量子テレポテーション及びオラクル問題のための量子回路自動生成

矢吹 太朗　　伊庭 斉志

東京大学新領域創成科学研究科基盤情報学専攻
yabuki@miv.t.u-tokyo.ac.jp, http://www.miv.t.u-tokyo.ac.jp/ibalab/

概要　量子アルゴリズムの具体的な表現である量子回路を、遺伝的アルゴリズムによって自動的に生成する方法と、それを量子テレポテーションの回路とオラクル問題（database search, early promise）を解く回路の生成に適用した結果を示す。特に量子テレポテーションについてはオリジナルなものよりも単純な回路が得られている。

キーワード　遺伝的アルゴリズム、量子回路、量子テレポテーション、オラクル問題

1　はじめに

1994 年に Shor が因数分解を多項式時間で行うアルゴリズム [5] を発見して以来、量子計算は大きな注目を集めている。現在は実験的に小規模な量子計算を行える段階にあるが、ある量子アルゴリズムを小規模な、つまり少ない qubits とゲートしか使わないような量子回路で表現する一般的な方法は知られていない。また、ある問題の古典的複雑さと量子的複雑さの違いを知る方法も未知で、量子計算機ならば古典計算機よりも効率良く解けると考えられている問題の数は少ない。

そこで我々は遺伝的アルゴリズム [10] を用いて量子回路を自動的に生成する手法を提案する。遺伝的アルゴリズムはランダムな解の候補から始め、回路の出力を見て解の候補を評価するだけで探索を進めることができるため、問題についての深い知識を必要としないが、以下に示すように、ある問題についてはオリジナルな回路よりもゲートの種類も数も少ないものを生成することに成功している。また遺伝的アルゴリズムと同じく進化論的計算手法のひとつである遺伝的プログラミングによって、古典アルゴリズムよりも効率の良い量子アルゴリズムを発見した例もある [6]。

2　問題設定

2.1　オラクル問題

オラクル問題とは、与えられたブラックボックス（オラクル）の性質を決定する問題である。ここで扱うのは以下の 2 種類のオラクルである。

2.1.1　Database Search Problem

ある数が入力された場合にのみ、出力用の qubit を反転するオラクルを考える。例えば 0, 1 番目の qubits [1]に 00, 01, 10, 11 のいずれかを入力するが、10 を入力した場合にのみ、2 番目の qubit を反転するようなオラクルは、

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

というユニタリ変換で表現できる。

Database search problem は、この種のオラクルが与えられたときに、何を入力するとビットが反転するかのを求める問題である。上の例のオラクルが与えられた場合には 10 を出力するような回路が得られれば良い[2]。古典的なアルゴリズムではオラクルを $O(n)$ 回呼び出さなければならないが、量子アルゴリズムならば $O(\sqrt{n})$ 回の呼び出しで良いことが示されている [4]。

---

[1] この論文では qubits は 0 番から数えることとし、数式においては右端が、回路図においては一番下が 0 番目とする。ゲートを表す記号は L, R, H, S, T, CNOT, M がそれぞれ $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$, $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$, $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$, $\begin{pmatrix} i & 0 \\ 0 & 1 \end{pmatrix}$, $\begin{pmatrix} -1 & 0 \\ 0 & -i \end{pmatrix}$, 制御 NOT, 観測を表す。添え字は作用する qubits を表す。ただし、CNOT の添え字は左側が target である。

[2] ここで扱うのは 2 ビット入力のオラクルで、0, 1 番目の qubits を入力用に、2 番目の qubit を出力用にする。結果は 1, 2 番目の qubits から読み取る。

2.1.2 Early Promise Problem

2 種類のオラクルを考える。Uniform なオラクルはすべての入力に対して何もしないか、すべての入力に対して出力用の qubit を反転させるかのいずれかである。Balanced なオラクルはあり得る入力の半分に対して何もしないが、残りの半分に対しては出力用の qubit を反転させる。

