A-2 クリーク発見問題に対する量子アルゴリズム
A quantum algorithm for clique finding problem

中川 仁* Hitoshi Nakagawa　谷 聖一† Sei'ichi Tani　原 正雄‡ Masao Hara　山本 慎§ Makoto Yamamoto

1 はじめに

Brassard らは，Grover のデータベース検索量子アルゴリズム [4] の一般化および振幅の増幅手法を提案した [2, 3]．本論文では，その手法を応用して，l-クリーク発見問題，l-サイクル発見問題，二部グラフ判定問題に対する量子アルゴリズムを提案し，それらの計算時間を評価する．

2 Grover のデータベース検索アルゴリズム

$N = 2^n$ 個のデータベース $\{S_0, S_1, \ldots, S_{N-1}\}$ のなかから与えられた条件を満たす唯一の要素を発見する問題をデータベース検索問題という．

Grover は，各要素が検索対象かどうかを 1 ステップで判定できるオラクル関数 f （x が検索対象のとき $f(x) = 1$, そうでないとき $f(x) = 0$）を利用できるとき，既知の古典アルゴリズムより高速に検索問題を解く **Algorithm 1** を提案した [4].

Algorithm 1 (グローバーのデータベース検索アルゴリズム)

Step 1. $|0\rangle^{\otimes n}$ に Walsh-Hadamard 変換 $\mathbf{W}^{\otimes n}$ を適用し，N 個の状態 $|x_i\rangle$ の振幅を全て等しく $\frac{1}{\sqrt{N}}$ とする．

Step 2. 以下のユニタリ変換 (a), (b) を $\lfloor \frac{\pi}{4}\sqrt{N} \rfloor$ 回適用．

(a) 全ての状態 x_i に対して，

$$f(x_i) = \begin{cases} 1 \text{ ならば } phase \text{ を} \pi \text{回転} \\ 0 \text{ ならば何も変えない} \end{cases}$$

(b) 次のように定義される拡散変換 D を適用．

$$D_{ij} = \frac{2}{N} - \delta_{ij}$$

Step 3. 観測する．

Step 2 の (a), (b) をユニタリ変換を $\mathbf{G}_f$ と表す．

定理 1 (Grover ([4])) $\mathbf{W}^{\otimes n}|0\rangle^{\otimes n}$ に $\mathbf{G}_f$ を $\lfloor \frac{\pi}{4}\sqrt{N} \rfloor$ 回適用することにより，$\mathcal{O}\left(\sqrt{\frac{N}{t}}\right)$ 時間で $1 - \frac{1}{N}$ 以上の確率でデータベース検索問題の解を得る．

Boyer らは，複数個の解が存在する場合への拡張を考え，次の定理 2，定理 3 を示した [2].

定理 2 解が t 個 *(既知)* のとき，$\mathcal{O}\left(\sqrt{\frac{N}{t}}\right)$ 回 $\mathbf{G}_f$ を適用すると，確率 $1 - \frac{t}{N}$ 以上で解を得る．

*法政大学第二高等学校
†日本大学文理学部情報システム解析学科
‡東海大学理学部情報数理学科
§中央大学理工学部数学科

定理 3 解が t 個 *(未知)* のとき，次の **Algorithm 2** は $\mathbf{G}_f$ の適用回数の期待値が $\mathcal{O}\left(\sqrt{\frac{N}{t}}\right)$ で，解を得る．

Algorithm 2

Step 1. $m = 1, \lambda = \frac{6}{5}$．($\lambda$ は $1 \le \lambda \le \frac{4}{3}$ の任意の数)

Step 2. $0 \le j \le m-1$ からランダムに整数 j を選ぶ．

Step 3. 初期状態 $|\psi_0\rangle = \sum_i \frac{1}{\sqrt{N}} |i\rangle$ に $\mathbf{G}_f$ を j 回適用．

Step 4. **Step 3** の後，観測して状態 $|i\rangle$ を得る．

Step 5. もし $f(i) = 1$ ならば，終了．
そうでなければ，m を $\min\left\{\lambda m, \sqrt{N}\right\}$ として，**Step 2** に戻る．

