

量子ウォークの多ステップへの拡張とその応用可能性

Expand the implementation of Quantum Walk to any qubits and its possibility of application.

佐藤 綾祐 *

指導教官: ロドニー・バンミーター[†] 佐藤 貴彦[‡]

概要

3量子ビットで実装されているトポロジカル量子ウォークを、任意の量子ビットまで拡張実装し、実際に ibmq_20_tokyo で動かしてその応用可能性を探る.

キーワード: トポロジカル量子ウォーク 量子フーリエ変換 ヒルベルト空間

1 はじめに

量子ウォークとは古典のランダムウォークと対をなすような概念で、いわゆる量子版のランダムウォークである. しかし古典のランダムウォークとは全く異なった性質を持っている. この量子ウォークは行列積の検証 [1] や grid search [2] などといった探索のアルゴリズムにも応用され、時間計算量のオーダーにおいて速度上昇が示されている. また、量子ウォークには「連続時間量子ウォーク」と「離散時間量子ウォーク」があるが、今回は「離散時間量子ウォーク」を扱う.

2 実装の概要

2.1 ヒルベルト空間

ヒルベルト空間とはダフィット・ヒルベルトにその名を因み、ベクトル空間の中でも内積空間が完備性を持つ空間のことを指す. これがどういうことかというと、この空間内のベクトルが連続である必要があるということであり、つまりはそのベクトルの要素が実数または複素数の範囲でなければならないということを意味している. 今回は量子コイン (ランダムウォークでのコインの表と裏の確率を定めるもの) と量子ウォーカー (ランダムウォークで

の移動する人) の2つのヒルベルト空間 ($\mathcal{H}_c, \mathcal{H}_w$) を以下のように定義する.

$$\mathcal{H}_c = \{|x\rangle, x \in \mathbb{Z}_N\} \quad (1)$$

$$\mathcal{H}_w = \{|0\rangle, |1\rangle\} \quad (2)$$

この時 N はコインオペレーターの量子ビットを c , その他の量子ビットを n (全体で $c+n$ 量子ビット) とした時に $N = 2^n$ で表される. また \mathbb{Z} は複素数の集合を表す.

2.2 Step operator (S^\pm)

次に実際にウォーカーがどのように動くのかということを数学的に記述していく. この操作は Step operator と呼ばれ、ウォーカーがどちらに動くのかということを記述している. 今回は円上にある 2^n 個の格子点を考える. この n というのはコインを作らせるビットを除いた量子ビットの数である. まず Shift operator (L^\pm) というものを定義する.

$$L^\pm |x\rangle_w = |(x \pm 1) \bmod N\rangle_w \quad (3)$$

これはある位置 (x) にいるウォーカーが L^\pm という演算子によって、位置が $(x+1)$ へ移動したことを示している. これにより Step operator は、

$$S^+ = |0\rangle_c \langle 0| \otimes L^+ + |1\rangle_c \langle 1| \otimes \mathbb{I} \quad (4)$$

$$S^- = |0\rangle_c \langle 0| \otimes \mathbb{I} + |1\rangle_c \langle 1| \otimes L^- \quad (5)$$

* 慶應義塾大学 Keio University

[†] 慶應義塾大学 Keio University

[‡] 慶應義塾大学 Keio University

と書くことができる. これは以下のような $|\bar{k}\rangle_w$ を用いて次のように表せる. まず

$$|\bar{k}\rangle_w = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} e^{-\frac{2\pi i k x}{N}} |x\rangle_w \quad (6)$$

とする. つまり演算子 $\frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} e^{-\frac{2\pi i k x}{N}} |x\rangle_w$ に作用させると $|\bar{k}\rangle_w$ が得られるということを示している. この時 $|\bar{k}\rangle_w$ は Shift operator L^\pm の固有状態となっており、

$$L^\pm |\bar{k}\rangle_w = e^{\pm \frac{2\pi i k}{N}} |\bar{k}\rangle_w \quad (7)$$

という固有方程式を導くことができる. この固有方程式に対し両辺右側から $\langle \bar{k}|$ をかけて和をとると、

$$\begin{aligned} L^\pm \sum_{x=0}^{N-1} |\bar{k}\rangle_w \langle \bar{k}| &= \sum_{x=0}^{N-1} e^{\pm \frac{2\pi i k}{N}} |\bar{k}\rangle_w \langle \bar{k}| \\ \Rightarrow L^\pm &= \sum_{x=0}^{N-1} e^{\pm \frac{2\pi i k}{N}} |\bar{k}\rangle_w \langle \bar{k}| \end{aligned} \quad (8)$$

という風にかくことができる. (6) より、この式をさらに量子フーリエ変換をかませた形に書き換えることができる.

$$|\bar{k}\rangle_w = U_{QFT} |k\rangle_w \quad (9)$$

となる. ここで U_{QFT} は量子フーリエ変換を表すユニタリー変換とする. ここへ左側から、 $\langle x|$ をかけることによって、

$$\begin{aligned} \langle x| U_{QFT} |k\rangle &= |\bar{k}\rangle_w \langle x| \\ &= \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} e^{-\frac{2\pi i k x}{N}} |x\rangle_w \langle x| \quad (10) \\ &= \frac{1}{\sqrt{N}} e^{-\frac{2\pi i k x}{N}} \end{aligned}$$

ここから L^\pm は (8) (10) を用いて次のように書き換えることができる.

$$L^\pm = U_{QFT} \prod_{j=0}^{n-1} e^{\frac{\mp \pi i Z_j}{2^{j+1}}} U_{QFT}^\dagger \quad (11)$$

そして (4)(5)(11) から、Step operator (S^\pm) は、

$$\begin{aligned} S^\pm &= U_{QFT} S^\pm U_{QFT}^\dagger \\ &= \prod_{j=0}^{n-1} e^{\frac{\pi i (Z_c \mp Z_j)}{2^{j+2}}} e^{-\frac{\pi i Z_c Z_j}{2^{j+2}}} \end{aligned} \quad (12)$$

となる. ただしここでの Z_c はコインを作用させている量子ビットへの Z 軸回転の量子ビット操作、 Z_j は j 量子ビットへの Z 軸回転の量子ビット操作とする.

