

NISQ時代を見据えたバッチ型量子回路シミュレータの開発

高橋 慧智¹, 森 俊夫²

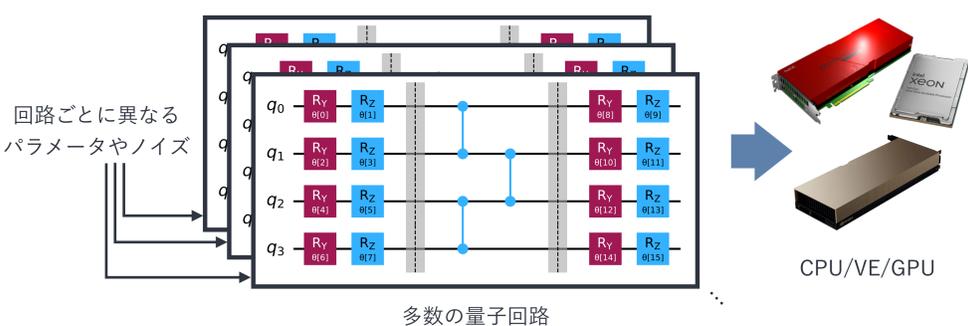
¹ 東北大学 サイバーサイエンスセンター keichi@tohoku.ac.jp

² 大阪大学 量子情報・量子生命研究センター (QIQB) t.mori.qiqb@osaka-u.ac.jp

研究目的

ノイズ耐性量子アルゴリズムや変分量子アルゴリズムの研究開発に資する、多数の量子回路を同時にシミュレート可能なバッチ型量子回路シミュレータの開発を目的とする。量子計算への注目にとともに、古典計算機上で量子計算機を再現し、量子アルゴリズムの開発や評価を可能とする量子回路シミュレータが活発に開発されている。シミュレート可能な量子ビット数を拡大するため、様々な工夫が行われているが、これらのシミュレータのほとんどはノイズが存在しない理想的な量子計算機を再現するものである。

一方、量子計算は萌芽期にあり、短期的にはノイズが大きく量子ビット数も限定的な量子計算機しか実現できない**Noisy Intermediate-Scale Quantum (NISQ)**時代が続くと考えられている。我々は、NISQ時代の量子計算機を対象とした量子アルゴリズムの開発には、次の理由から多数の量子回路を効率的にシミュレートするバッチ型シミュレータが有用であると考えます。

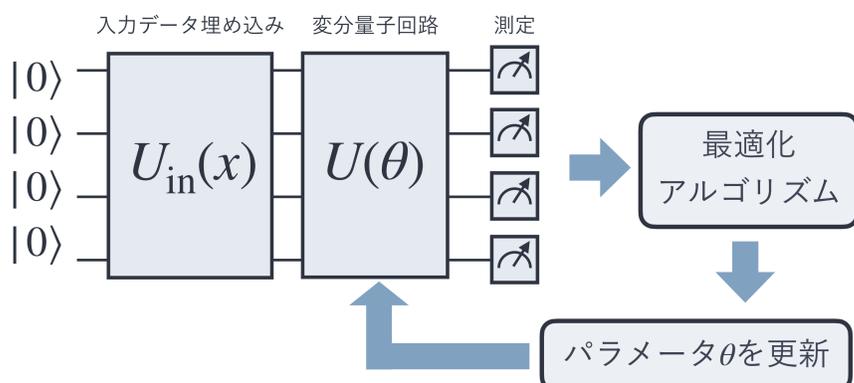


ノイズを有する量子計算機のシミュレーション

NISQ量子計算機を実用化するためには、量子誤り訂正や誤り耐性アルゴリズムなどノイズの影響を抑制するための手法の開発が不可欠である。そのためには実機の量子計算機におけるノイズを再現する必要があるが、従来の密度行列に基づくノイズのシミュレーション手法は時間および空間計算量が大きい。一方、近年提案された**モンテカルロ法に基づくシミュレーション手法**では、それぞれ異なるノイズを注入した多数の量子回路のアンサンブルにより密度行列を推定する。このような手法の効率的な実装には、多数の回路をシミュレーションすることが必須である。

