Working with IBM Qiskit SDK, the Toolkit for Quantum Computing

By **Dr Kumar Gaurav** - May 5, 2023

Quantum computing is miles ahead of conventional computing when it comes to speed and ability. The IBM Qiskit SDK is a free and open source toolkit for quantum computing, and provides a set of handy tools for developers working with quantum computers.

Quantum computing uses quantum mechanics principles to do certain tasks quicker than conventional computers. Although quantum computing is still in its infancy, it has the potential to transform industries such as cryptography, materials research, and drug discovery.

In a quantum computer, the fundamental unit of information is called a qubit (short for 'quantum bit'). Unlike traditional bits, which are restricted to a value of 0 or 1, qubits can occur in a superposition of 0 and 1. This superposition characteristic is what allows quantum computers to outperform conventional computers in terms of exponential speed. Qubits have another unique quantum feature known as entanglement. This is a phenomenon in which two or more qubits get coupled in such a way that their state can be correlated even though they are physically apart. Superconducting circuits, entrapped ions, and quantum dots are just a few of the physical systems that may be utilised to construct qubits. In terms of scaling, coherence time (how long a qubit can keep its quantum state), and error rates, these physical systems have various strengths and limitations.

The need for quantum computing stems from the fact that some computational problems are exceedingly difficult to solve with traditional computers. These issues frequently include complicated systems such as molecules, substances, and subatomic particles, and thus need a large amount of processing power to solve.

By utilising quantum mechanics concepts, quantum computing can tackle many of these issues significantly quicker than classical computers. For example, in drug development, materials design, and other domains, quantum computing may be used to model the behaviour of molecules and materials. Quantum computers are also capable of helping decrypt some encryption algorithms, which could have consequences for strong encryption and financial transactions.

The exponential expansion of data in current computing systems is another argument for the need for quantum computing. The amount of time required to solve large issues in classical computing rises exponentially with the complexity of the problem. Quantum computing may provide a solution to this difficulty by allowing computations to be performed tenfold quicker than on traditional computers, which can have ramifications for data analysis and machine learning.

So the demand for quantum computing stems from the need to tackle complicated issues that are beyond the capacity of traditional computers, as well as to provide a more efficient method of processing and analysing vast volumes of data.

## Key use cases and application areas of quantum computing
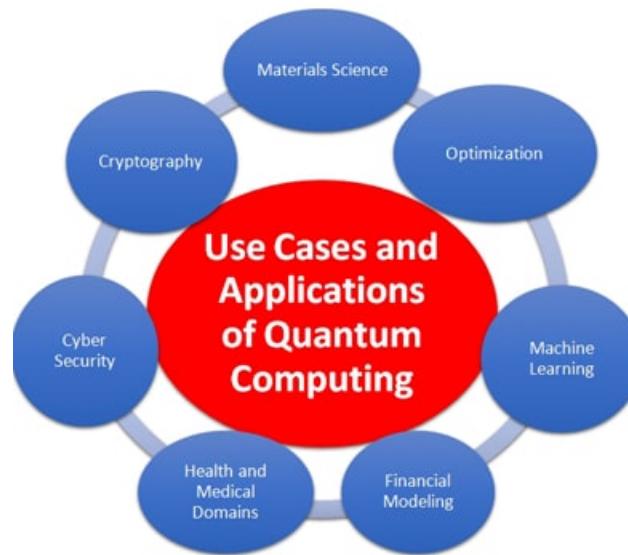
Figure 1: Use cases and applications of quantum computing

Cryptography: Quantum computing has the ability to decrypt today's most widely used encryption algorithms, such as RSA and elliptic curve cryptography. It can also be used to create novel encryption mechanisms that are immune to quantum attacks.

Materials science: Quantum computing may be used to model the behaviour of molecules and materials, which can be beneficial in drug development, materials design, and other domains. Quantum computers may conduct some simulations tenfold quicker than classical computers, allowing researchers to investigate complicated chemical interactions and model the behaviour of materials at the quantum level.

Optimisation: Several real-world problems, including scheduling, transportation, and logistics, may be expressed as optimisation problems. Quantum computing has the potential to handle some of these issues much faster than traditional computers, which can have implications in industries such as finance and aviation.

Machine learning: Some machine learning techniques, such as support vector machines and clustering algorithms, can benefit from quantum computing. Quantum machine learning has the ability to tackle issues that traditional machine learning cannot.

Quantum simulation: Quantum computing can imitate the behaviour of quantum systems including chemical processes and superconductivity. This can reveal details about the behaviour of quantum systems that are challenging or impossible to detect experimentally.
Portfolio allocation and optimisation: Quantum computing may be utilised to optimise risk and return in portfolio allocations. This can aid investors in making more informed investment decisions and risk management. Quantum computing has the potential to transform the financial industry by allowing quicker and more precise calculations and simulations.

