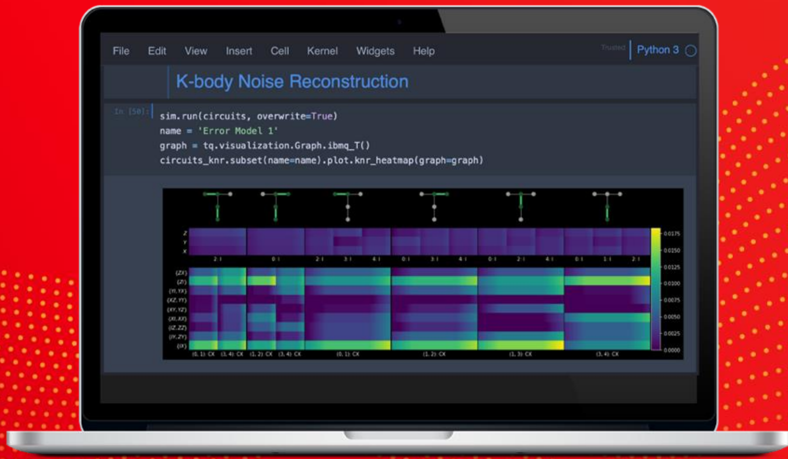

TECHNICAL OVERVIEW

M5402AxxA

Quantum Benchmark

Overview

Current quantum computing architectures are prone to miscalculations due to errors occurring during computation steps. Because these errors limit the performance of quantum computing, they are of central interest for a wide community including hardware makers as well as QC users and developers.

The True-Q software solutions are designed to equip the QC community with state-of-the-art methods to both identify and mitigate performance-limiting effects. True-Q provides methods to measure, assess, calibrate, and optimize the performance of quantum devices. These include diagnostic tools that generate quantum circuits that measure noise properties of the quantum computer. These also include runtime error suppression solutions that tailor and reduce noise profiles.

All of True-Q's tools can be used for any application/algorithm on any gate-based quantum platform including superconducting, ion-trap, and photonic quantum computers. True-Q provides methods for interfacing with other major software platforms for cross-compatibility.

Dr. Thomas Monz

CEO & Chief Scientist, Alpine Quantum Technologies

"Before True-Q, we simply lacked the tools for verifying that we are on the right track towards scalable quantum computing."

KEYSIGHT TECHNOLOGIES

Core Components of True-Q

True-Q consists of four core components:

1. **Error Diagnostics Tools**
   A suite of protocols that diagnose the key aspects of the error profiles. The diagnostic results have several use cases:
   - they can be fed into the True-Q simulator to emulate a real device,
   - they can help hardware developers to efficiently and reliably optimize the performance of their device, and
   - they can serve as a benchmark for circuit design strategies.

2. **Error Suppression Tools**
   A suite of protocols that suppress noise in any quantum device in order to optimize performance at runtime.

3. **Compiler**
   State-of-the-art compiling tools to bridge the gap between abstract quantum algorithms and actual quantum computing instructions. True-Q compiling tools are highly versatile, which makes the True-Q software an ideal platform to explore different circuit design strategies.

4. **Simulator/Emulator**
   A fully customizable quantum emulator informed by real-world error models to simulate a quantum computation on any ideal or error-prone device.

Dr. Matthew Reagor

Director of Engineering, Rigetti

"Now with Keysight, we are actually able to diagnose and correct these errors…at this intermediate scale of algorithms development, partnering with Keysight has actually been really critical in order to maximize the performance of these cloud deployed quantum computers."

Error Diagnostic Tools

In order to improve the performance of error-prone quantum devices, one needs to diagnose the errors affecting the device. True-Q has many tools to characterize errors in quantum computing devices to help hardware developers and end users to better understand and optimize the system. Individual error diagnostic protocols can be used to retrieve specific information about the noise in a system.

1. **Streamlined Randomized Benchmarking (SRB)** provides an estimate of the average fidelity (or probability of correctness) of particular sets of single- or two-qubit gates. Randomized benchmarking is the canonical example of a fast error diagnostic protocol, and our streamlined implementation minimizes the number of circuits needed to run the protocol.