Early promise problem は与えられたオラクルが uniform(0) なのか balanced(1) なのかを決定する問題である。例えば 00, 01, 10, 11 の入力に対して反転, 不変, 不変, 反転と振舞うオラクルが与えられた場合に、1(balanced) を出力するような回路が得られれば良い[3]。決定的古典アルゴリズムでこの問題を解くには、最悪の場合、オラクルを $(2^n/2+1)$ 回呼び出さなければならない。確率的古典アルゴリズムは、同じ出力が $2^n/2$ 回続くことが稀なためにいくらか良いが、いずれにしても量子アルゴリズムが古典アルゴリズムよりも優れていることが示されている [3]。

2.2 量子テレポテーション

量子テレポテーションは、2 つの古典的なビットを転送するだけで、qubit の状態を伝えることができる情報伝達法である [1]。その基本的なアイディアを図 1 に、具体的な回路を図 2 に示した [2]。

問題は次の 4 つの条件のみから量子テレポテーションの回路を自動的に生成することである。1) 3-qubit のシステムである。2) EPR pair 生成部では 0, 1 番目の qubits しか操作できない。3) Alice は 1, 2 番目の qubits しか操作できない。4) 観測は 1 回のみで、1, 2 番目の qubits を測定する。

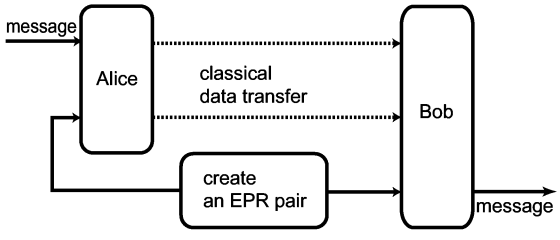

図 1: Idea of the quantum teleportation.

[3]Database search problem と同じ形のオラクルを用い、結果は 2 番目の qubit から読み取る。

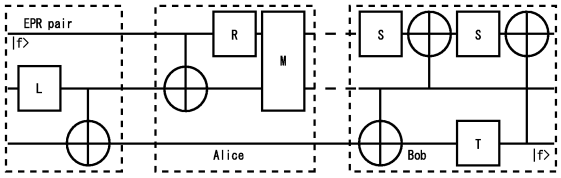

図 2: Brassard's circuit for the teleportation[2].

3 自動回路生成法

3.1 遺伝的アルゴリズム

遺伝的アルゴリズムは生物の進化を模倣した最適化手法で、その基本的な手続きは以下のとおり。

1. 最適化したい問題の解の候補を文字列（遺伝子）に写像する規則を定める。
2. 複数の遺伝子をランダムに生成し、それらを初期世代とする（その要素を個体と呼ぶ）。
3. 各遺伝子が表す解の候補が解にどの程度近いか（適合度）を計算する。
4. 適合度の高い個体を選ぶ。
5. 選ばれた個体の遺伝子の一部を書き換える（突然変異）、2 つの遺伝子を混ぜ合わせる（交叉）などの操作によって次の世代を生成し、3 に戻る。

3.2 遺伝子

オラクル問題のための遺伝子は 101| 112| 220| 211| 201| 112| 021| 210 のような $\{0,1,2\}$ のいずれかの文字からなる固定長の文字列である。遺伝子は表 1 にしたがって 3 文字単位で解釈され、1 つの量子回路を表す。例えば上に挙げた遺伝子は L_0, L_1, H_2, ORACLE, H_0, L_1, $CNOT_{21}$, H_1 を表す。量子テレポテーションの回路生成には表 1 とは異なり 2.2 に挙げた条件を組み込んだ表を用いた [11][9]。

3.3 適合度関数

オラクル問題の場合、1) 1/2 より大きい確率で正しく性質を決定できたオラクルの数、2) 正答確率、3) 回路のサイズの順に個体を評価する。適合度 f を

		2 文字目			
		0	1	2	
1 文字目	0		CNOT_{10}	CNOT_{20}	0
		CNOT_{01}		CNOT_{21}	1
		CNOT_{02}	CNOT_{12}		2
	1		L_1	L_2	0
		L_0		L_2	1
		L_0	L_1		2
	2	ORACLE	H_1	H_2	0
		H_0	ORACLE	H_2	1
		H_0	H_1	ORACLE	2

