

# クラウド経由で本物の量子コンピュータを動かす

東野 仁政

乱数はスマートフォン・アプリのガチャや金融機関のリスク計算など、さまざまところで使われています。しかし、実際に乱数を生成するのは難しいため、基本的には疑似的な乱数を生成して計算に利用します。ただし、疑似乱数は長い周期で乱数が循環するなどの課題があります。実は量子力学を利用すると、理論的には真の乱数を生成できるため、量子コンピュータによる乱数生成も期待されています。

本稿はIBMがクラウドで公開している本物の量子コンピュータを用いて乱数生成の実験を行います。最初にシミュレータで乱数生成をする量子回路を作成し、動作を確認します。その後、実機で使うためにコードを一部書き換えて、実機で動かします。本稿では、その一通りの手順を解説します。ライブラリはシミュレータと実機の両方に対応するQiskitを利用します。

## ステップ1…量子コンピュータのシミュレータを利用してみる

### ● Qiskitのインストール

実際に量子コンピュータのシミュレータを利用して計算してみましょう。本稿では、IBMが開発しているQiskitという量子プログラミングのライブラリを利用します。Qiskitはオープンソース・ソフトウェアとして公開されており、Pythonで動作します。

実行結果の可視化に便利のため、Google Colaboratory

リスト1 インストールしたQiskitのバージョンを確認する

```
import qiskit.tools.jupyter
%qiskit_version_table
%qiskit_copyright
```

リスト2 量子回路の定義…Xゲート

```
1 from qiskit import QuantumCircuit
2
3 circuit = QuantumCircuit(1, 1) # 量子回路を初期化
4 circuit.x(0) # Xゲートを適用
5 circuit.measure(0, 0) # 測定
6
7 circuit.draw("mpl") # 量子回路を可視化
```

(以降、Colab)で実行します。Colabは、ウェブ・ブラウザ上でPythonを実行・可視化できる環境で、Googleが無料で提供しています。次のURLにアクセスし、[ノートブックを新規作成]をクリックすることでPythonの実行環境が起動します。

<https://colab.research.google.com/>

Colab上で次のコマンドを実行し、Qiskitとその可視化機能をインストールしてください。

```
!pip install qiskit[visualization]
```

Colabでリスト1を実行すると、インストールしたQiskitのバージョンを確認できます。筆者の環境では、図1のように表示されました。

### ● 簡単な量子プログラミング…Xゲートの動作確認

まずは、1量子ビットから動かしてみましょう。量子ビットは $|0\rangle$ で初期化されます。Xゲートを適用し、 $|1\rangle$ に変化することを確認します。リスト2のプログラムを実行してみます。すると、図2の結果が出力されます。

#### ▶ 量子回路の解説

図2について説明します。

#### • 量子ビット

1重線のレーンが量子ビットを表しています。ここでは、左側にqと書かれたレーンです。

#### Version Information

Qiskit Software	Version
qiskit-terra	0.19.2
qiskit-aer	0.10.3
qiskit-ignis	0.7.0
qiskit-ibmq-provider	0.18.3
<b>System information</b>	
Python version	3.7.12
Python compiler	GCC 7.5.0
Python build	default, Jan 15 2022 18:48:18
OS	Linux
CPU	1
Memory (Gb)	12.686653137207031
Thu Feb 10 01:03:01 2022 UTC	

図1 Qiskitのバージョンを表示