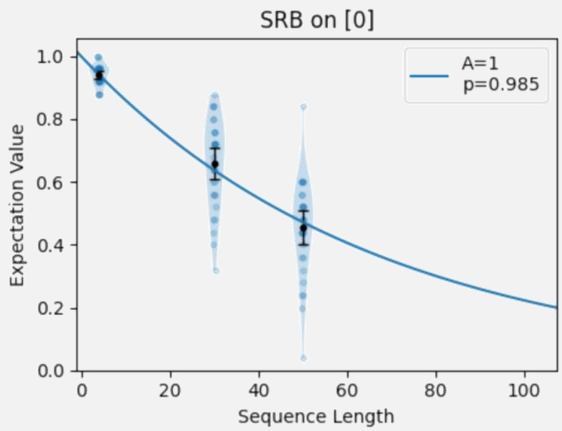

Figure 1. True-Q auto-generated visualization of a Randomized Benchmarking experiment with fitted parameters. Benchmarking protocols fit a decay function of the form y=Apm, where m is the sequence length and p is a measure for the fidelity of the benchmarked operation.

2. **Interleaved Randomized Benchmarking (IRB)** estimates the probability of an error occurring when implementing a user-specified gate. This is useful when noise is not uniform for every operation as it provides users with a more detailed understanding of the error landscape.

3. **Extended Randomized Benchmarking (XRB)** provides an estimate which quantifies how much of the noise is coherent. This is valuable because it informs users how much performance can be gained by suppressing coherent errors using randomized compiling (one of True-Q's proprietary error suppression techniques) and gives an indication of how much of the error is due to calibration, which is exceedingly useful information for hardware developers.

4. **K-Body Noise Reconstruction (KNR)** is a scalable, efficient state-of-the-art protocol which provides a blueprint of the noise acting on a system. More precisely, KNR estimates the probabilities of Pauli errors acting on up to "K" qudits, where K is user-specified. The output can be displayed as a heat map.

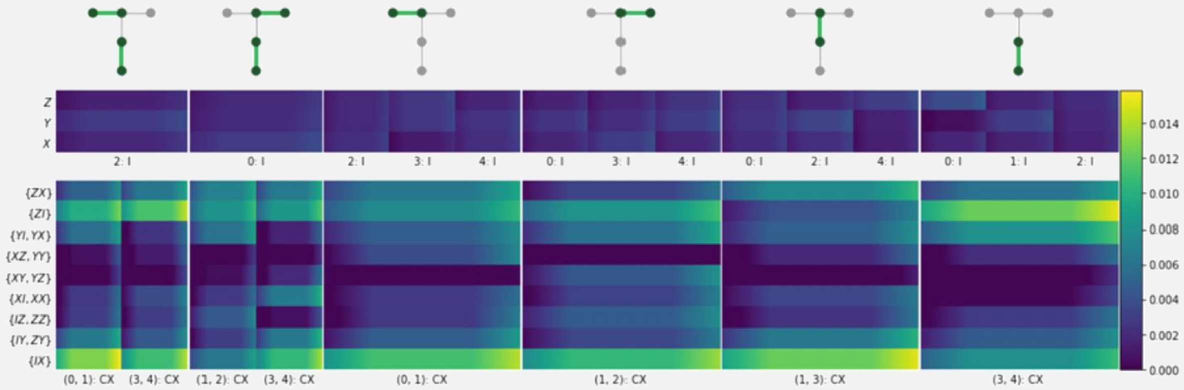

Figure 2. True-Q auto-generated visualization of a K-Body Noise Reconstruction (KNR) experiment.

Higher probability of error is shown on the yellow end of the spectrum and lower probability in deep blue. Each vertical partition corresponds to a cycle with two-qubit gates acting on the highlighted qubits in the T-shaped graph that shows the layout of the simulated hardware. Each horizontal segment corresponds to a Pauli error, labeled on the right.

5. **Crosstalk Diagnostics** locates and quantifies crosstalk errors in a system. Crosstalk is one of the most common and most difficult to characterize sources of noise in quantum devices. It is introduced when an operation is applied to some qubits and other qubits are inadvertently impacted.