3 量子振幅増幅

Brassard らは，グローバーのデータベース探索アルゴリズムを一般化を考案した [2, 3]．これは，ユニタリ変換 $\mathbf{Q} = -\mathbf{A}S_0A^{-1}S_\chi$ を繰り返すことで解となるものの振幅を増加させる．

定理 4 (Quadratic speedup without knowing a [3]) $\mathcal{A}$ を観測しない任意の量子アルゴリズムとし，$\chi : \mathbb{Z} \to \{0, 1\}$ を任意のブール関数とする．a を $\mathbf{A}$ の初期実行後に観測を行った場合に解を得る確率とする．ただし，解とは $\chi(z) = 1$ となる z のことである．

このとき，$a > 0$ ならば，$\mathcal{A}$ と $\mathcal{A}^{-1}$ の適応回数の期待値が $\mathcal{O}(\frac{1}{\sqrt{a}})$ 回で回を見つけるアルゴリズムが存在する．

Algorithm 3 (Qsearch)

Step 1. $l = 0, 1 < c < 2$.

Step 2. $l \leftarrow l + 1, M \leftarrow \lceil c^l \rceil$.

Step 3. $\mathcal{A}|0\rangle$ を行い，観測する．観測結果 $|z\rangle$ が $\chi(z) = 1$ となれば z を出力して終了．

Step 4. **Step 3.** で $\chi(z) = 0$ となったならば，再び $\mathcal{A}|0\rangle$ を行い観測はしない．

Step 5. $\{1, \ldots, M\}$ からランダムに j を選ぶ．

Step 6. $\mathcal{A}|0\rangle$ に $\mathbf{Q}^j$ を適用する．

Step 7. **Step 6.** の実行後に観測を行い $\chi(z) = 1$ となる z を得たならば z を出力して終了．そうでなければ，**Step 2.** に戻る．

4 Qsearch の適用

4.1 クリーク発見問題

l-クリーク発見問題とは，グラフ $G=(V,E)$ ($|V|=n$, $|E|=m$) が与えられたとき，l-クリーク $\{v_{i_1}, v_{i_2}, \ldots, v_{i_l}\}$ を見つける問題である．この問題を古典的アルゴリズムで単純な方法で解くと，$\mathcal{O}(n^l)$ 時間かかる．Qsearch を用いて解くアルゴリズムを以下のように構成する．

Algorithm 4 (l-clique finder)

Step 1. l 頂点の組 $V\times\cdots\times V$ に対しオラクル関数 f_1 を

$$f_1(i_1, i_2, \ldots, i_l) = \begin{cases} 1 & \text{if } (v_{i_1}, v_{i_2}) \in E \\ 0 & \text{otherwise} \end{cases}$$

とする．$\mathbf{G}_{f_1}$ を $\mathcal{O}\left(\sqrt{\frac{n^2}{m}}\right)$ 回適用する．

Step 2. $j=2,3,\ldots,j=l-1$ に対してオラクル関数 f_j

$$f_j(i_1, i_2, \ldots, i_l) = \begin{cases} 1 & \text{if } (v_k, v_{i_{j+1}}) \in E \\ & \quad \text{for any } k \in \{i_1, i_2, \ldots, i_j\} \\ 0 & \text{otherwise} \end{cases}$$

とし，$\mathbf{G}_{f_j}$ を順次適用する．

Step 3. ブール関数 χ を

$$\chi(i_1, i_2, \ldots, i_l) = \begin{cases} 1 & \text{if } (v_{i_1}, v_{i_2}, \ldots, v_{i_l}) \\ & \quad \text{がクリークである.} \\ 0 & \text{otherwise} \end{cases}$$

と定義し，**Step 1–2** を量子アルゴリズム **A** として，Qsearch を行う．

このアルゴリズムを用いると，解が 1 つ以上存在するとき，$\mathcal{O}\left(l^2 n^{\frac{l}{2}} \log n\right)$ 時間で解が求まる．よって次の定理を得る．

定理 5 l-クリーク探索問題は解が少なくとも 1 つは存在するとき，*Qsearch* を用いると $\mathcal{O}\left(l^2 n^{\frac{l}{2}} \log n\right)$ 時間で解くことができる．