表 1: Genetic code table.

$$f = 10\left(\tfrac{1}{2} < p_i\text{なる } i \text{ の数}\right) + \sum p_i$$

とする[4]（p_i は i 番目のオラクルの性質を正しく決定できる確率。）。さらに、すべてのオラクルについて 1/2 より大きい確率で正答した場合には f に $(A/\text{ゲート数})$ を加える[5]。

量子テレポテーションの場合、初期状態が $(p|0\rangle + q|1\rangle) \otimes |0\rangle \otimes |0\rangle = (p\ 0\ 0\ 0\ q\ 0\ 0\ 0)$ とすると、望ましい終状態は $(a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle) \otimes (p|0\rangle + q|1\rangle) = (ap\ aq\ bp\ bq\ cp\ cq\ dp\ dp)$ である。解の候補の回路の出力を $(a_0 \cdots a_7)$ とし、

$$error = \sum_{i=0,2,4,6} \left| \frac{a_i}{a_{i+1}} - \frac{p}{q} \right| \ (a_{i+1} \neq 0 \text{ ならば})$$

のような関数で望ましい終状態との差を表す。適合度 f を $f = 1/(1 + error)$ のように定めると差が小さくなれば f は大きくなる。この f が 1 の時、つまり正しい回路が得られている時は f に $(1/\text{ゲート数})$ を加えて、回路のサイズが小さければ適合度がさらに大きくなるようにする。

4 結果と考察

Database search problem を解く回路（図 3）はオラクルを 1 回呼び出すだけで必ず正答を出力する。Spector らは遺伝的プログラミングによって生成された回路を人手で修正して同様（ゲートの種類が 2 つ多い）な回路を得ている [7]。

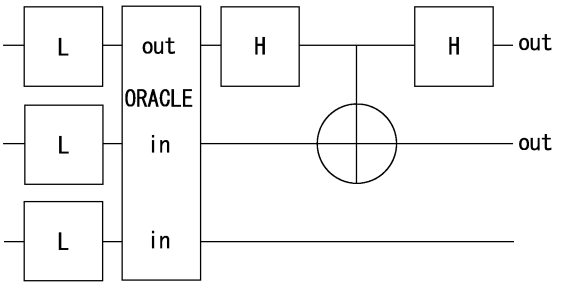

図 3: Evolved circuit for Database search problem.

彼らはまた、オラクルを 1 回呼び出すだけで early promise problem に 0.7 程度の確率で正答する回路を得ている [7]。これに対し我々の回路（図 4）はオラクルを 2 回呼び出すが必ず、つまり確率 1 で正答する。

量子テレポテーションについては得られた図 5 の回路は、オリジナルなもの（図 2）に比べてゲートの数も種類も少ない。長さ 60（ゲートは最大で 20 個）の遺伝子を使った場合、常にゲート数 8 の回路が生成された。この回路は平均 350 世代（標準偏差 280）で発見できているが、それまでに試す回路は $5,000(\text{個体}) \times 350(\text{世代}) \sim 1.8 \times 10^6$ 通りである。一方、ゲートは全部で 14 種類あるため[6]、回路のすべての可能性は $20^{14} \sim 10^{18}$ 通りである。解がどの程度の密度で存在するかわからないため、これらの数値を直接比べることはできないが、この手法の効率は悪くないと思われる。

Williams らは量子テレポテーションをユニタリ変換・観測・ユニタリ変換という 3 つの部分に分け、ユニタリ変換の部分を遺伝的アルゴリズムを使って合成している [8]。目標のユニタリ変換を与えて進化させていることと、ユニタリ変換としては正しいが、量子テレポテーションとしては適当でない回路を生成してしまう可能性があるという点で彼らの方法には不満がある [11][9]。それに対し我々は、より少ない知識からより良い回路を生成させることに成功している。

