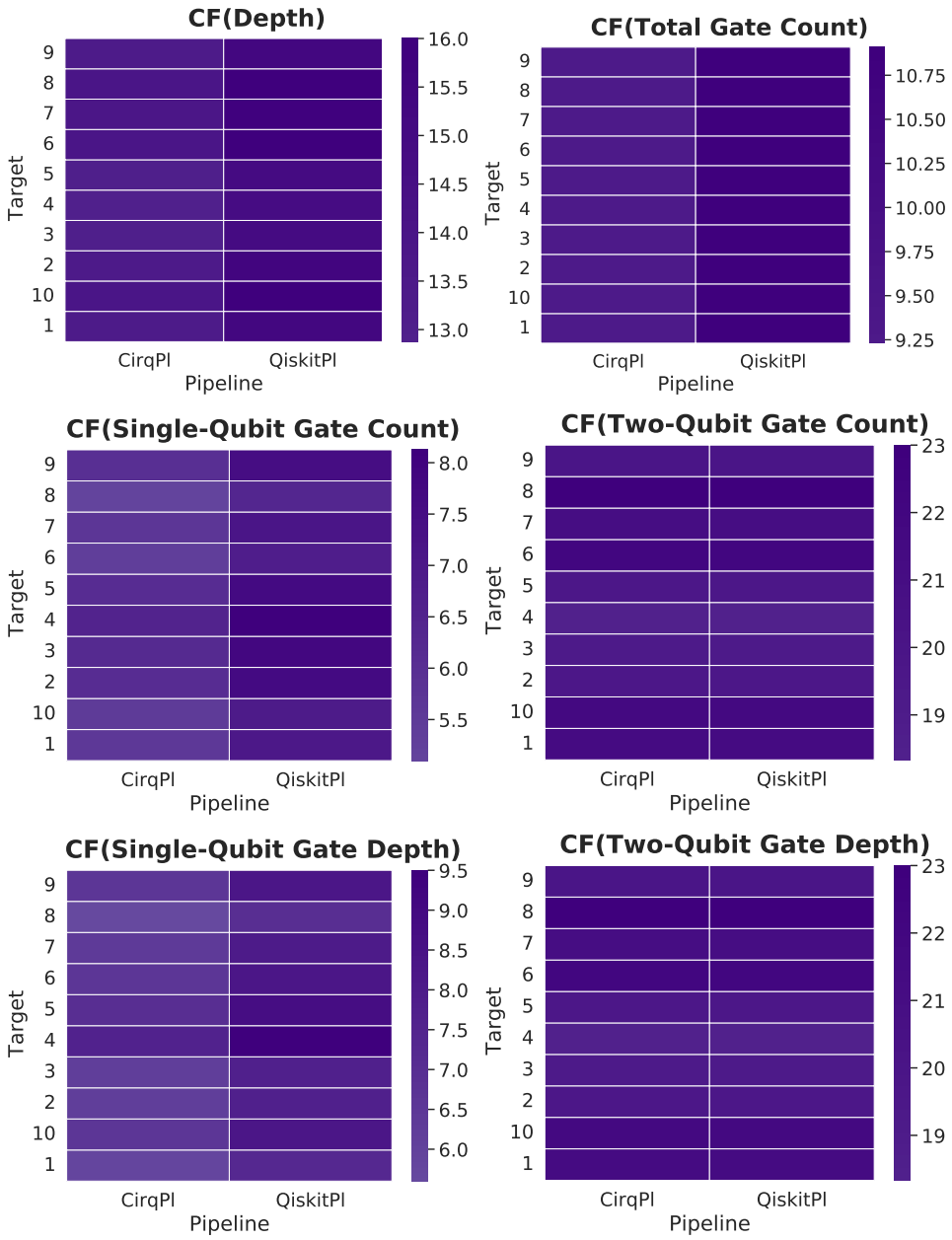


Figure 4.7: Compression factor ($CF$) vs circuit for IbmAll2All2Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | 1 | 7 | CirqPl | 1 | 8 |
| Total Gate Count | QiskitPl | 1 | 11 | CirqPl | 1 | 13 |
| Single-Qubit Gate Count | QiskitPl | 1 | 8 | CirqPl | 1 | 10 |
| Two-Qubit Gate Count | QiskitPl | 1 | 3 | QiskitPl | 1 | 3 |
| Single-Qubit Gate Depth | QiskitPl | 1 | 4 | CirqPl | 1 | 5 |
| Two-Qubit Gate Depth | QiskitPl | 1 | 3 | QiskitPl | 1 | 3 |
| Circuit Cost Function | QiskitPl | 1 | 0.073 | CirqPl | 1 | 0.08 |
| Execution Time | QiskitPl | 3 | 0.07 | CirqPl | 7 | 0.312 |

## 4.2 Hardware Comparison

This analysis is performed across all classes of target circuit.

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Depth | IbmAll2All2Q | IbmAll2All2Q |
| Total Gate Count | IbmAll2All2Q | IbmAll2All2Q |
| Single-Qubit Gate Count | IbmAll2All2Q | IbmAll2All2Q |
| Two-Qubit Gate Count | IbmAll2All2Q | IbmAll2All2Q |
| Single-Qubit Gate Depth | IbmAll2All2Q | IbmAll2All2Q |
| Two-Qubit Gate Depth | IbmAll2All2Q | IbmAll2All2Q |
| Circuit Cost Function | IbmAll2All2Q | IbmAll2All2Q |
| Execution Time | IbmAll2All2Q | IbmAll2All2Q |

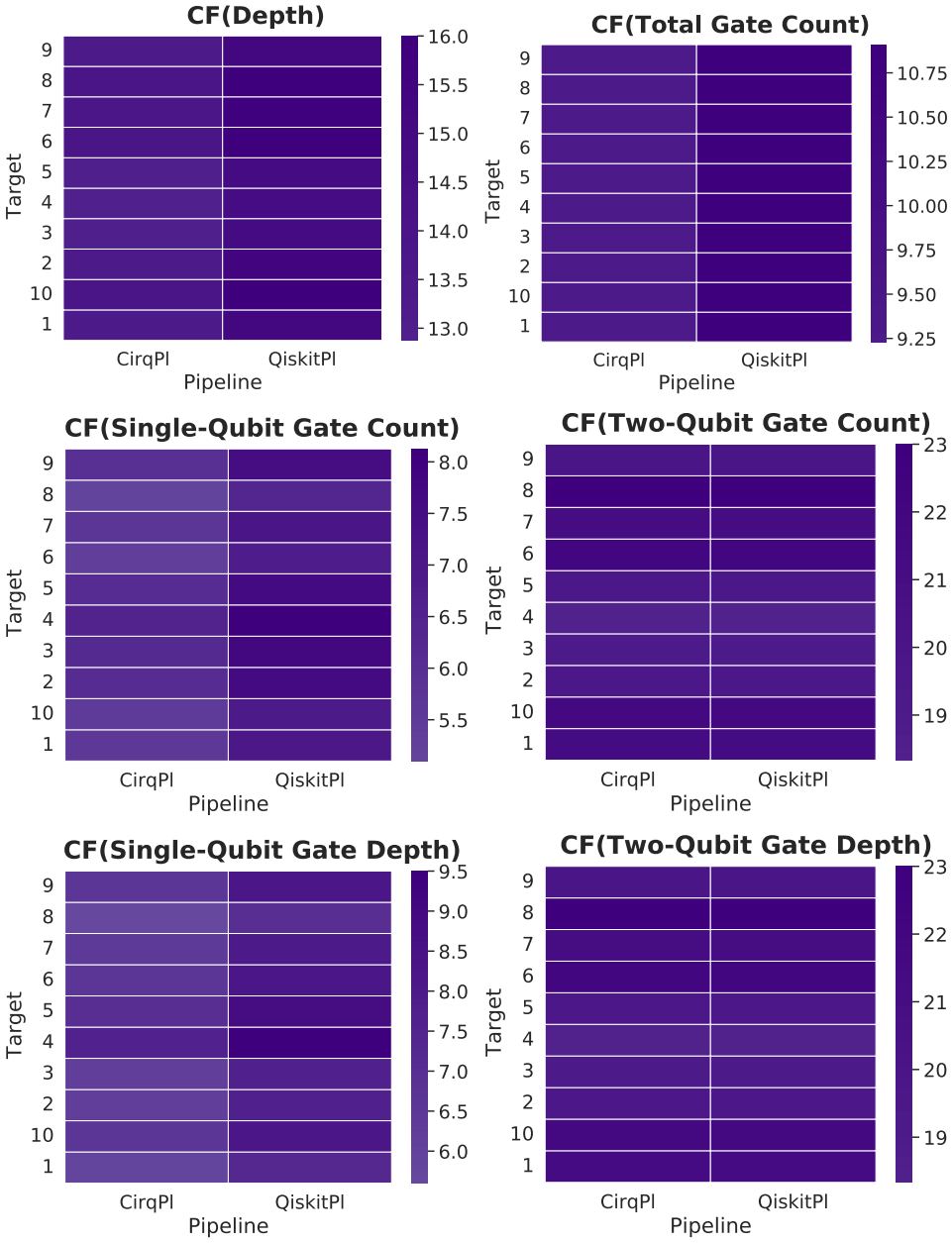


Figure 4.8: Final circuit metrics after compilation vs hardware backend.

# Chapter 5

# Quick Summary

## 5.1 Aggregate Multi-Factor Comparison

This analysis is performed across all compilation frameworks, target circuits and hardware. From this result, we can deduce the degree of compressibility of different target circuits.

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance. This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

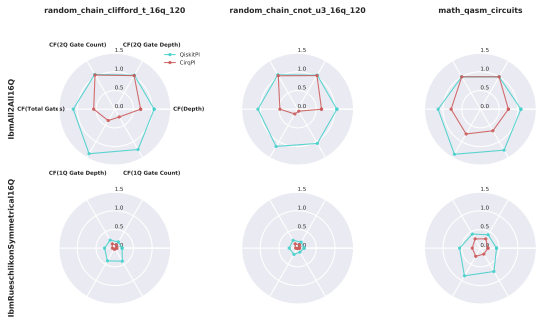


Figure 5.1: Aggregate multi-factor comparison of compilation frameworks: compression factor ($CF$) across various features. Columns correspond to specific target circuit classes, and row correspond to different hardware architectures. Better performance corresponds to a larger polygon area.

# Chapter 6

# Target: Random Circuits from [Clifford + T] Gate Set

## 6.1 Hardware: IbmRueschlikonSymmetrical16Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

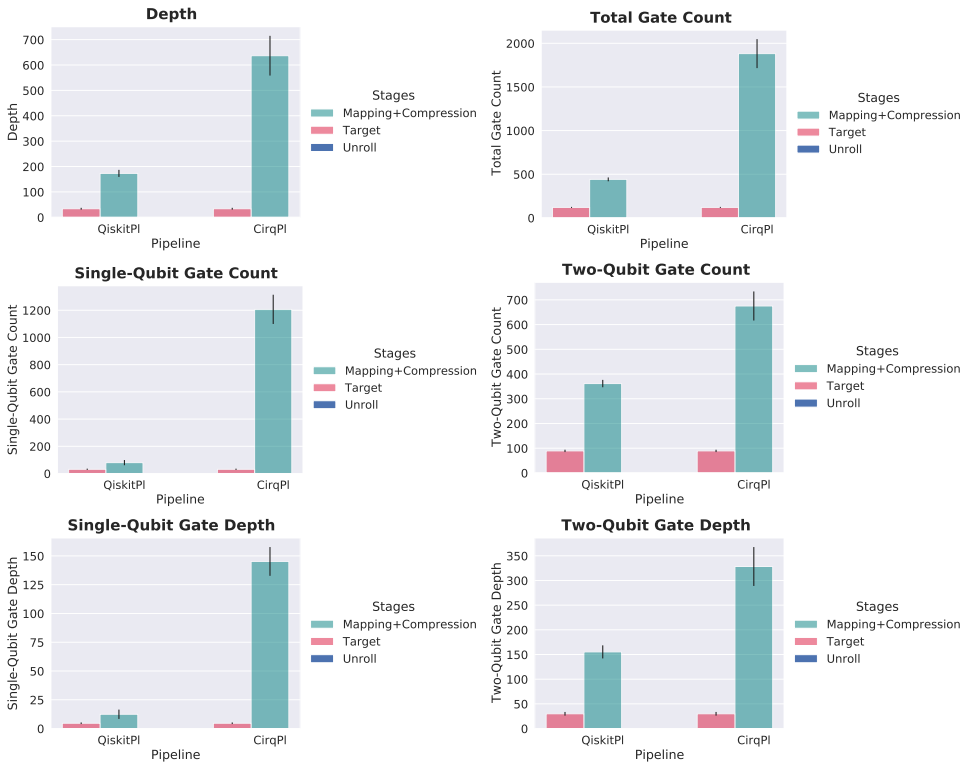


Figure 6.1: Circuits metrics for each compilation pipeline stage for IbmRueschlikonSymmetrical16Q.