4.2 サイクル探索問題

l–サイクル探索問題とは，グラフ $G=(V,E)$, ($|V|=n, |E|=m$) が与えられたとき，l–サイクル $v_{i_1}, v_{i_2}, \ldots, v_{i_l}$ を発見する問題である．この問題を古典的アルゴリズムで単純な方法で解くと，$\mathcal{O}(n^{l+2})$ 時間かかる．Qsearch を用いて解くラルゴリズムを以下のように構成する．

Algorithm 5 (l-cycle finder)

Step 1. l 頂点の組 $V\times V\times\cdots\times V$ に対してオラクル関数 f_j を

$$f_j(i_1, i_2, \ldots, i_l) = \begin{cases} 1 & \text{if } (v_{i_j}, v_{i_{j+1}}) \in E \\ 0 & \text{otherwise} \end{cases}$$

とする．ただし，$j=1,2,\ldots,l-2$ である．$\mathbf{G}_{f_j}$ を順次適用する．

Step 2. オラクル関数 f_{l-1} を

$$f_{l-1}(i_1, i_2, \ldots, i_l) = \begin{cases} 1 & \text{if } (v_{i_1}, v_{i_l}), (v_{i_{l-1}}, v_{i_l}) \in E \\ 0 & \text{otherwise} \end{cases}$$

とし，$\mathbf{G}_{f_{l-1}}$ を適用する．

Step 3. ブール関数 χ を

$$\chi(i_1, i_2, \ldots, i_l) = \begin{cases} 1 & \text{if } (v_{i_1}, v_{i_2}, \ldots, v_{i_l}) \\ & \quad \text{が } l\text{-サイクルである.} \\ 0 & \text{otherwise} \end{cases}$$

と定義し，**step 1–2** を量子アルゴリズム **A** として，Qsearch を行う．

このアルゴリズムを用いると，解が少なくとも 1 つ存在するとき，$\mathcal{O}\left(\sqrt{2^{l-1}} l^{\frac{3}{2}} n^l \log n\right)$ 時間で解が求まる．よって次の定理を得る．

定理 6 l–サイクル検索問題は，解が少なくとも 1 つ存在するとき，*Qsearch* を用いると $\mathcal{O}\left(\sqrt{2^{l-1}} l^{\frac{3}{2}} n^l \log n\right)$ 時間で解ける．

4.3 2 部グラフ判定問題

2 部グラフ判定問題とは，グラフ $G=(V,E)$ ($|V|=n$, $|E|=m$) に対して，そのグラフが 2 部グラフかどうかを判定する問題である．2 部グラフは，奇サイクルを持たないことと必要十分なので，**Algorithm 5** (l-cycle finder) を利用してこの問題を解くアルゴリズムを以下に示す．

Algorithm 6

Step 1. $i=1$ から，n が偶数ならば $i=\frac{n-1}{2}$，n が奇数ならば $i=\frac{n-2}{2}$ まで以下を繰り返す．

(a) $l=(2i+1)$ として **Algorithm 5** (l-cycle finder) を行う．

(b) 観測し，閉路が見つかれば，`NO` を出力し終了．

Step 2. Step 1. において，`NO` とならないとき `YES` を出力する．

計算時間は，n が偶数のときは $\mathcal{O}(n^{\frac{n}{2}} \log n)$ n が奇数のときは $\mathcal{O}(n^{\frac{n-1}{2}} \log n)$ となる．

参考文献

[1] M. Boyer, G. Brassard, P. Høyer and A. Tapp. Tight bounds on quantum searching, quant-ph/9605034.

[2] G. Brassard, P. Høyer, M. Mosca and A. Tapp. Quantum Amplitude Amplification and Estimation, quant-ph/0005055.

[3] H. Buhrman, C. Durr, M. Heiligman, P. Høyer, F. Magnize, M. Santha, and R. Wolf. Quantum Algorithm for Element Distinctness, quant-ph/0007016.

[4] L. K. Grover. A fast quantum mechanical algorithm for database search, in Proceedings of 28th STOC, pp.212–219, 1996.