2.3 Coin operator (τ_i)

最後に量子ウォーカーが進む方向（今回の場合は 2 通り）を記述する Coin operator について見ていく. まず次のような行列を定義する.

$$T(\theta) = e^{-i\theta Y} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \quad (13)$$

ここで θ は N に応じて以下のように定義される.

$$(\theta_1, \theta_2) = \begin{cases} (\theta_1^-, \theta_2^-), & 0 \leq x < \frac{N}{2} \\ (\theta_1^+, \theta_2^+), & \frac{N}{2} \leq x < N \end{cases} \quad (14)$$

この \pm というのは、先に進むのか後に戻るのかということを示している. そして (13)(14) より、2 つの Coin operator は、

$$\begin{aligned} \tau_i &= \sum_{x=0}^{N-1} T(\theta) \otimes |x\rangle_w \langle x| \\ &= \sum_{x=0}^{N-1} e^{-i\theta_i Y_c} \otimes |x\rangle_w \langle x|, \quad i = 1, 2 \end{aligned} \quad (15)$$

となる.

3 実装

3.1 実装環境

今回は以下のような環境で実装を行なった.

- Python (3.6.5)
- Qiskit (0.6.1)
- ibmq_20_Tokyo (real quantum machine)
- ibmqx_hpc_qasm_simulator

3.2 実装

今回は Qiskit を用いて、IBMQ20 量子ビットマシンおよびシミュレータで任意の量子ビットに対して量子ウォークができるように拡張実装を行なった。以下は5量子ビットへの拡張例である。今回はこれを任意の量子ビットに拡張できるような実装を行なった.[図 1]<https://github.com/Chibikuri/QuantumWalk>

また、今回コインパラメータとして

$$(\theta_1, \theta_2) = \begin{cases} (-\frac{1}{8}, -\frac{1}{8}), & 0 \leq x < \frac{N}{2} \\ (-\frac{1}{8} - \pi, \frac{1}{8} + \pi), & \frac{N}{2} \leq x < N \end{cases} \quad (16)$$

を用いている。

4 結果

以下に 3 量子ビット、4 量子ビット、5 量子ビットの拡張による結果を示す。(全て始点は x=1)

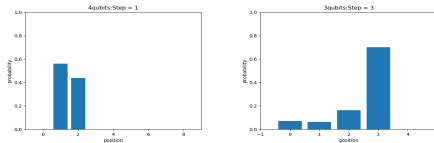


図 1 3 量子ビット:step1(左) step3(右)

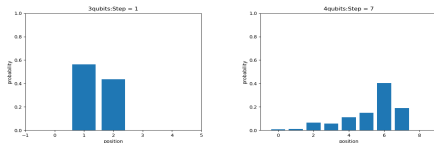


図 2 4 量子ビット:step1(左) step7(右)

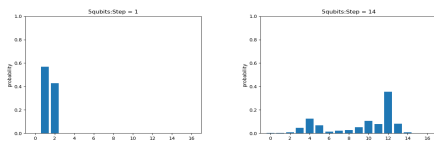


図 3 5 量子ビット:step1(左) step14(右)

5 考察

今回は 1 ステップのトポロジカル量子ウォークを任意のステップ数に拡張できるように実装を行なった。左にはシミュレータの結果のみ載せた。この結果から確率が外側に向かって広がっているのが確認できる。今回は円上にある格子点を考えているため、端の点で円状に繋がっている。6 量子ビット以降で、step 数を増やしていくと、デバイスのスペック上結果を得ることができなかった。また ibmq_20_tokyo で測定を行なってみたが、目立った収束は見ることができなかった。これは現状のデバイスが NISQ デバイスと呼ばれるエラーの多いデバイスであることに起因すると考えられる。

6 今後の課題

今後の課題として、以下に 2 点挙げる。

- 2 次元平面格子状での、4 方向ウォークへの拡張
- グラフカーネルへの応用

参考文献

- [1] Harry Buhrman, Robert Spalek(2004) Quantum Verification of Matrix Products arXiv:quant-ph/0409035
- [2] Matthew Falk (2013) Quantum Search on the Spatial Grid arXiv:1303.4127[quant-ph]
- [3] Radhakrishnan Balu, Daniel Castillo, George Siopsis(2017) Physical realization of topological quantum walks on IBM-Q and beyond arXiv:1710.03615[quant-ph]
- [4] Sansoni L1, Sciarrino F, Vallone G, Mataloni P, Crespi A, Ramponi R, Osellame R. (2012) Two-particle bosonic-fermionic quantum walk via integrated photonics.[PhysRevLett.108.010502]
- [5] 今野 紀雄「量子ウォーク」森北出版 (2014)