Risk management: Quantum computing may be used to model the behaviour of complicated financial systems, potentially assisting financial organisations in risk management. This may involve simulating market collapse behaviour, evaluating the impact of legislative reforms, and enhancing risk management measures.

Fraud detection: Quantum computing may be used to detect suspicious transactions by analysing massive volumes of financial data. This could help prevent financial crimes.

Free and open source tools for quantum computing

A range of tools and platforms is available for the simulation of quantum computing. These open source and free tools are excellent options for researchers and practitioners, and offer valuable resources for building, modelling and running quantum computing programs.
The key free and open source tools available for implementing quantum computing scenarios are listed below.
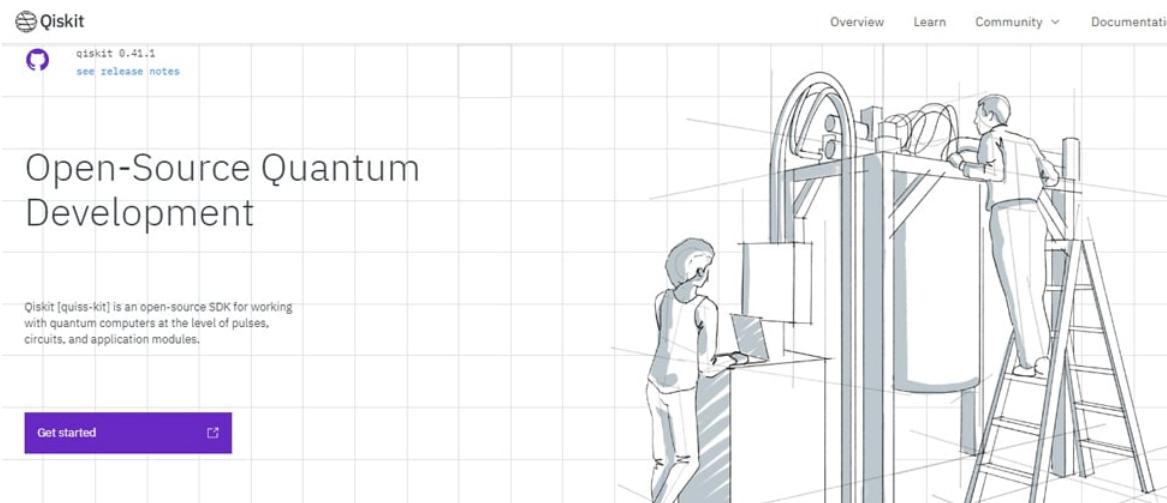
Figure 2: IBM Qiskit platform

IBM Qiskit

https://qiskit.org/

IBM Qiskit is a comprehensive platform under free and open source distribution for quantum computing. It offers tools for creating quantum circuits, simulating them, and conducting experiments on IBM's quantum computers. Lots of libraries and plugins are available in Qiskit to work with high performance quantum scenarios for a range of applications.

PyQuil

https://github.com/rigetti/pyquil

Rigetti Computing's PyQuil is an open source Python toolkit for building quantum applications and interacting with quantum processors. It offers tools for developing quantum programs, simulating them, and interacting with Rigetti's quantum computers.

Google Cirq

https://quantumai.google/cirq

Google Cirq is an open source Python library for creating quantum algorithms. It offers tools for creating quantum programs, simulating them, and interacting with Google's quantum computers.
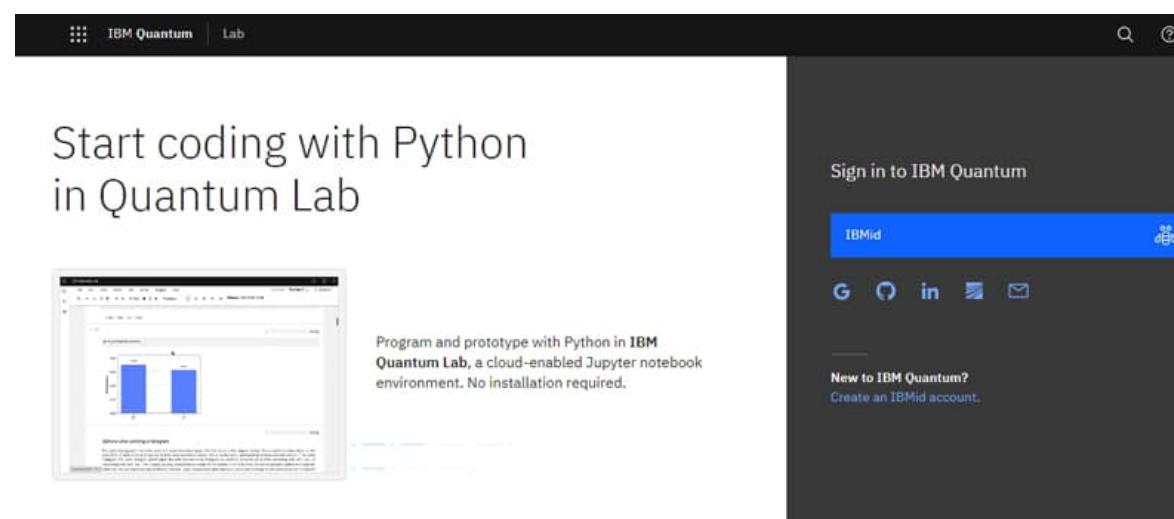