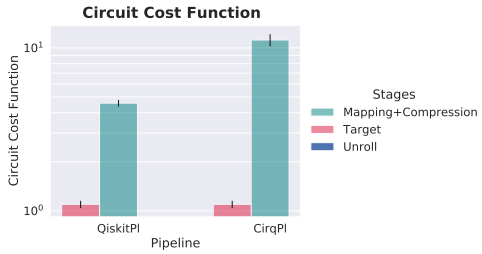


Figure 6.2: Circuit cost function for IbmRueschlikonSymmetrical16Q.

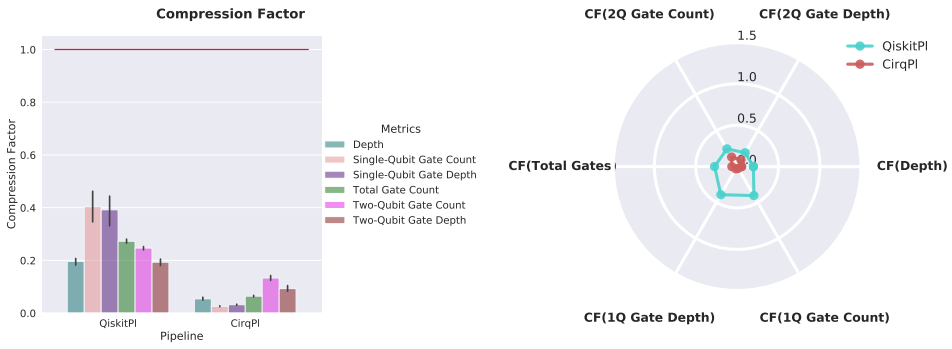


Figure 6.3: Compression factor ($CF$) between target and final compilation stage for IbmRueschlikon-Symmetrical16Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

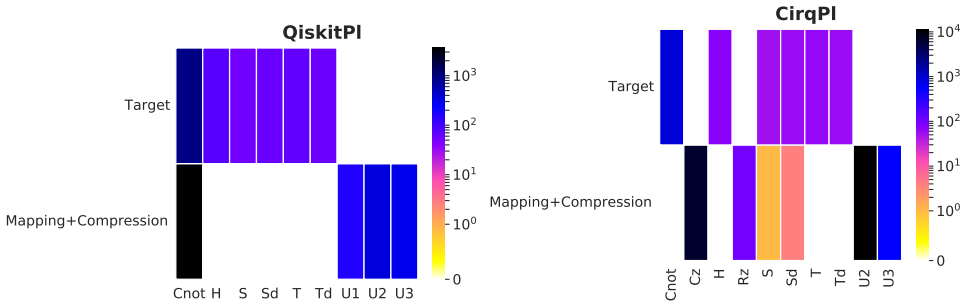


Figure 6.4: Gate frequencies in each pipeline stage for IbmRueschlikonSymmetrical16Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

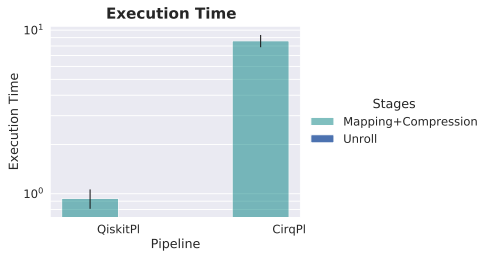


Figure 6.5: Mean execution time of each compilation stage for IbmRueschlikonSymmetrical16Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | QiskitPl | CirqPl |

Continued on Next Page

| Metrics | Min | Max |
|---|---|---|
| Single-Qubit Gate Depth | QiskitPl | CirqPl |
| Two-Qubit Gate Depth | QiskitPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

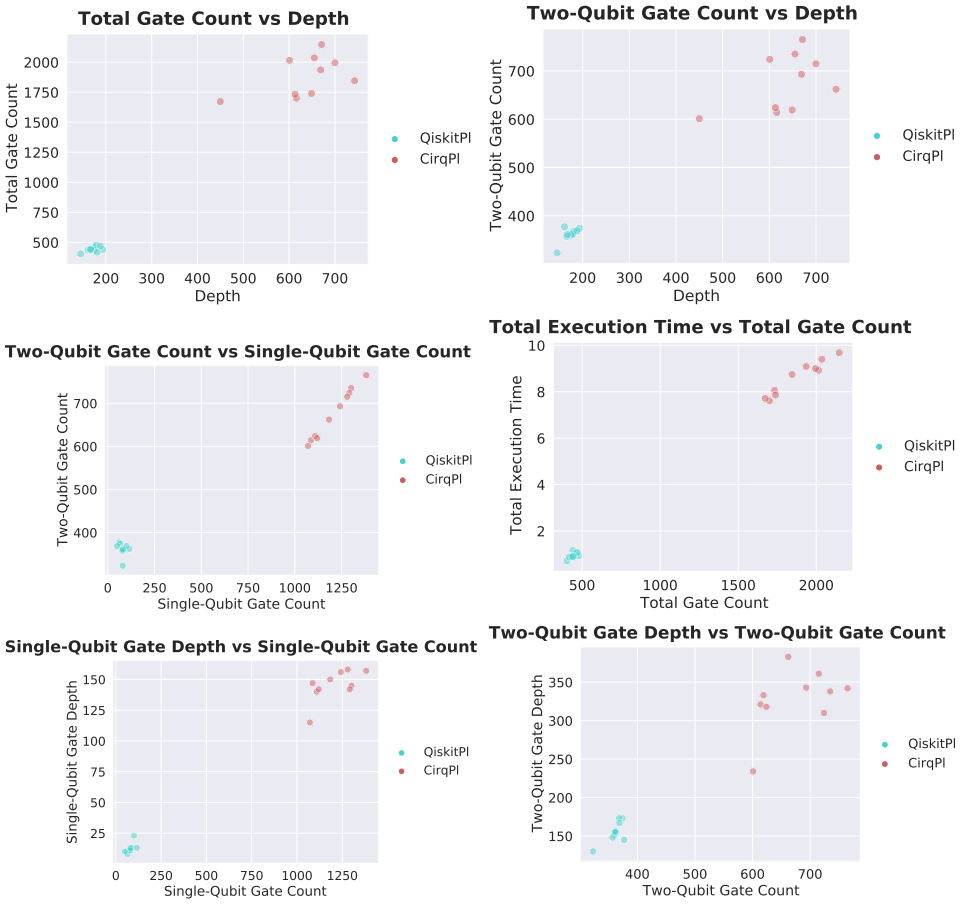


Figure 6.6: Cluster analytics for IbmRueschlikonSymmetrical16Q. Each point corresponds to an individual target quantum circuit from the target generator.

## Breakdown by individual circuits

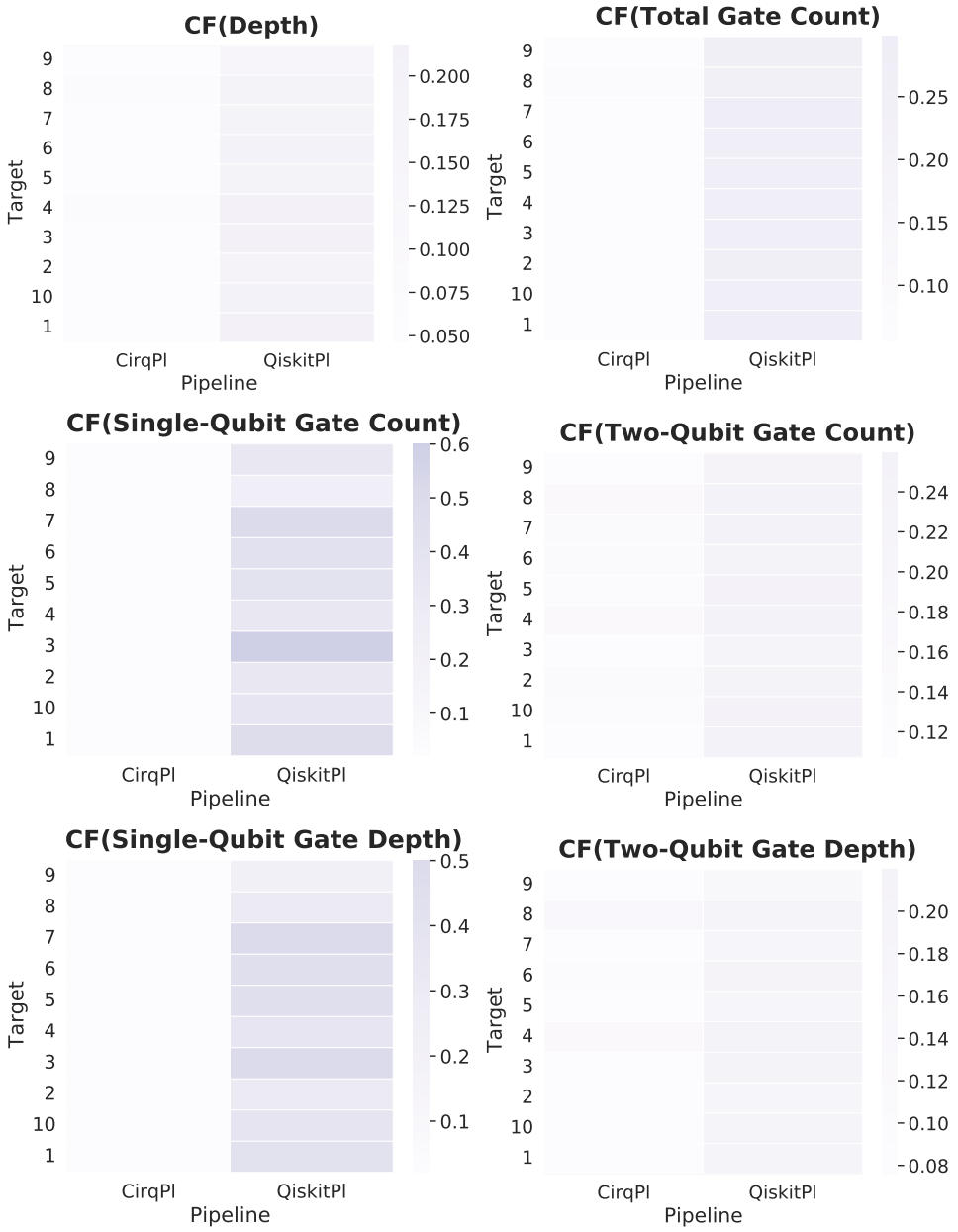


Figure 6.7: Compression factor ($CF$) vs circuit for IbmRueschlikonSymmetrical16Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | 1 | 145 | CirqPl | 7 | 743 |
| Total Gate Count | QiskitPl | 1 | 403 | CirqPl | 1 | 2147 |
| Single-Qubit Gate Count | QiskitPl | 7 | 50 | CirqPl | 1 | 1382 |
| Two-Qubit Gate Count | QiskitPl | 1 | 323 | CirqPl | 1 | 765 |
| Single-Qubit Gate Depth | QiskitPl | 4 | 8 | CirqPl | 10 | 158 |
| Two-Qubit Gate Depth | QiskitPl | 1 | 130 | CirqPl | 7 | 383 |
| Circuit Cost Function | QiskitPl | 1 | 4.053 | CirqPl | 1 | 12.435 |
| Execution Time | QiskitPl | 1 | 0.71 | CirqPl | 1 | 9.678 |

## 6.2 Hardware: IbmAll2All16Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

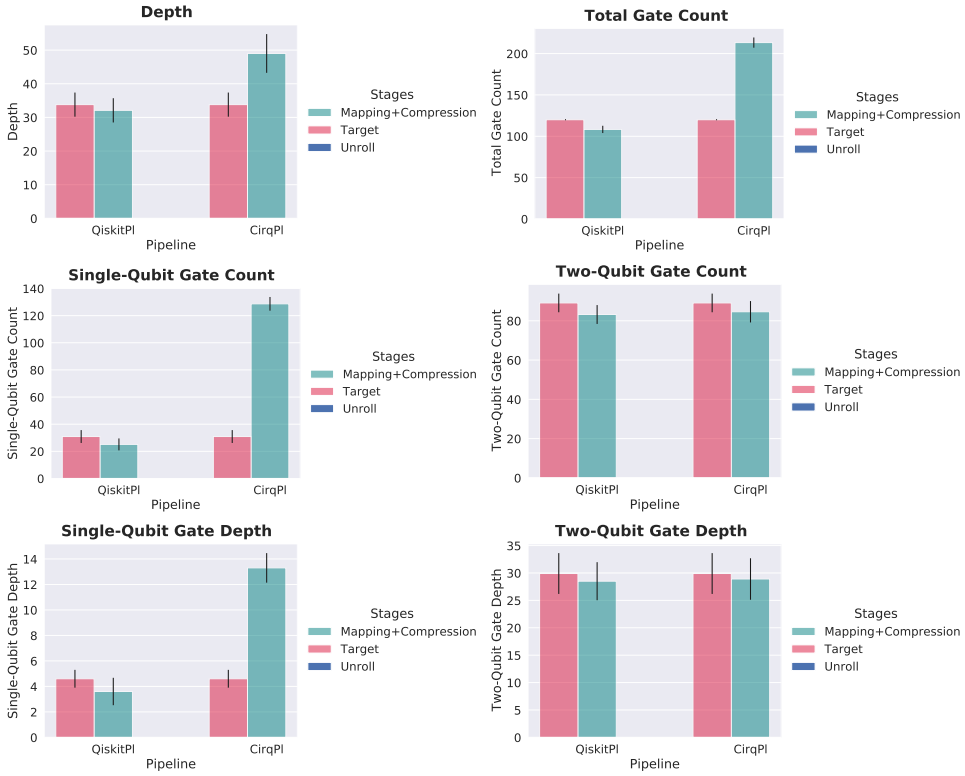


Figure 6.8: Circuits metrics for each compilation pipeline stage for IbmAll2All16Q.