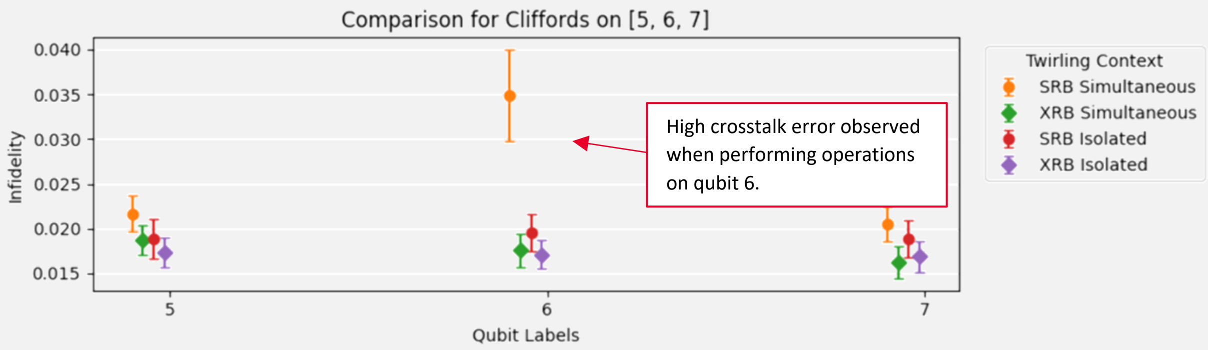

Figure 3. True-Q auto-generated visualization of Crosstalk Diagnostics on three different qubits. Randomized benchmarking protocols are applied both simultaneously to all qubits as well as isolated. A large infidelity in for simultaneous RB indicates a large crosstalk error.

6. **Quantum Capacity (QCAP)** estimates the total error on any user-specified circuit using information from some of our fundamental error diagnostic tools. This is effectively a plug-and-play solution that enables users to determine whether a specific circuit will return reliable results if run on their noisy device.

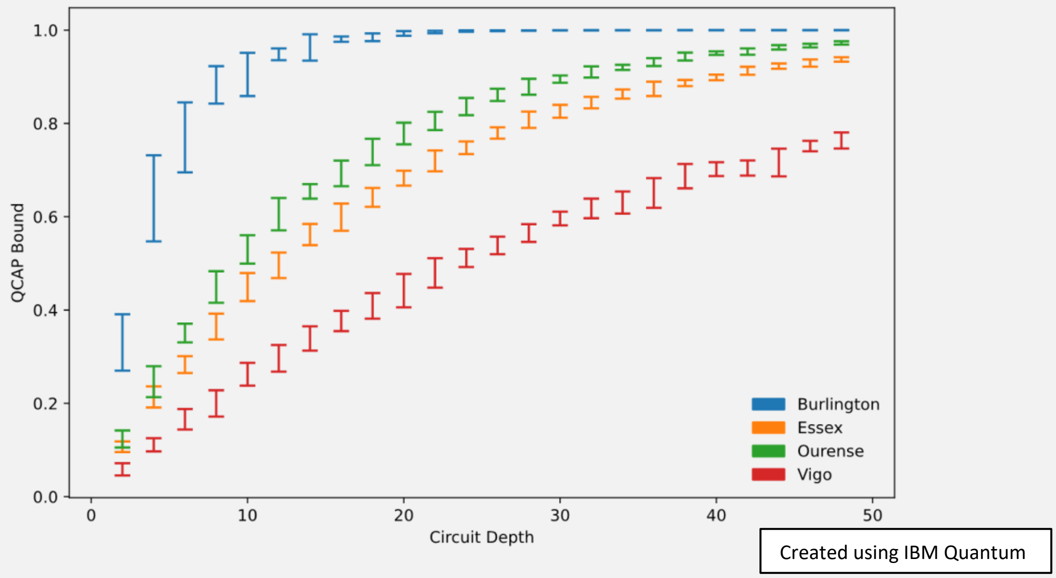

Figure 4. True-Q auto-generated visualization of the estimated Quantum Capacity (QCAP) bounds on four different IBM backends.

7. **Custom Configuration** allows True-Q users to combine the fundamental error diagnostic protocols to retrieve relevant information about the error landscape in a device.

Error Suppression Tools

Quantum computing hardware is highly susceptible to errors. Therefore, error suppression is an integral part of the successful implementation of a quantum computer. The True-Q software suite contains several error suppression protocols to help users to obtain more accurate results when running algorithms on quantum computers.