Figure 3: IBM Quantum Lab for simulations of quantum implementations

ProjectQ

https://projectq.ch/

ETH Zurich has created ProjectQ, an open source Python library for quantum computing. It offers tools for authoring quantum programs, simulating them, and interacting with quantum computers.

OpenQASM

https://github.com/openqasm/openqasm

OpenQASM is a free and open source platform for building and running quantum applications. It works in tandem with IBM's Qiskit platform and offers tools for creating quantum circuits, performing simulations, and conducting experiments on IBM's quantum computers.

Qiskit SDK for simulation of quantum scenarios

As stated earlier, Qiskit is an open source quantum computing platform developed by IBM, which contains tools for creating quantum circuits, performing computations, and running operations on IBM's quantum computers. The Qiskit SDK provides a rich set of resources for quantum simulation, enabling researchers and developers to investigate quantum computing possibilities, and build novel algorithms and applications.

High performance tools and libraries integrated with the Qiskit SDK
Qiskit Aer
Qiskit Ignis
Qiskit Aqua
Qiskit Nature
Circuit Composer
Qiskit Terra
Qiskit Pulse
Qiskit Machine Learning

Installation and working with IBM Qiskit

After installation of Python in the system, the IBM Qiskit library can be integrated by running the following command in the command prompt or terminal:

```
$ pip install qiskit
```

With this command, the latest version of Qiskit will be installed with the dependencies.
To ensure that the Qiskit platform is properly installed in the system, the following command is executed on the command prompt or terminal:

```
$ qiskit version
```

After Qiskit is installed, the designing, simulation and execution of quantum circuits can be done very effectively. Qiskit includes a variety of tools for dealing with quantum circuits. Circuit Composer helps to implement this technology, and Aer is used for simulating circuit designs and executing circuits on quantum computers.
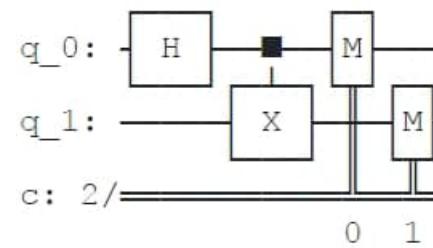

Figure 4: Generation of quantum circuit after execution of code

Here's an example code snippet to create a quantum circuit:

```python
import numpy as np
from qiskit.providers.aer import QasmSimulator
from qiskit import QuantumCircuit, transpile
simulator1 = QasmSimulator()
circuit1 = QuantumCircuit(2, 2)
circuit1.h(0)
circuit1.cx(0, 1)
circuit1.measure([0,1], [0,1])
compiled_circuit1 = transpile(circuit1, simulator1)
job1 = simulator1.run(compiled_circuit1, shots1=1000)
result1 = job1.result()
counts1 = result.get_counts(compiled_circuit)
print("\nCount 00 and 11 :",counts1)
circuit1.draw()
```

This code can be executed on the IBM Qiskit platform and the output circuit will be generated. The code snippet can be executed on the IBM Quantum Lab to visualise the results.

Quantum computing is a continuously emerging technology with enormous research and development potential. New quantum-based algorithms that take advantage of quantum computers' unique capabilities to solve problems quicker and more effectively than classical computers can be developed. Quantum error correction is another field of study, in which methods for detecting and correcting faults in quantum calculations can be developed, which is essential for developing large scale, dependable quantum computers. The opportunities for study and development in this subject will only grow as quantum computers evolve and become more powerful.

Dr Kumar Gaurav

The author is the managing director of Magma Research and Consultancy Pvt Ltd, Ambala Cantonment, Haryana. He has 16 years experience in teaching, in industry and in research. He is a projects contributor for the Web-based source code repository SourceForge.net. He is associated with various central, state and deemed universities in India as a research guide and consultant. He is also an author and consultant reviewer/member of advisory panels for various journals, magazines and periodicals. The author can be reached at kumargaurav.in@gmail.com.

Show/Write Comments