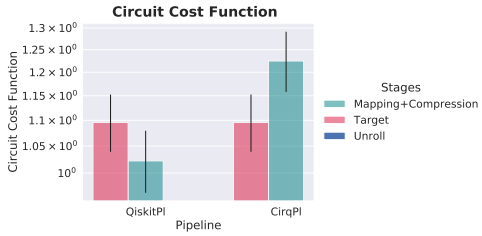


Figure 6.9: Circuit cost function for IbmAll2All16Q.

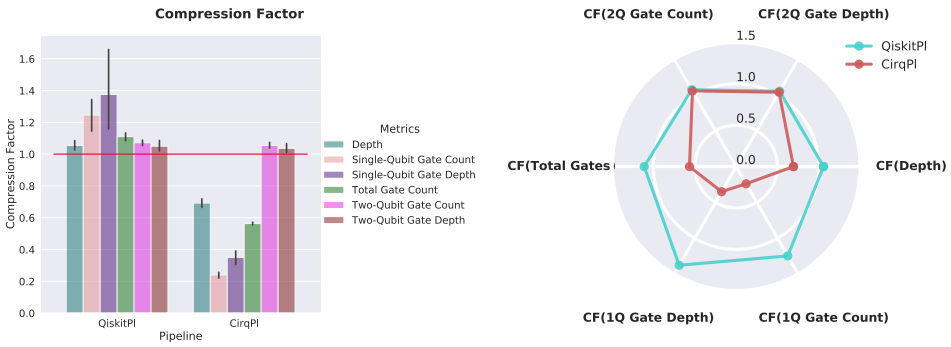


Figure 6.10: Compression factor ($CF$) between target and final compilation stage for IbmAll2All16Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

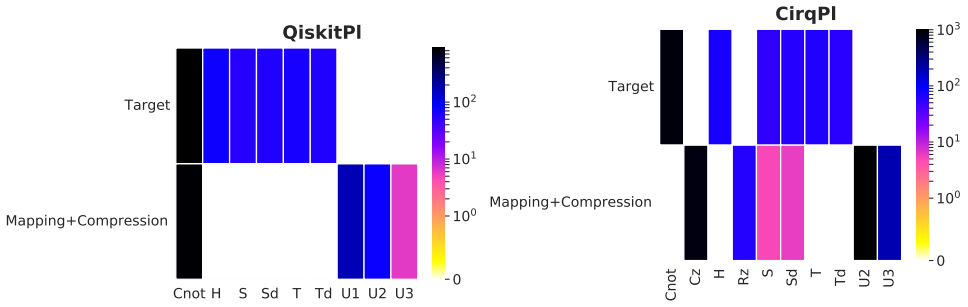


Figure 6.11: Gate frequencies in each pipeline stage for IbmAll2All16Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

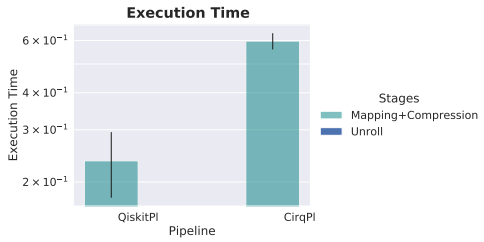


Figure 6.12: Mean execution time of each compilation stage for IbmAll2All16Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| Metrics | Min | Max |
|---|---|---|
| Two-Qubit Gate Depth | QiskitPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

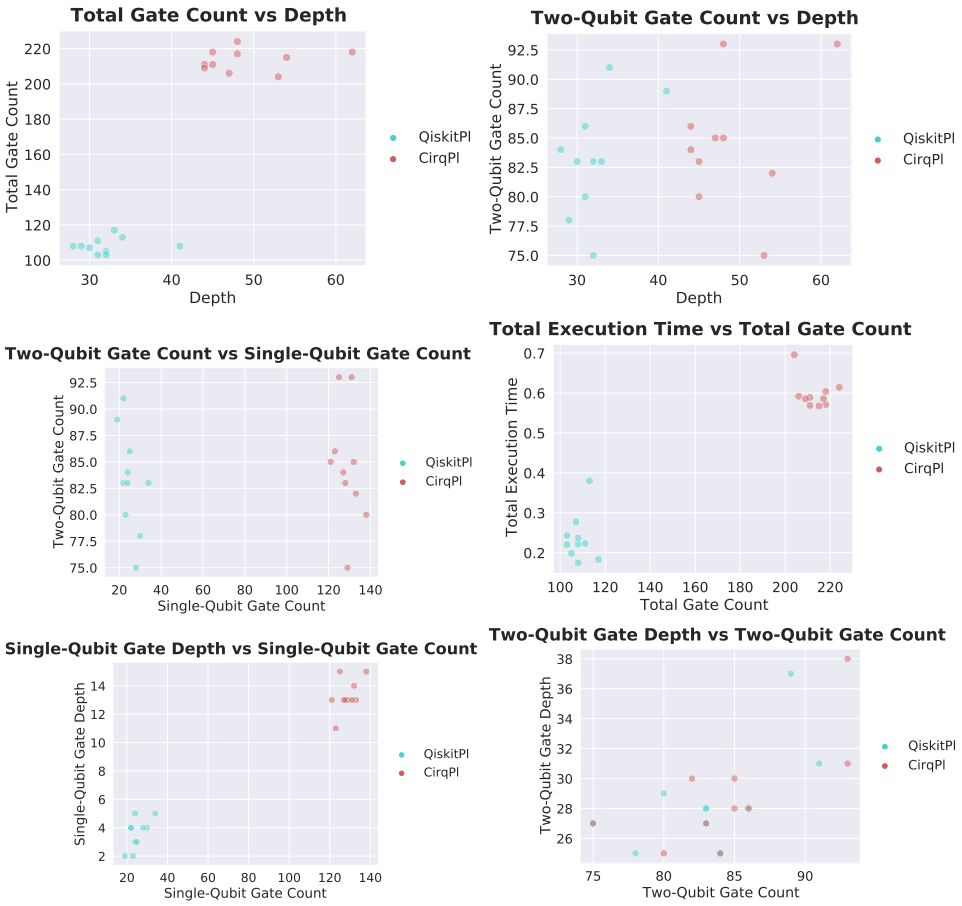


Figure 6.13: Cluster analytics for IbmAll2All16Q. Each point corresponds to an individual target quantum circuit from the target generator.

### Breakdown by individual circuits

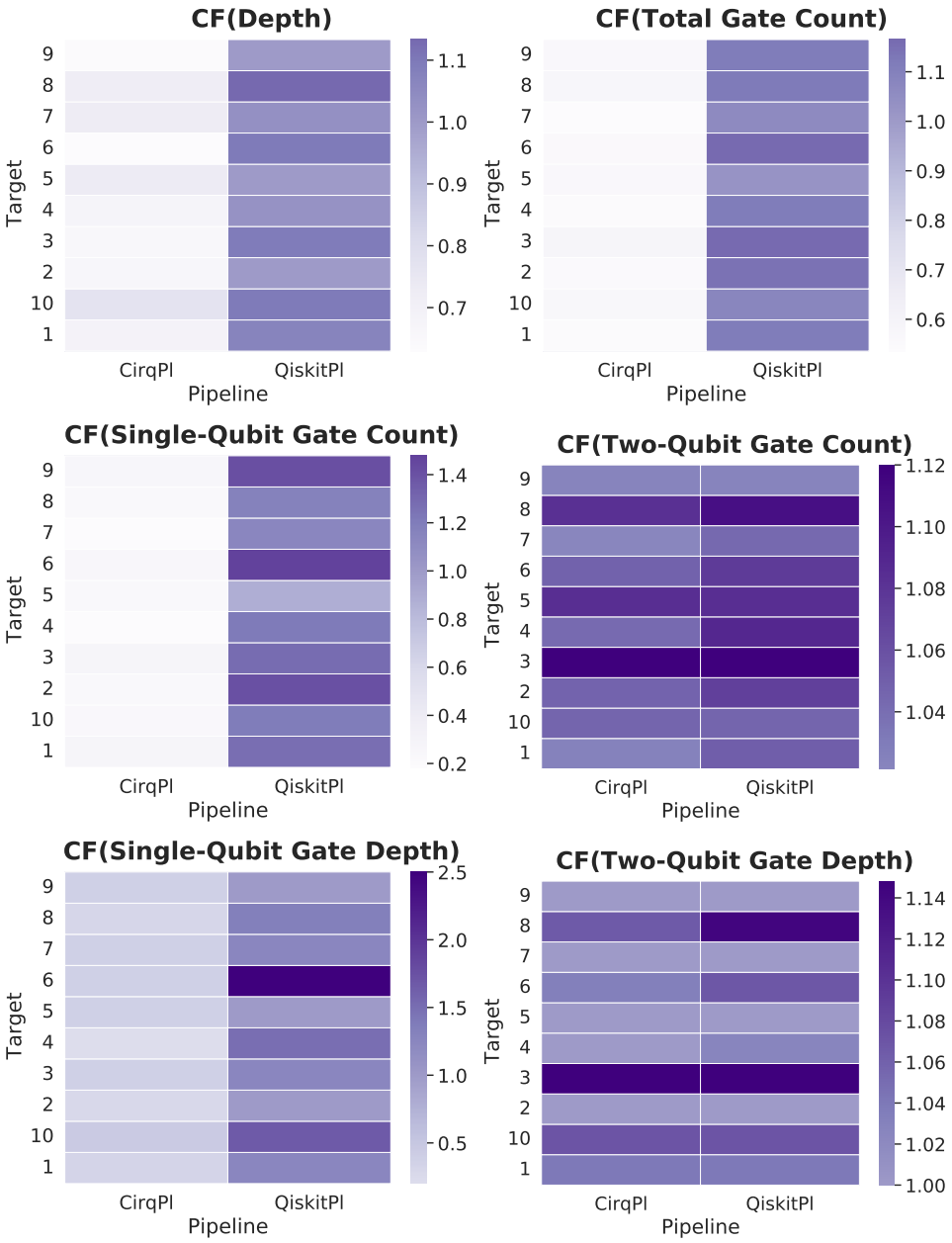


Figure 6.14: Compression factor ($CF$) vs circuit for IbmAll2All16Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | 9 | 28 | CirqPl | 4 | 62 |
| Total Gate Count | QiskitPl | 3 | 103 | CirqPl | 7 | 224 |
| Single-Qubit Gate Count | QiskitPl | 4 | 19 | CirqPl | 1 | 138 |
| Two-Qubit Gate Count | QiskitPl | 3 | 75 | CirqPl | 4 | 93 |
| Single-Qubit Gate Depth | QiskitPl | 4 | 2 | CirqPl | 1 | 15 |
| Two-Qubit Gate Depth | QiskitPl | 1 | 25 | CirqPl | 4 | 38 |
| Circuit Cost Function | QiskitPl | 3 | 0.942 | CirqPl | 4 | 1.371 |
| Execution Time | QiskitPl | 9 | 0.174 | CirqPl | 3 | 0.696 |

## 6.3 Hardware Comparison

This analysis is performed across all classes of target circuit.