[4] Early promise problem の場合、オラクルは 8 種類あるが、そのうち uniform なのは 2 個だけなので、このままでは回路が balanced という結果を出す方向に偏って進化してしまう可能性がある。それを避けるために uniform なものに関しては 3 倍の重みをかけた。

[5] A は定数で Database search problem の場合は 40、Early promise problem の場合は 120 とした。

[6] ここで使ったゲートは CNOT が 6 通り、L, R がそれぞれ 3 通り、そして観測と区切りである [11][9]。

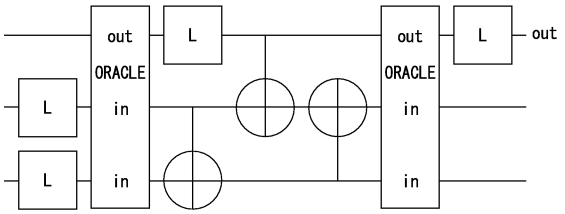

図 4: Evolved circuit for Early promise problem.

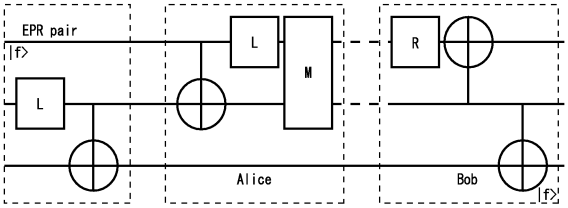

図 5: Evolved circuit for the teleportation.

5 おわりに

遺伝的アルゴリズムを使って量子回路を自動的に生成する方法と、それをいくつかの問題に適用した結果を示した。遺伝的アルゴリズムなどの進化論的計算によって、それまで知られていたよりも単純な回路を生成できる可能性があり、このことは問題の量子計算的複雑性の解明にも役立つと期待される。使用するゲートの種類や qubit 数[7]などの仮定を取り除くこと、それによって広くなる探索空間を遺伝的アルゴリズムに向くように調整することが今後の課題である。

参考文献

[1] C. H. Bennett, et al. Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Phys. Rev. Lett.*, Vol. 70, p. 1895, 1993.

[2] G. Brassard. Teleportation as a quantum computation. In *Proceedings of the Fourth Workshop on Physics and Computation*, p. 44. New England Complex Systems Institute, 1996. quant-ph/9605035.

[3] D. Deutsch. Quantum theory, the church-turing principle and the univarsal quantum computer. In *Proceeding Royal Society London*, Vol. A400, p. 97, 1985.

[4] L. Grover. Quamtum mechanics helps in searching for a needle in a haystack. *Phys. Rev. Lett.*, Vol. 79, p. 325, 1997. quant-ph/9706033.

[5] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal of Computing*, Vol. 26, p. 1484, 1997. quant-ph/9508027.

[6] L. Spector, et al. Finding a better-than-classical quantum and/or algorithm using genetic programming. In *Proceedings of the 1999 Congress on Evolutionary Computation*. IEEE Press, 1999.

[7] L. Spector, et al. Quantum computing applications of genetic programming. In *Advances in Genetic Programming 3*, p. 135. MIT Press, 1999.

[8] C. P. Williams and A. G. Gray. Automated design of quantum circuits. In *QCQC'98 LNCS 1509*, p. 113. Springer-Verlag, 1999.

[9] Taro Yabuki and Hitoshi Iba. Genetic algorithms for quantum circuit design -evolving a simpler teleportation circuit-. In *Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference*, p. 425, 2000. http://www.miv.t.u-tokyo.ac.jp/~yabuki/paper/2000/gecoo/yabuki-gecco-late-425-2000.pdf.

[10] 伊庭斉志. 遺伝的アルゴリズムの基礎. オーム社, 1994.

[11] 矢吹太朗, 伊庭斉志. 遺伝的アルゴリズムによる量子回路生成. 信学技報 Vol.100 No.89, p. 9. 電子情報通信学会, 2000. http://www.miv.t.u-tokyo.ac.jp/~yabuki/paper/2000/ieice/yabuki-eic-100-89-2000.pdf.

[7] ここでは 3-qubit に限定して探索した。