1. **Randomized Compiling (RC)** converts circuits into a form where coherent noise is suppressed without adding depth to the circuit, that is, without increasing the number of clock cycles needed to implement the circuit. This is important to a successful implementation because coherent errors are more difficult to correct than non-coherent errors, so suppressing that portion of the noise makes the behavior of the noise both more predictable and more correctable. This is one of the most lauded tools in True-Q.
2. **Stochastic Calibration (SC)** allows users to diagnose how errors on an operation affect specific components of a state. The returned information allows users to compensate for the noise by modifying the implementation of gates in a way that offsets the errors.
3. **Readout Calibration (RCAL)** diagnoses and corrects readout errors. Every quantum circuit ends in a readout operation that returns the results of that run. The readout operation is, however, susceptible to noise. This protocol characterizes and corrects for errors introduced by the readout operation. True-Q can apply this correction when implementing any circuits, including circuits used in the error diagnostic and suppression protocols. This includes convenience methods to automatically include readout calibration circuits in every batch of circuits when splitting up a large collection to submit to a device.
4. **Error Mitigation Tools** allow users to mitigate the overall error on a computation through extrapolating the observed noise in the system. We offer two proprietary error mitigation techniques.

Dr. John Martinis

Chief Scientist of Quantum Hardware and leader of Google's Quantum AI Lab

"The randomized compiling method from Quantum Benchmark is an extremely helpful technique for suppressing residual calibration errors. I couldn't imagine building a quantum computer without it."

Compiler

There are two common issues that come up when attempting to run an algorithm on specific hardware platforms:

1. Gates inside of theoretical circuits are not directly compatible with the hardware's native gateset.
2. Gates or circuits are specified as unitary matrices, rather than specific gate decompositions.

If True-Q has access to the configuration of the system, it is possible to solve both issues. Our compilation tools take a circuit and rewrite it using only the gates and connections contained in a user-specified device configuration file so that it can be implemented directly on the corresponding hardware platform.

In addition to True-Q turn-key solutions to concrete compilation issues, the complete True-Q compilation toolkit allows the more advanced QC users to seamlessly customize their own compiler and delve into the craft of circuit design.

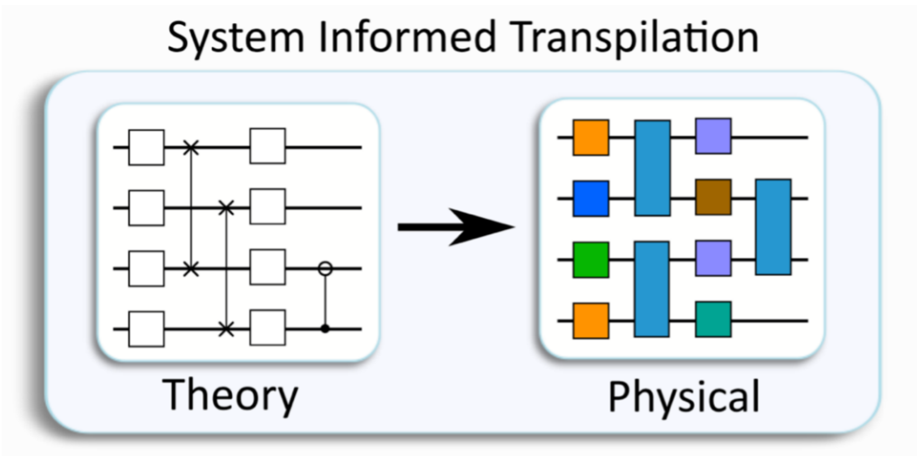

Figure 5. Graphical representation of True-Q circuit compilation capabilities. The gates within the theoretical circuit are converted to physical gates for a specific hardware backend.

Simulator/Emulator

Instead of directly running circuits on quantum hardware, QC users and developers may quickly explore and test their ideas on the True-Q built-in simulator. The simulator acts as a customizable mock device that can incorporate user-specified error profiles or emulate the error profiles observed in hardware through True-Q error diagnostic tools. This feature provides the "quantum experience" for users that do not have access to a real quantum device. It also provides a means to reduce the number of costly physical circuit runs for those who rely on quantum hardware.

More specifically, the simulator can either be noiseless or noisy, and can be used to:

1. Simulate the final quantum state of a computation.
2. Simulate the total effective operator of a circuit.
3. Sample from the output distribution that would be returned by a quantum computer for a given circuit.

Other Features

- **Qudit Support**
  True-Q tools can be applied to traditional quantum bits (qubits), and also to multi-dimensional quantum dits (qudits) to allow users to diagnose and suppress errors in qudit computations.

- **Submission Tools**
  True-Q includes convenience tools that allow users to submit directly to devices which use a Qiskit backend. There is also an automatic batching function that splits collections of circuits into batches of the appropriate size when a device has a limited number of circuits allowed for a single submission.