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Depth | IbmAll2All16Q | IbmAll2All16Q |
| Total Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Single-Qubit Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Two-Qubit Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Single-Qubit Gate Depth | IbmAll2All16Q | IbmAll2All16Q |
| Two-Qubit Gate Depth | IbmAll2All16Q | IbmAll2All16Q |
| Circuit Cost Function | IbmAll2All16Q | IbmAll2All16Q |
| Execution Time | IbmAll2All16Q | IbmAll2All16Q |

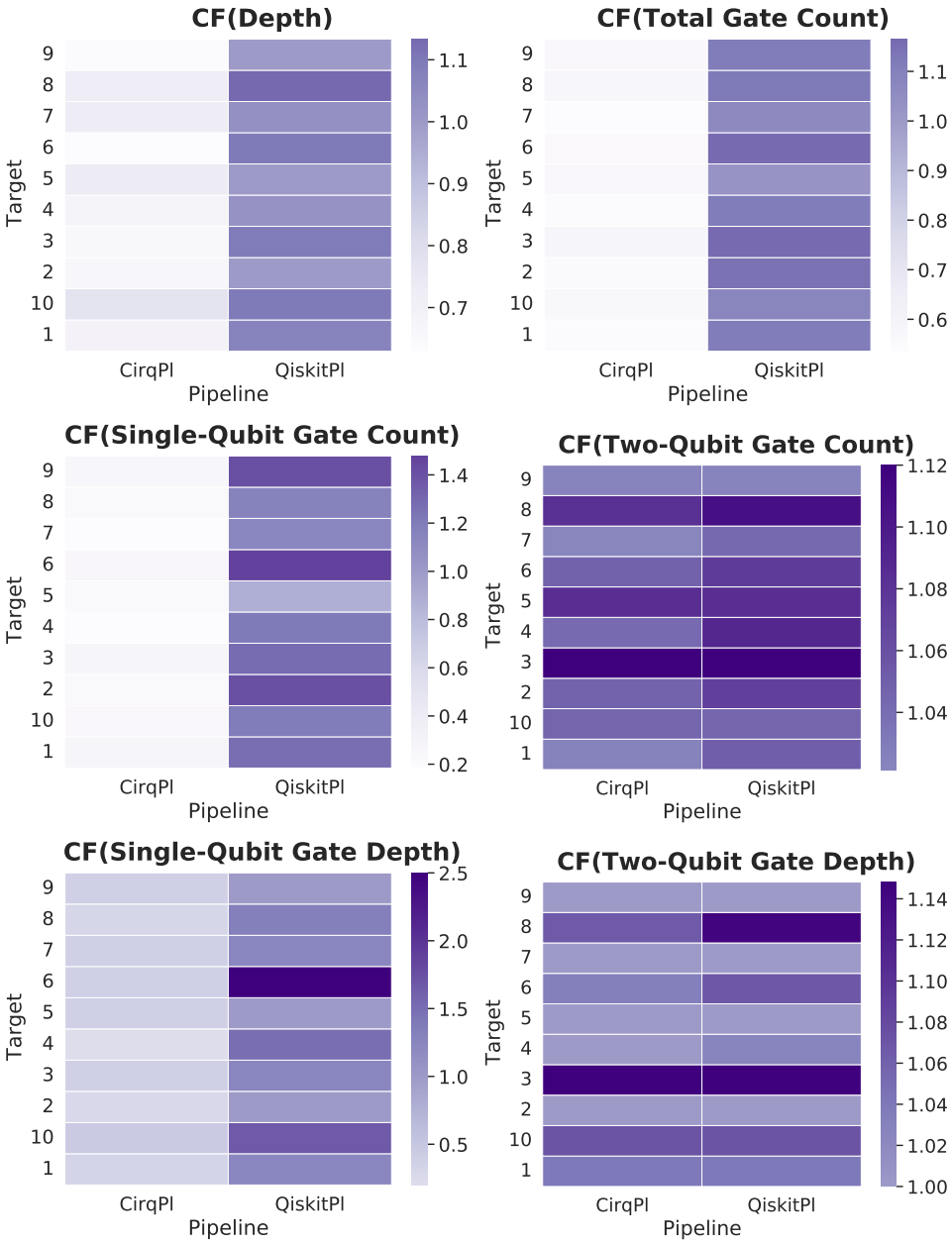


Figure 6.15: Final circuit metrics after compilation vs hardware backend.

# Chapter 7

# Target: Random Circuits from [CNOT, $U_3$] Gate Set

## 7.1 Hardware: IbmRueschlikonSymmetrical16Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

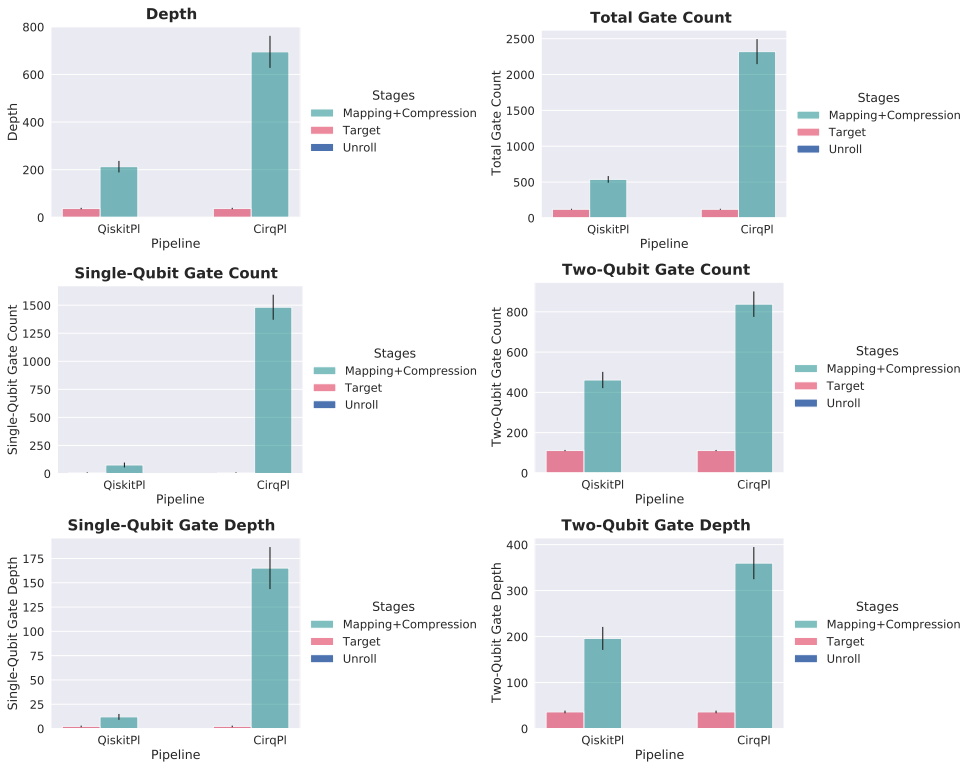


Figure 7.1: Circuits metrics for each compilation pipeline stage for IbmRueschlikonSymmetrical16Q.

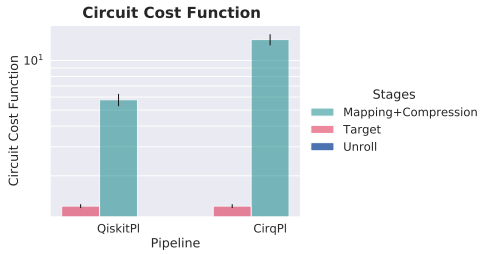


Figure 7.2: Circuit cost function for IbmRueschlikonSymmetrical16Q.

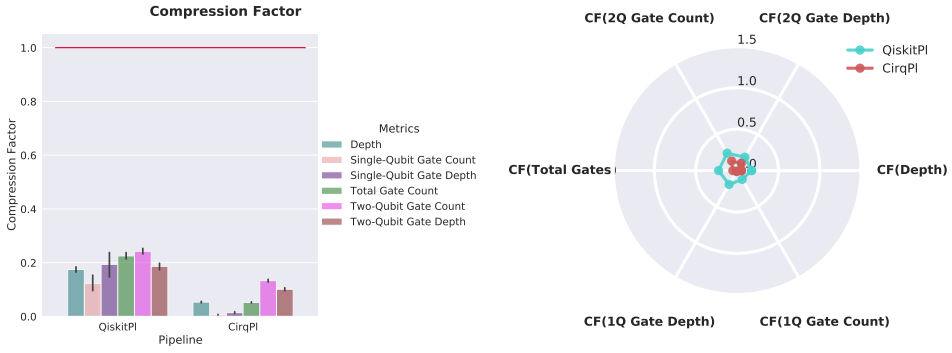


Figure 7.3: Compression factor ($CF$) between target and final compilation stage for IbmRueschlikon-Symmetrical16Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

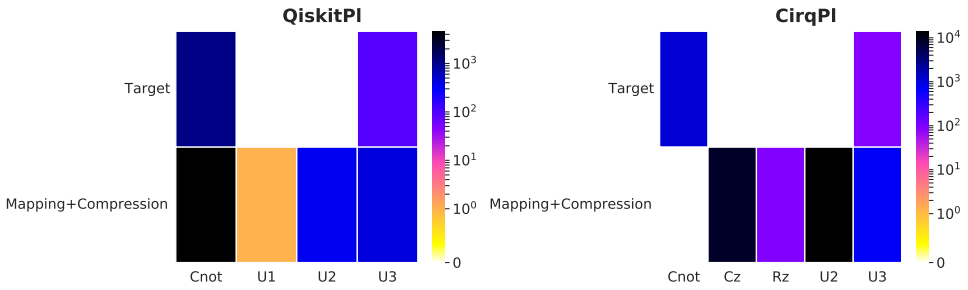


Figure 7.4: Gate frequencies in each pipeline stage for IbmRueschlikonSymmetrical16Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

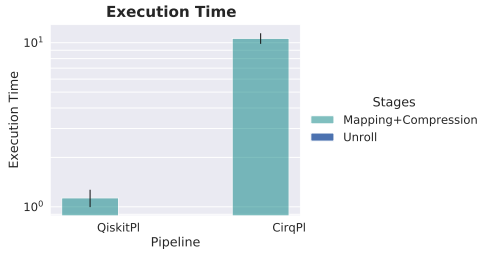


Figure 7.5: Mean execution time of each compilation stage for IbmRueschlikonSymmetrical16Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| Metrics | Min | Max |
|---|---|---|
| Two-Qubit Gate Depth | QiskitPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

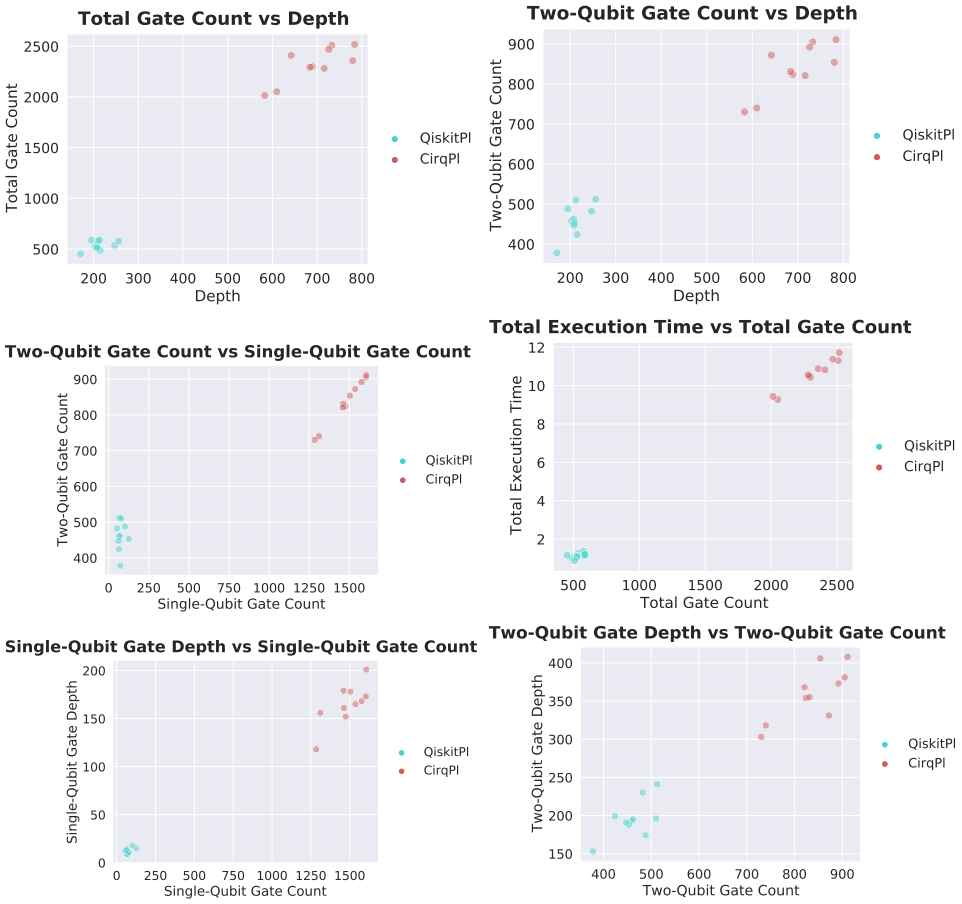


Figure 7.6: Cluster analytics for IbmRueschlikonSymmetrical16Q. Each point corresponds to an individual target quantum circuit from the target generator.