変分量子アルゴリズムの評価

Variational Quantum Algorithm (VQA) は量子・古典ハイブリッドアルゴリズムの一種であり、NISQ量子計算機でも古典計算機に対する優位性を実証できる可能性が期待されている。VQAでは**パラメータ化された量子回路を繰り返し実行**するため、パラメータが異なる多数の量子回路をシミュレートする必要がある。

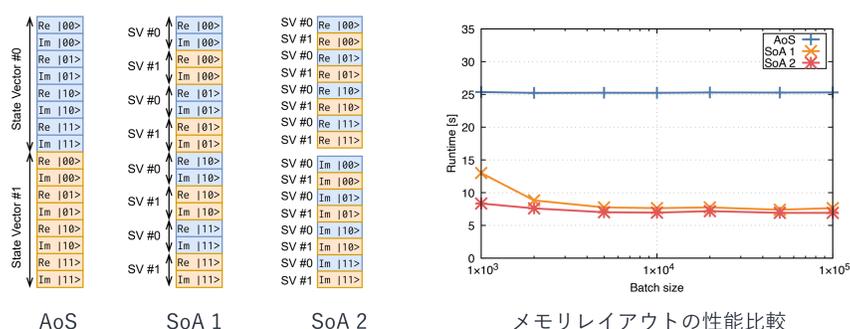


研究計画

本課題では、**NEC SX-Aurora TSUBASA**システム向けに多数の量子回路を同時にシミュレート可能なバッチ型量子回路シミュレータを開発する。具体的には、以下の研究項目に取り組む。

1. バッチ型計算に適する並列化・ベクトル化方式の検討

単一の量子回路のシミュレーションでは状態ベクトルに対する行列ベクトル積を並列化するの一般的だが、多数の量子回路をシミュレーションする場合は、さらに回路レベルでの並列性も利用可能である。これら2階層の並列性を活用し、SX-Aurora TSUBASAのアーキテクチャに適する並列化およびベクトル化方式を検討する。さらに、状態ベクトルのメモリレイアウトも検討する。



2. 多数の量子回路を効率的に表現するためのAPIの設計

ノイズのシミュレーションでは、回路毎に回路中の異なる場所に「ノイズゲート」と呼ばれるゲートを挿入する。また、VQAで用いるパラメータ化量子回路では、回路ごとに量子ゲートのパラメータ（回転ゲートの回転角度等）を異なる値に設定する。このように多数の回路について、少しずつ異なる回路構成を保持する必要がある。回路間で一致する部分を抽出・共通化して保持するなど、ユーザビリティの高さ、メモリ効率の高さ、並列計算方式との親和性等を可能な限り並立できるAPIやデータモデルを設計する。

```
from qulacs import QuantumCircuit, QuantumState
n=3
circuit = QuantumCircuit(n)
circuit.add_H_gate(1)
circuit.add_RX_gate(2, 0.1)
state = QuantumState(n)
circuit.update_quantum_state(state)
```

阪大QulacsのAPI

```
from qiskit import QuantumCircuit
from qiskit_aer import AerSimulator
qc = QuantumCircuit(2)
qc.h(0)
qc.cx(0, 1)
qc.save_statevector()
simulator = AerSimulator(method="statevector")
state = simulator.run(qc).result().get_statevector()
```

IBM QiskitのAPI

3. NISQ向け量子アルゴリズムを用いた性能評価

予備的調査 [1]において単一ゲートレベルやランダム量子回路による動作確認と基礎的な性能評価は完了しているが、実際に研究されているNISQ向け量子アルゴリズムを用いた性能評価が必要である。本課題ではVQAの一種である、量子サポートベクトルマシンや量子ニューラルネットワークなどの量子機械学習アルゴリズムを用い、開発するシミュレータの性能評価を実施する。