- **Data Storage and Labeling**
  Results can be stored with circuits. This makes it easy to keep track of which circuit yielded which results, and circuits can be labeled within a collection and retrieved based on that label. This makes it easy for users to organize their results. For example, circuits from different protocols can be stored in the same collection and submitted together. The way that results are stored and labeled ensures that results from different protocols don't get mixed.

- **Visualization Tools and Interactive Graphics**
  True-Q includes many visualization tools, some of which are interactive. For example, mousing over an operation in a circuit drawn using True-Q shows information about that operation.

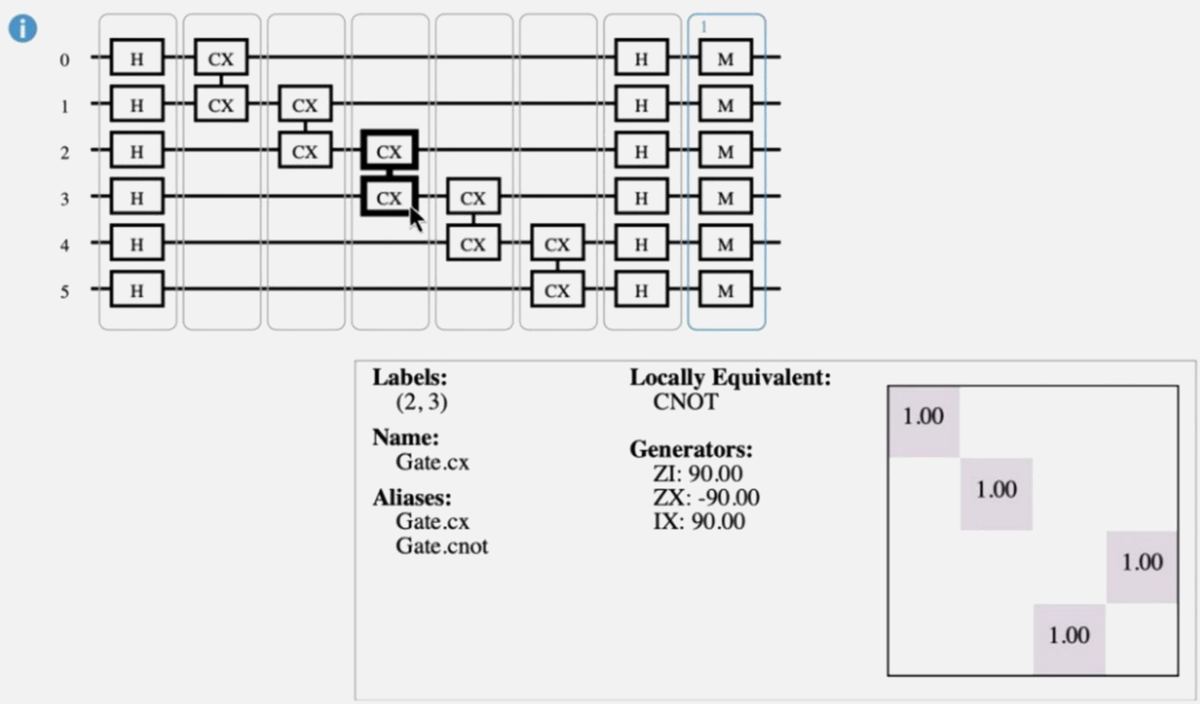

Figure 6. Sample interactive circuit visualization using True-Q. Users can visualize any circuit of interest and obtain additional information about the gates through a hover-over feature.

- **Compatibility with other Quantum Computing Software Packages**
  True-Q contains methods to translate objects between True-Q and other major quantum computing software packages, including Cirq, PyQuil, Qiskit, and QASM.

Procurement Information

Products	Model number
Quantum Benchmark Ultimate enterprise software license for 5 Qubits	M5402A00A
Quantum Benchmark Ultimate enterprise software license for 10 Qubits	M5402A01A
Quantum Benchmark Ultimate enterprise software license for 20 Qubits	M5402A02A
Quantum Benchmark Ultimate enterprise software license for 40 Qubits	M5402A04A
Quantum Benchmark Ultimate enterprise software license for 60 Qubits	M5402A06A
Quantum Benchmark Ultimate enterprise software license for 100 Qubits	M5402A10A
Quantum Benchmark Ultimate enterprise software license for 200 Qubits	M5402A20A

Licensing Terminologies

Technology name	Description
Subscription	Subscription licenses can be used through the term of the license only (6, 12, 24, or 36 months).