## Breakdown by individual circuits

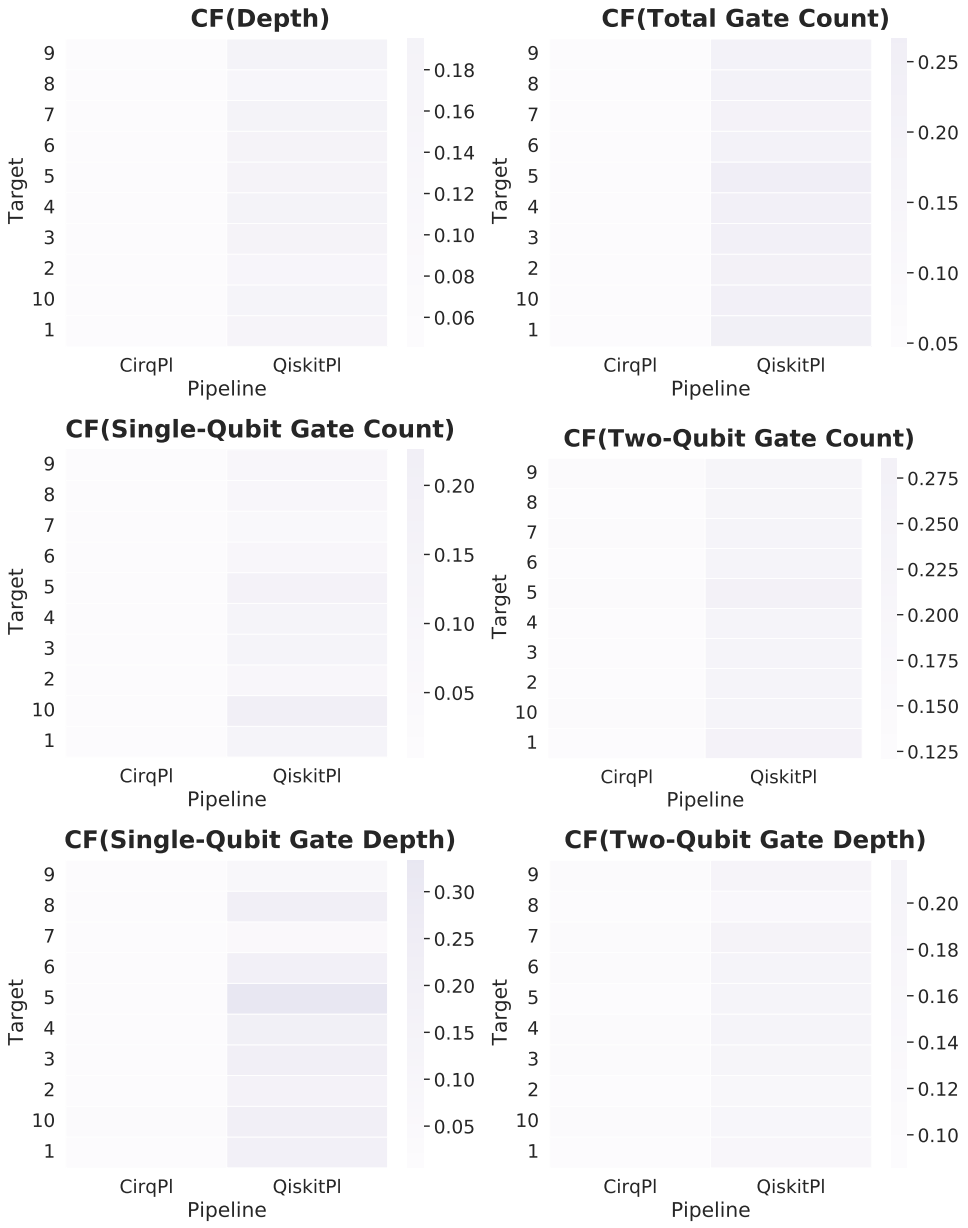


Figure 7.7: Compression factor ($CF$) vs circuit for IbmRueschlikonSymmetrical16Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | 5 | 171 | CirqPl | 8 | 784 |
| Total Gate Count | QiskitPl | 5 | 451 | CirqPl | 8 | 2517 |
| Single-Qubit Gate Count | QiskitPl | 2 | 53 | CirqPl | 8 | 1606 |
| Two-Qubit Gate Count | QiskitPl | 5 | 378 | CirqPl | 8 | 911 |
| Single-Qubit Gate Depth | QiskitPl | 3 | 9 | CirqPl | 8 | 201 |
| Two-Qubit Gate Depth | QiskitPl | 5 | 153 | CirqPl | 8 | 408 |
| Circuit Cost Function | QiskitPl | 5 | 4.729 | CirqPl | 8 | 14.692 |
| Execution Time | QiskitPl | 10 | 0.861 | CirqPl | 8 | 11.713 |

## 7.2 Hardware: IbmAll2All16Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

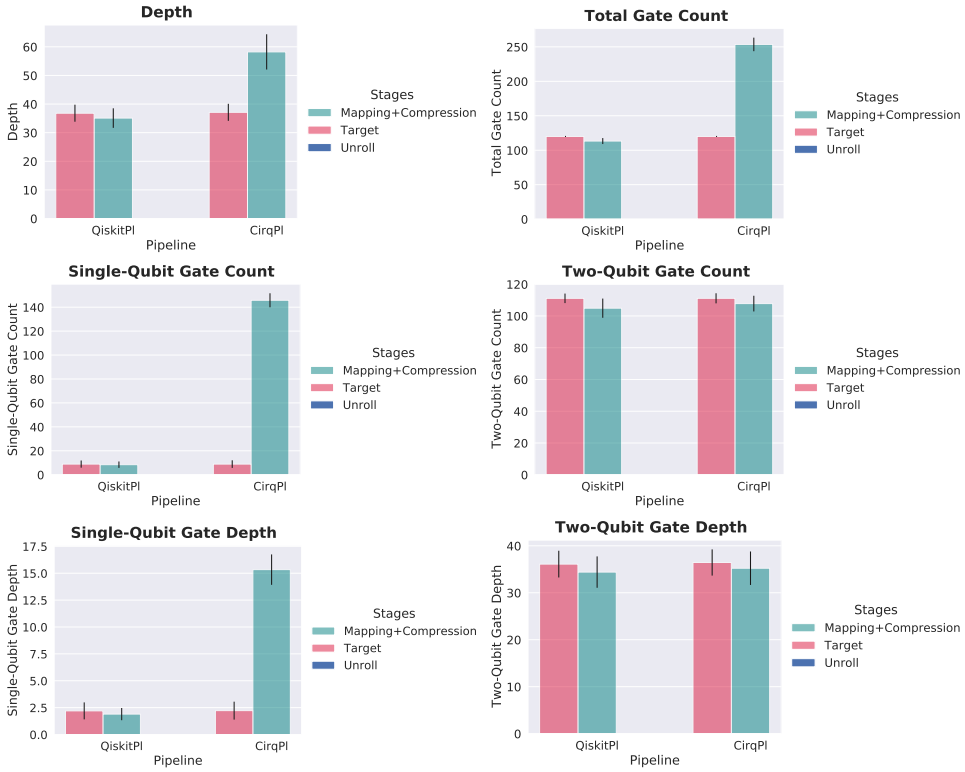


Figure 7.8: Circuits metrics for each compilation pipeline stage for IbmAll2All16Q.

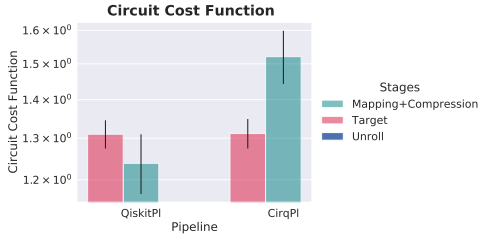


Figure 7.9: Circuit cost function for IbmAll2All16Q.

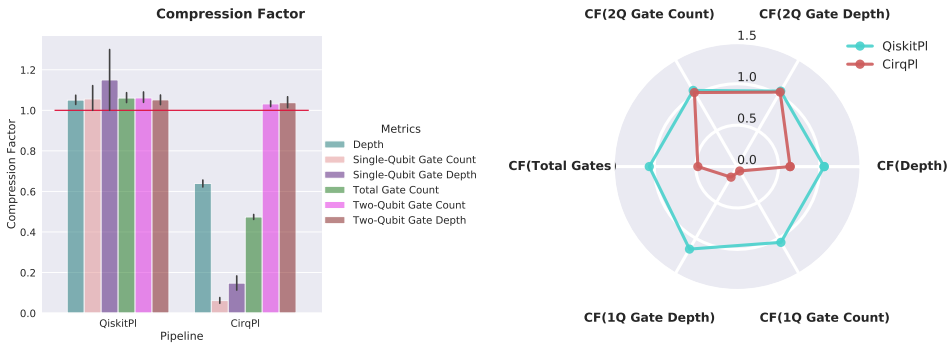


Figure 7.10: Compression factor ($CF$) between target and final compilation stage for IbmAll2All16Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

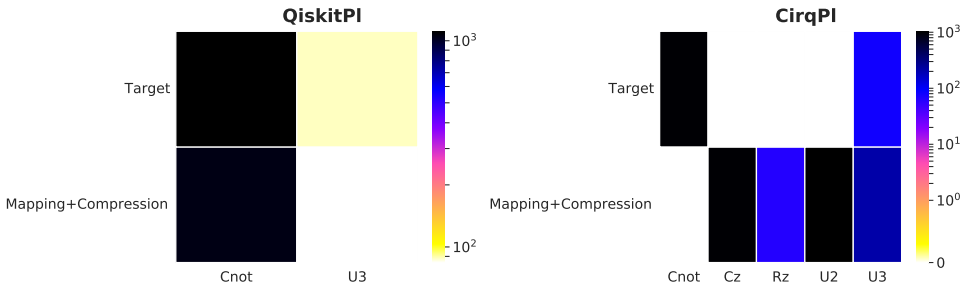


Figure 7.11: Gate frequencies in each pipeline stage for IbmAll2All16Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

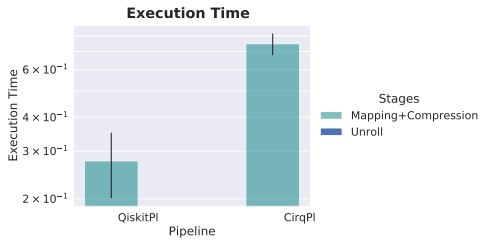


Figure 7.12: Mean execution time of each compilation stage for IbmAll2All16Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Two-Qubit Gate Depth | QiskitPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

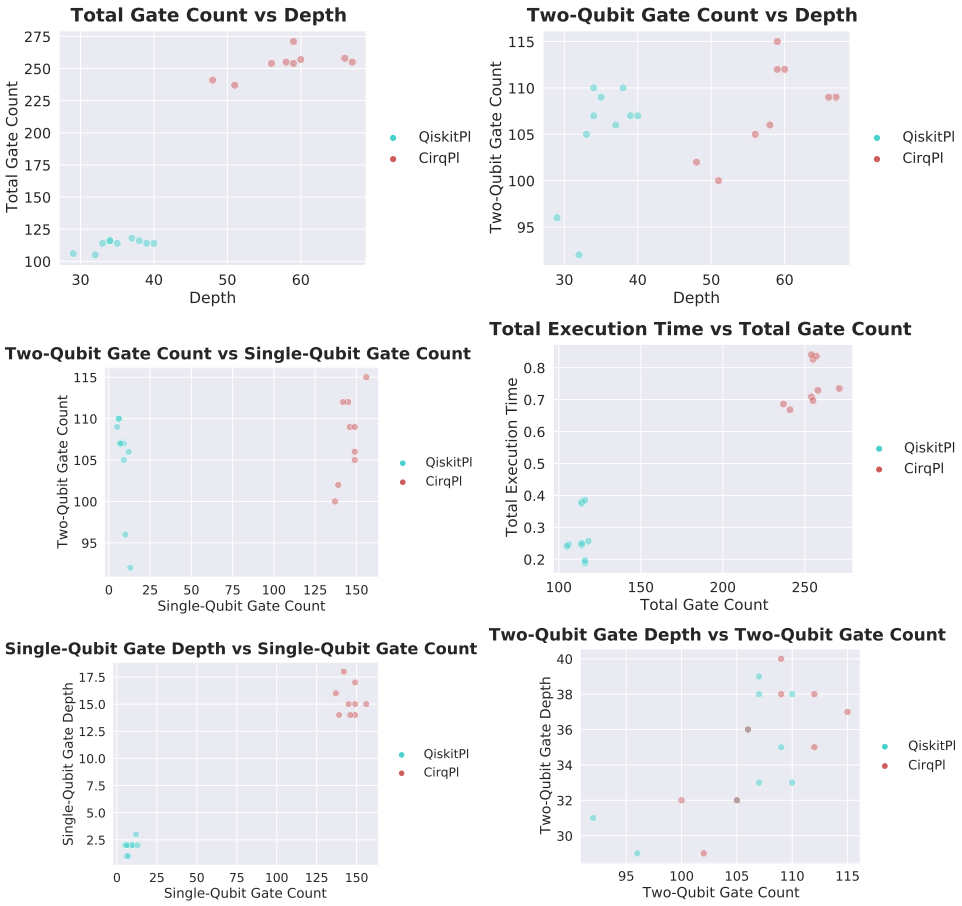


Figure 7.13: Cluster analytics for IbmAll2All16Q. Each point corresponds to an individual target quantum circuit from the target generator.

## Breakdown by individual circuits

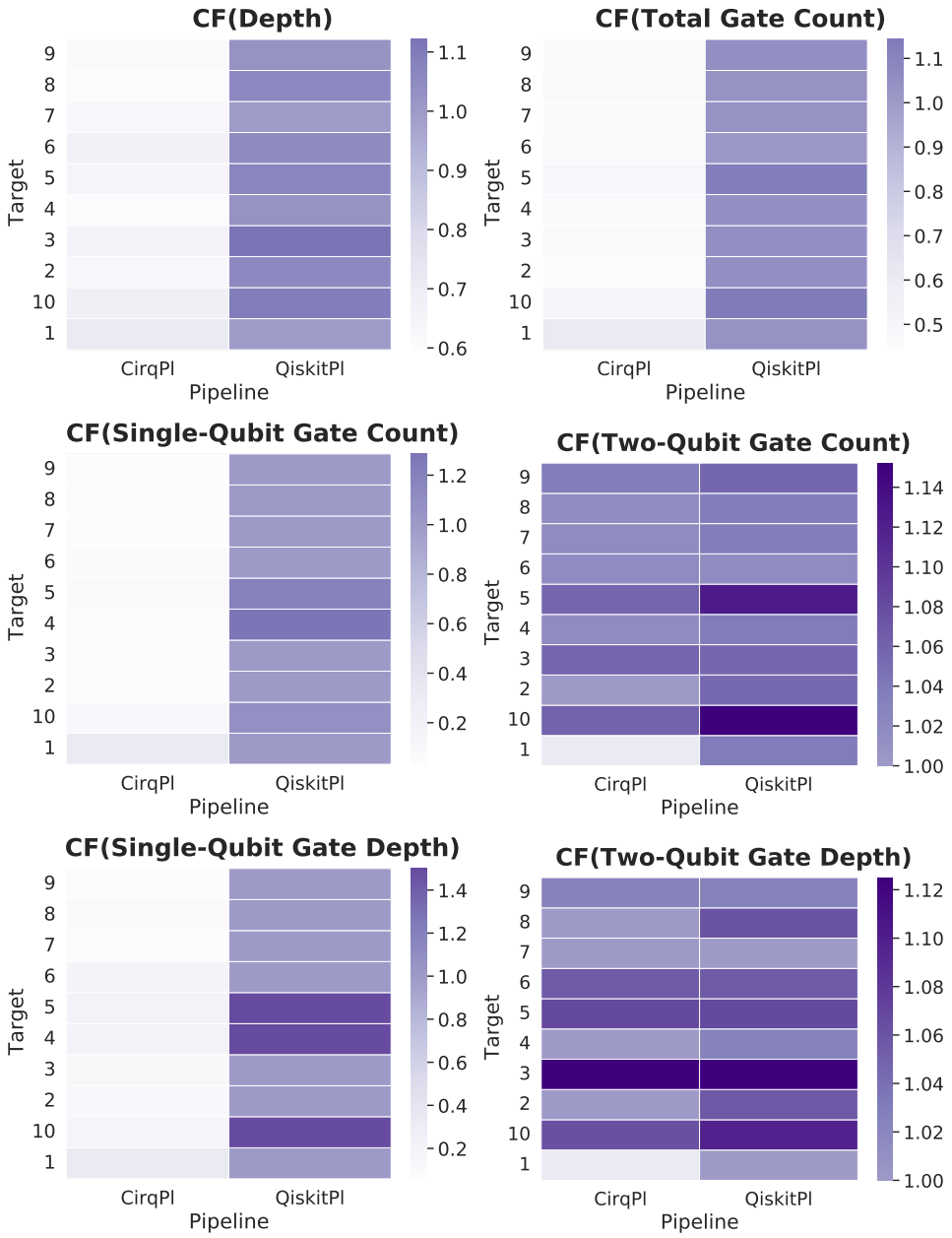


Figure 7.14: Compression factor ($CF$) vs circuit for IbmAll2All16Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | 5 | 29 | CirqPl | 4 | 67 |
| Total Gate Count | QiskitPl | 10 | 105 | CirqPl | 2 | 271 |
| Single-Qubit Gate Count | QiskitPl | 2 | 5 | CirqPl | 2 | 156 |
| Two-Qubit Gate Count | QiskitPl | 10 | 92 | CirqPl | 2 | 115 |
| Single-Qubit Gate Depth | QiskitPl | 7 | 1 | CirqPl | 8 | 18 |
| Two-Qubit Gate Depth | QiskitPl | 5 | 29 | CirqPl | 4 | 40 |
| Circuit Cost Function | QiskitPl | 10 | 1.098 | CirqPl | 2 | 1.608 |
| Execution Time | QiskitPl | 1 | 0.189 | CirqPl | 8 | 0.84 |

## 7.3 Hardware Comparison

This analysis is performed across all classes of target circuit.

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Depth | IbmAll2All16Q | IbmAll2All16Q |
| Total Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Single-Qubit Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Two-Qubit Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Single-Qubit Gate Depth | IbmAll2All16Q | IbmAll2All16Q |
| Two-Qubit Gate Depth | IbmAll2All16Q | IbmAll2All16Q |
| Circuit Cost Function | IbmAll2All16Q | IbmAll2All16Q |
| Execution Time | IbmAll2All16Q | IbmAll2All16Q |

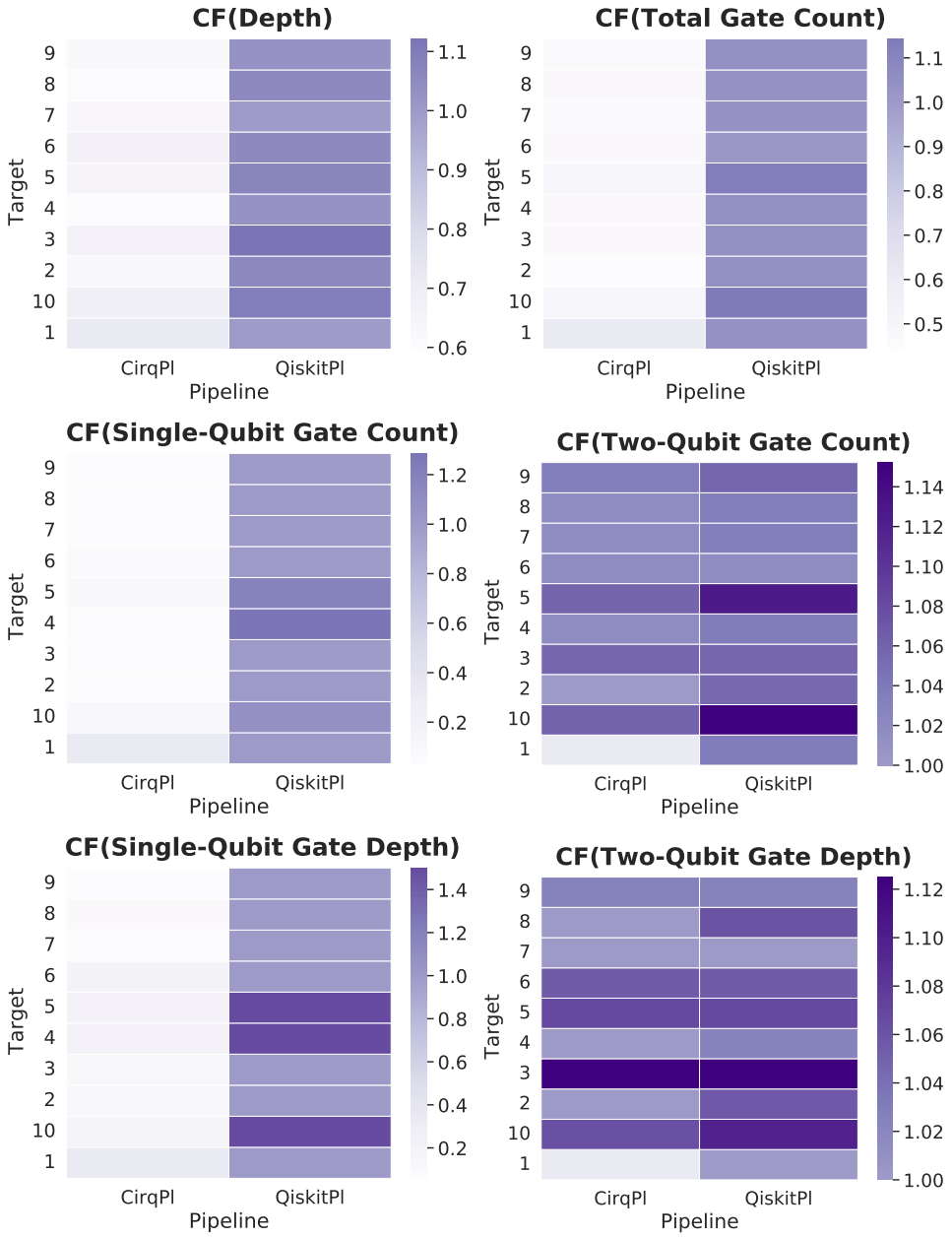


Figure 7.15: Final circuit metrics after compilation vs hardware backend.

# Chapter 8

# Target: QASM Circuits for Arithmetic Algorithms

## 8.1 Hardware: IbmRueschlikonSymmetrical16Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

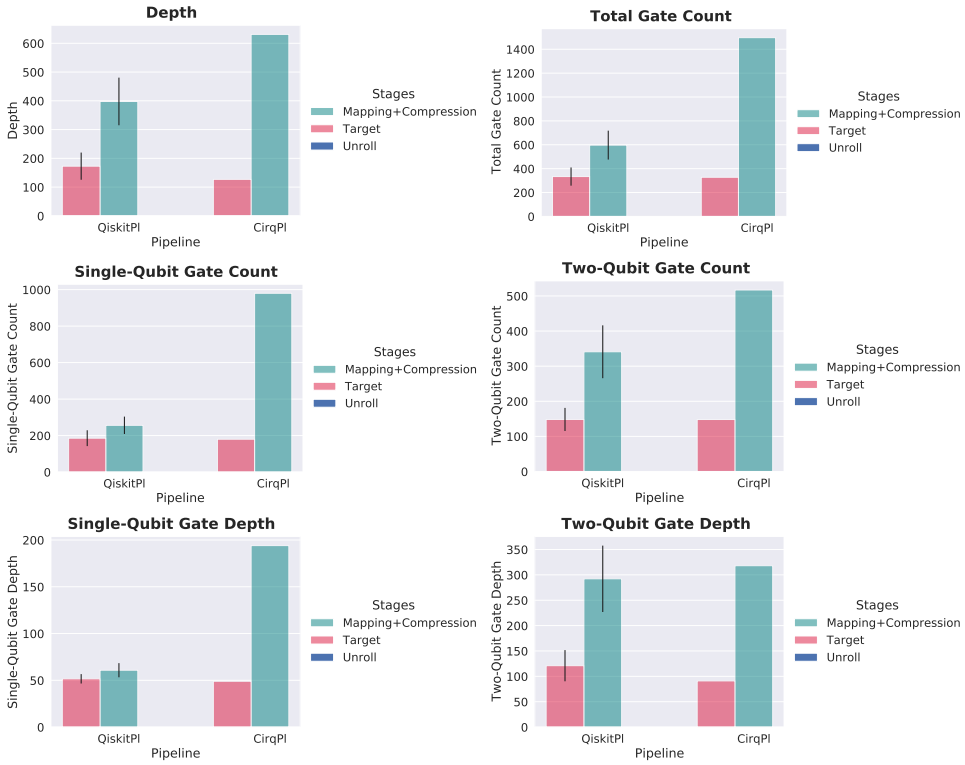


Figure 8.1: Circuits metrics for each compilation pipeline stage for IbmRueschlikonSymmetrical16Q.

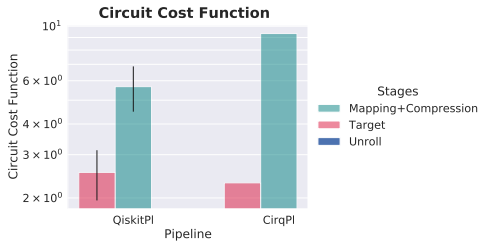


Figure 8.2: Circuit cost function for IbmRueschlikonSymmetrical16Q.

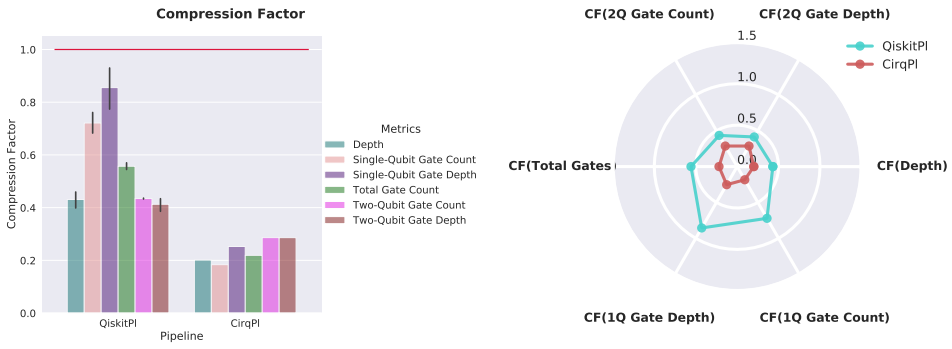


Figure 8.3: Compression factor ($CF$) between target and final compilation stage for IbmRueschlikon-Symmetrical16Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

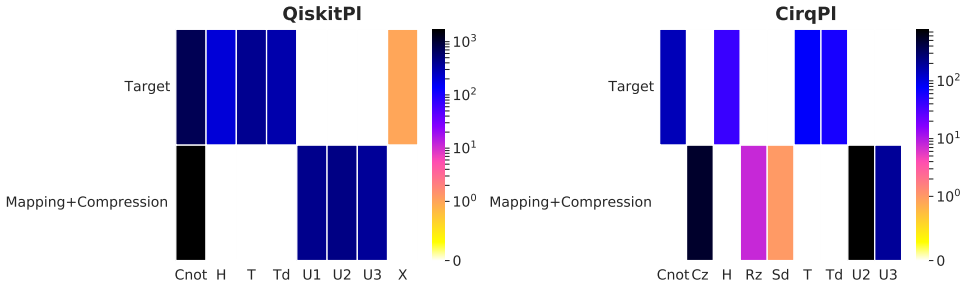


Figure 8.4: Gate frequencies in each pipeline stage for IbmRueschlikonSymmetrical16Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

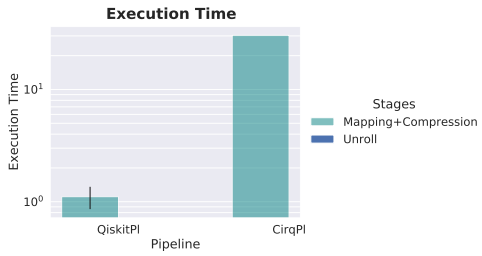


Figure 8.5: Mean execution time of each compilation stage for IbmRueschlikonSymmetrical16Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| Metrics | Min | Max |
|---|---|---|
| Two-Qubit Gate Depth | QiskitPl | CirqPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

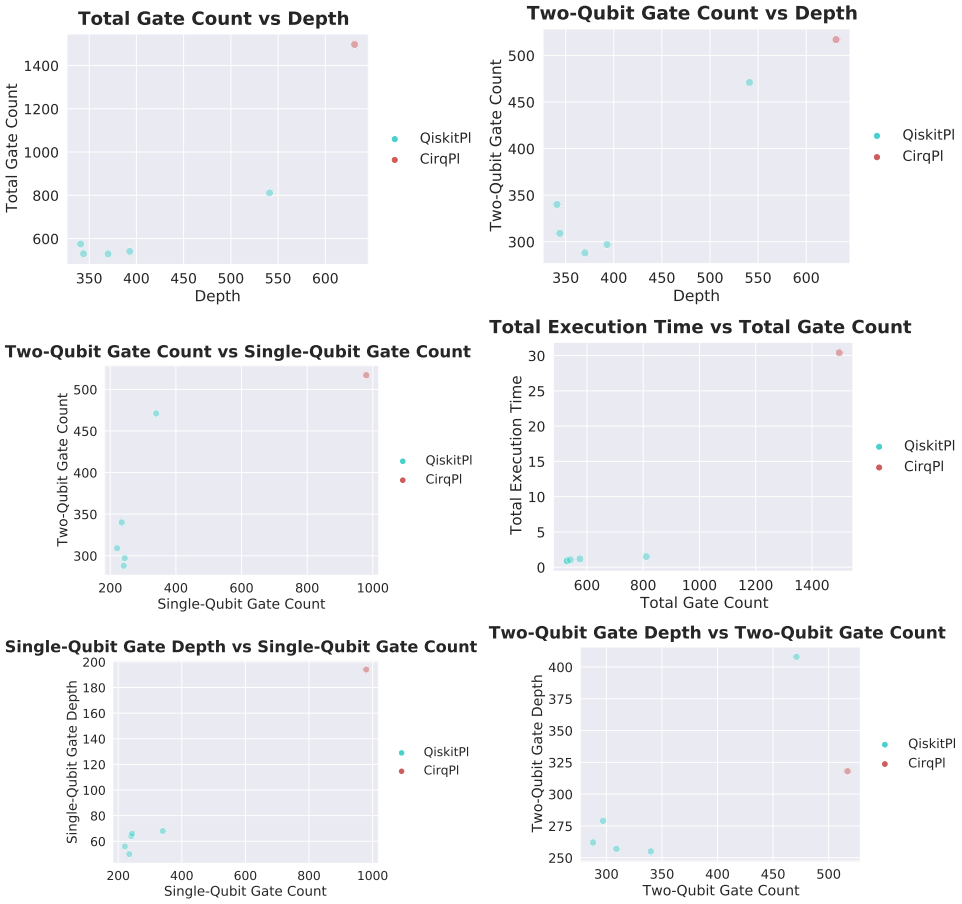


Figure 8.6: Cluster analytics for IbmRueschlikonSymmetrical16Q. Each point corresponds to an individual target quantum circuit from the target generator.

## Breakdown by individual circuits

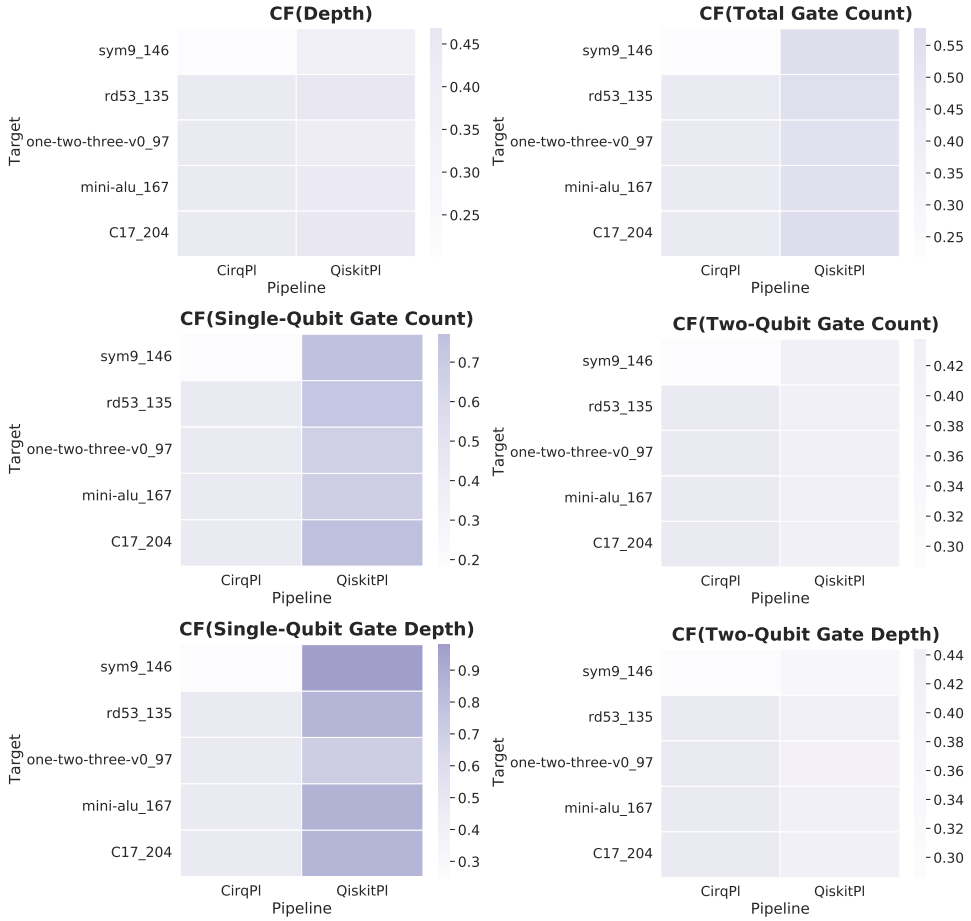


Figure 8.7: Compression factor ($CF$) vs circuit for IbmRueschlikonSymmetrical16Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | sym9_146 | 341 | CirqPl | sym9_146 | 631 |
| Total Gate Count | QiskitPl | mini-alu_167 | 529 | CirqPl | sym9_146 | 1497 |
| Single-Qubit Gate Count | QiskitPl | rd53_135 | 221 | CirqPl | sym9_146 | 980 |

Continued on Next Page

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Two-Qubit Gate Count | QiskitPl | mini-alu_167 | 288 | CirqPl | sym9_146 | 517 |
| Single-Qubit Gate Depth | QiskitPl | sym9_146 | 50 | CirqPl | sym9_146 | 194 |
| Two-Qubit Gate Depth | QiskitPl | sym9_146 | 255 | QiskitPl | C17_204 | 408 |
| Circuit Cost Function | QiskitPl | mini-alu_167 | 4.99 | CirqPl | sym9_146 | 9.339 |
| Execution Time | QiskitPl | mini-alu_167 | 0.892 | CirqPl | sym9_146 | 30.401 |

## 8.2 Hardware: IbmAll2All16Q

### Metrics for each stage of compilation pipeline and aggregate compression factor (initial/final)

Note that in most cases single-qubit gate count and single-qubit gate compression factor have only limited meaning as a metric of compiler performance.This is because single-qubit gate count is very sensitive to the choice single-qubit basis gates (e.g. $U_3$ is equivalent to a combination of 3 rotation gates $R_x$, $R_y$ and $R_z$).

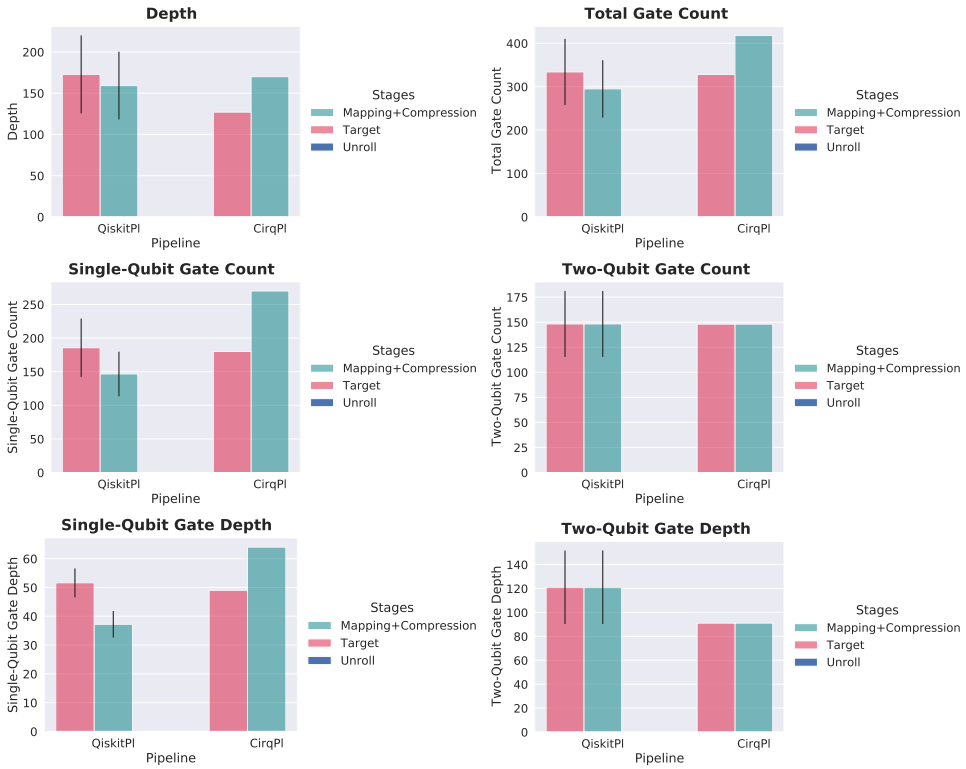


Figure 8.8: Circuits metrics for each compilation pipeline stage for IbmAll2All16Q.

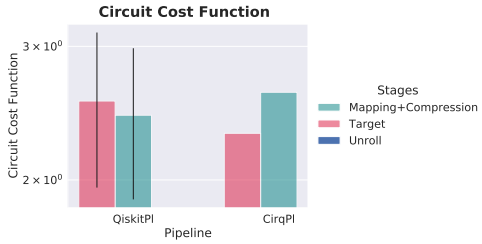


Figure 8.9: Circuit cost function for IbmAll2All16Q.

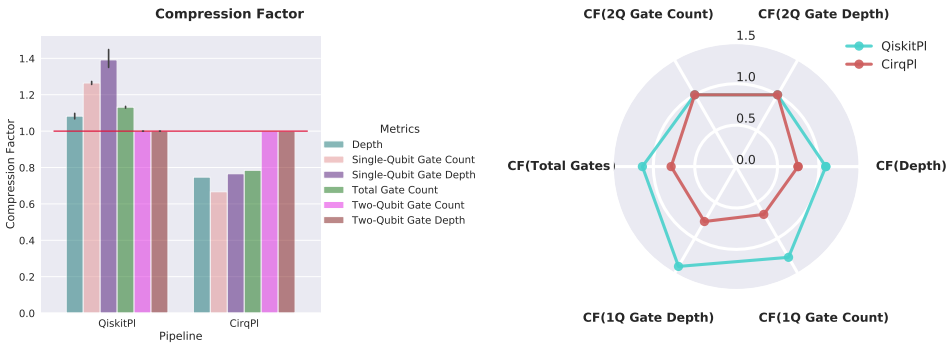


Figure 8.10: Compression factor ($CF$) between target and final compilation stage for IbmAll2All16Q (histogram and radar plot).

## Gate composition for each compilation pipeline stage

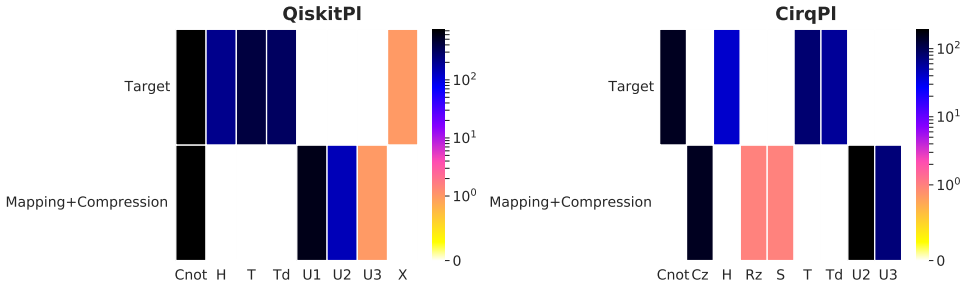


Figure 8.11: Gate frequencies in each pipeline stage for IbmAll2All16Q.

## Execution time stats

Here we present stats about execution time (in seconds) spent by frameworks for each compilation stage.

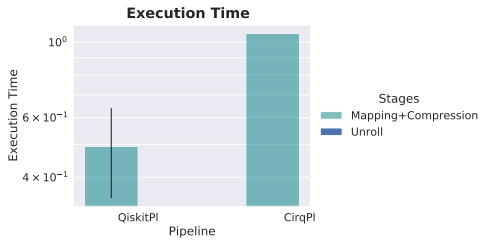


Figure 8.12: Mean execution time of each compilation stage for IbmAll2All16Q.

## Summary stats (averaged over target circuits) by pipeline

| Metrics | Min | Max |
|---|---|---|
| Depth | QiskitPl | CirqPl |
| Total Gate Count | QiskitPl | CirqPl |
| Single-Qubit Gate Count | QiskitPl | CirqPl |
| Two-Qubit Gate Count | CirqPl | QiskitPl |
| Single-Qubit Gate Depth | QiskitPl | CirqPl |

Continued on Next Page

| Metrics | Min | Max |
|---|---|---|
| Two-Qubit Gate Depth | CirqPl | QiskitPl |
| Circuit Cost Function | QiskitPl | CirqPl |
| Execution Time | QiskitPl | CirqPl |

## Cluster analytics

Scatter plots with axes representing: depth, (input/output) single-qubit gate count, (input/output) two-qubit gate count, (input/output) total gate count and execution time.

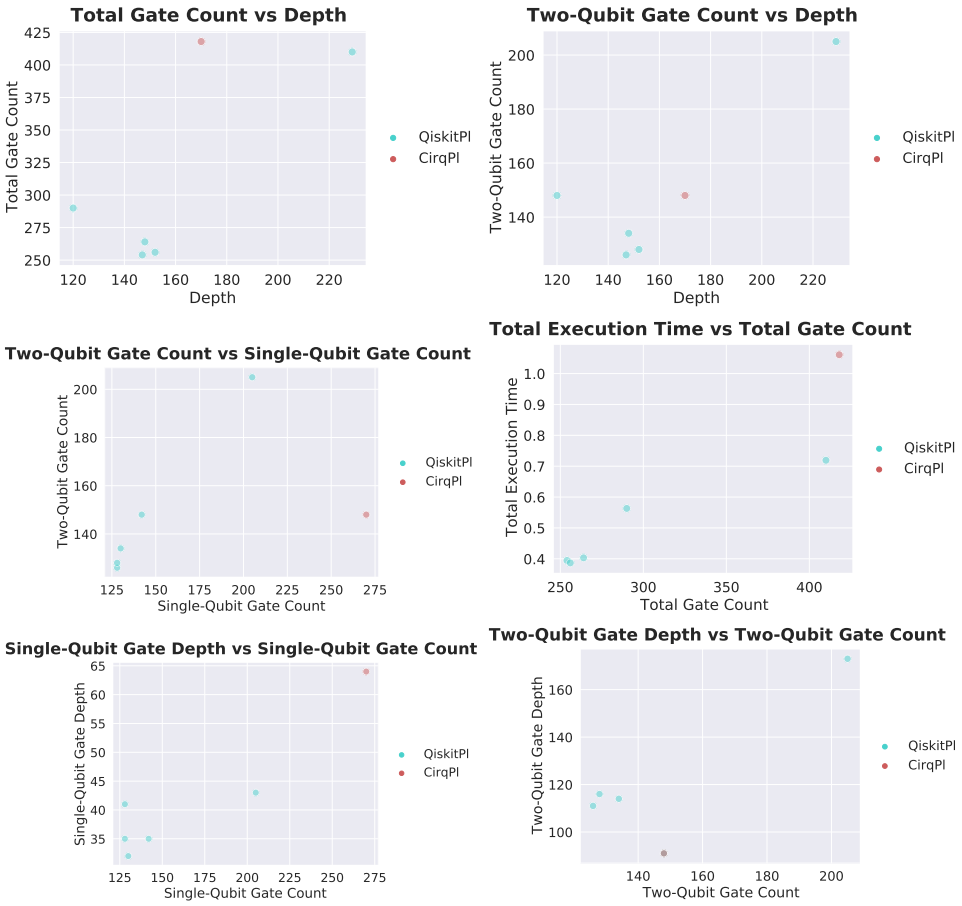


Figure 8.13: Cluster analytics for IbmAll2All16Q. Each point corresponds to an individual target quantum circuit from the target generator.

## Breakdown by individual circuits

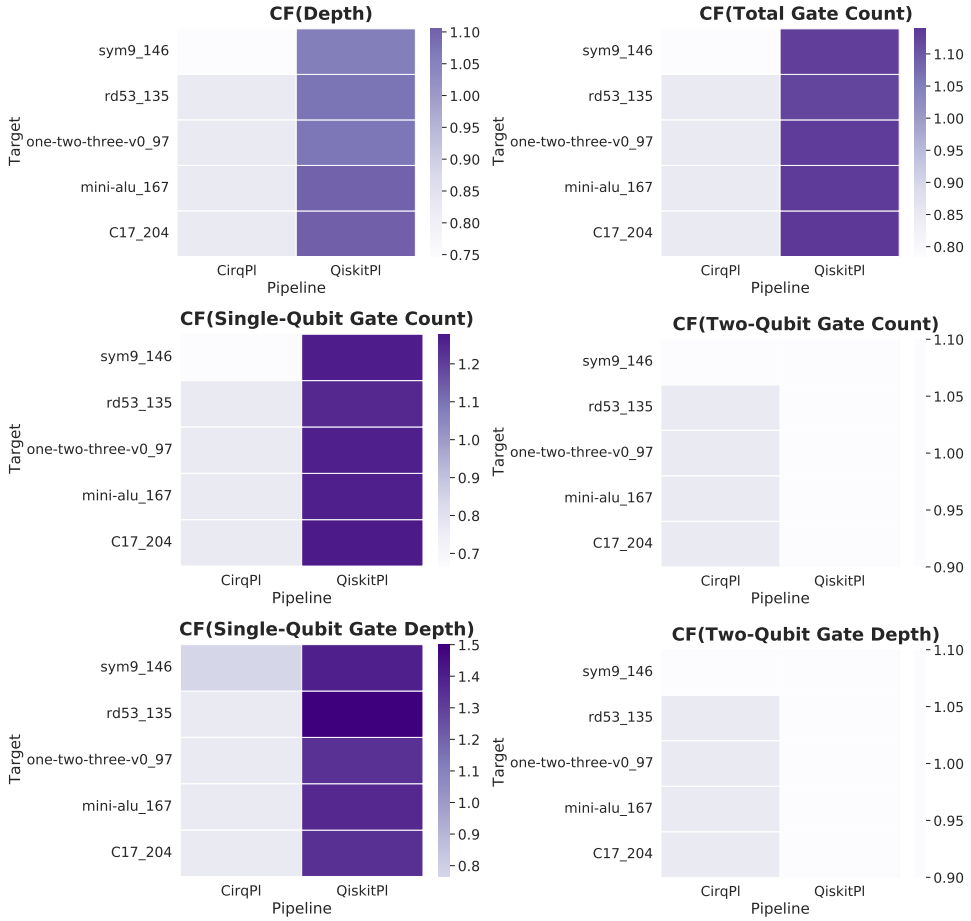


Figure 8.14: Compression factor ($CF$) vs circuit for IbmAll2All16Q.

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Depth | QiskitPl | sym9_146 | 120 | QiskitPl | C17_204 | 229 |
| Total Gate Count | QiskitPl | mini-alu_167 | 254 | CirqPl | sym9_146 | 418 |
| Single-Qubit Gate Count | QiskitPl | mini-alu_167 | 128 | CirqPl | sym9_146 | 270 |

Continued on Next Page

| **Metrics** | **Min** | | | **Max** | | |
|---|---|---|---|---|---|---|
| | **Pipeline** | **Circuit** | **Val** | **Pipeline** | **Circuit** | **Val** |
| Two-Qubit Gate Count | QiskitPl | mini-alu_167 | 126 | QiskitPl | C17_204 | 205 |
| Single-Qubit Gate Depth | QiskitPl | rd53_135 | 32 | CirqPl | sym9_146 | 64 |
| Two-Qubit Gate Depth | QiskitPl | sym9_146 | 91 | QiskitPl | C17_204 | 173 |
| Circuit Cost Function | QiskitPl | mini-alu_167 | 2.131 | QiskitPl | C17_204 | 3.413 |
| Execution Time | QiskitPl | one-two-three-v0_97 | 0.387 | CirqPl | sym9_146 | 1.061 |

## 8.3 Hardware Comparison

This analysis is performed across all classes of target circuit.

| **Metrics** | **Min** | **Max** |
|---|---|---|
| Depth | IbmAll2All16Q | IbmAll2All16Q |
| Total Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Single-Qubit Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Two-Qubit Gate Count | IbmAll2All16Q | IbmAll2All16Q |
| Single-Qubit Gate Depth | IbmAll2All16Q | IbmAll2All16Q |
| Two-Qubit Gate Depth | IbmAll2All16Q | IbmAll2All16Q |
| Circuit Cost Function | IbmAll2All16Q | IbmAll2All16Q |
| Execution Time | IbmAll2All16Q | IbmAll2All16Q |

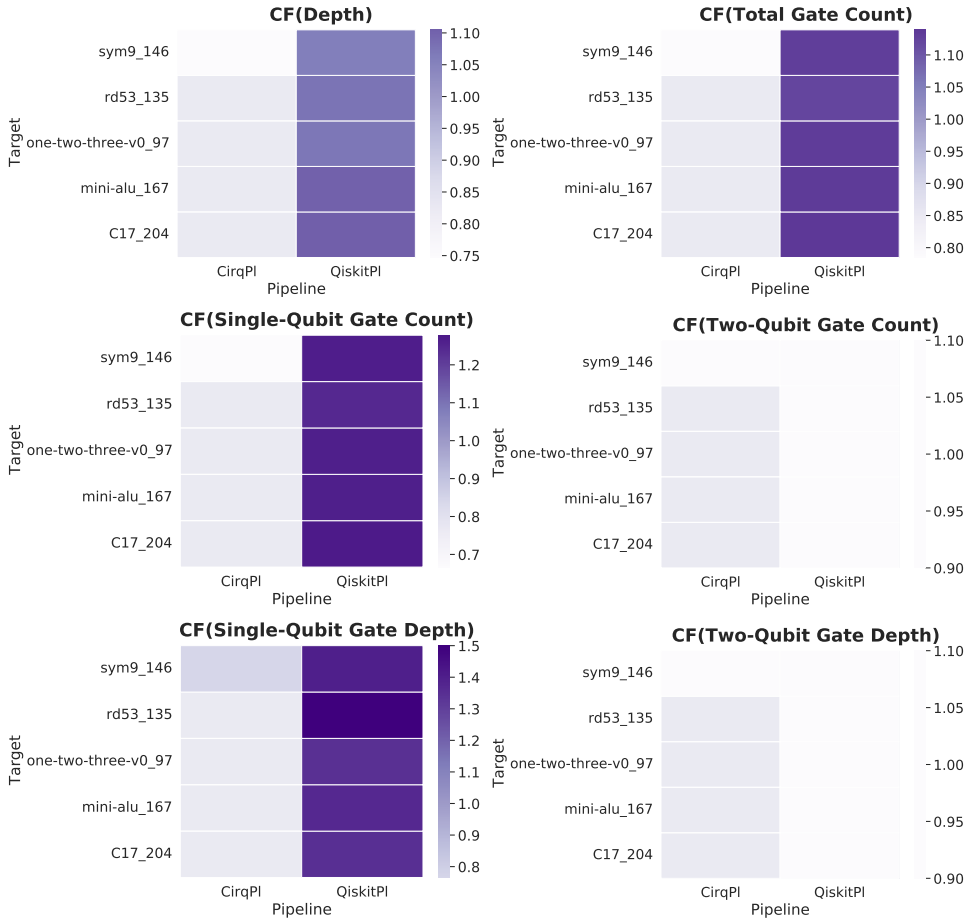


Figure 8.15: Final circuit metrics after compilation vs hardware backend.

# References


- G. Vidal and C.M. Dawson, “A universal quantum circuit for two-qubit transformations with three CNOT gates”, Phys. Rev. A 69, 010301 (2004), https://arxiv.org/pdf/quant-ph/0307177.pdf
- IBM Qiskit repository https://github.com/Qiskit
- Google Quantum AI Lab repository https://github.com/quantumlib/Cirq
- P. Jurcevic et al, "Demonstration of quantum volume 64 on a superconducting quantum computing system", https://arxiv.org/abs/2008.08571 (2020).