アルゴリズム 70－5
（1999．11．8）

Groverの量子探索アルゴリズムの応用

徳永 裕己　小林 弘忠　今井 浩

東京大学大学院理学系研究科情報科学専攻
〒113-0033 東京都文京区本郷 7-3-1
{tokunaga,hirotada,imai}@is.s.u-tokyo.ac.jp

あらまし N 個の未整列なインデックスからあるインデックスを探すとき，現状のコンピュータでは平均で $N/2$ 回の探索をする．しかし，Groverの量子探索アルゴリズムでは $O(\sqrt{N})$ 回で探すことができる．このアルゴリズムは“振幅の増幅”という手法を用いており，SATなど様々な問題に応用ができる．本研究では，Groverのアルゴリズムの応用として最小値探索，数え上げを取り上げる．我々はDürrとHøyerによる最小値のアルゴリズムを応用して，その近似アルゴリズムを考案した．また，Brassardらによる数え上げアルゴリズムの最適性を示した．さらにこれらの問題に対して徳永，長井，今井の量子計算機シミュレータを用いてシミュレーション実験による平均計算時間や解の分散の検証も行った．

Applications of Grover's Quantum Search Algorithm

Yuki TOKUNAGA　Hirotada KOBAYASHI　Hiroshi IMAI

Department of Information Science, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan
{tokunaga,hirotada,imai}@is.s.u-tokyo.ac.jp

Abstract In order to search an index among unsorted N indices, a present computer searches $N/2$ times on average. However, Grover's quantum algorithm can search in only $O(\sqrt{N})$ steps. This algorithm is based on the method of "amplitude amplification"and can apply various problem such as SAT. In this paper, minimum search and counting problem are taken up as applications of Grover's algorithm. Approximate minimum search algorithm applying Dürr and P. Høyer's algorithm is described. We show Counting algorithm of Brassard et al.'s is optimal. For above problems, mean time or variance are analyzed and verified using Tokunaga Nagai and Imai's quantum computer simulation program.

1　はじめに

量子計算の有用性を示したものとしては，Shorによる自然数の因数分解の多項式時間アルゴリズム [8] が有名である．Grover [5] による量子探索アルゴリズムも汎用性があるという点で注目すべきアルゴリズムである．これは N 個の未整列なインデックスから，あるインデックスを探すアルゴリズムである．現状のコンピュータでは平均で $N/2$ 回の探索をするが，量子計算では $O(\sqrt{N})$ 回で探すことができる．量子探索アルゴリズムはブラックボックス関数を用い“振幅の増幅”という手段で探索を行う．量子アルゴリズムの例としても非常にわかりやすく，SATなど様々な問題に応用ができる．

DürrとHøyerはGroverのアルゴリズムを応用して，最小値探索のアルゴリズムを考案した [4]．現

在のコンピュータでは N 回の比較計算が必要だが量子計算によると $O(\sqrt{N})$ で最小値をみつけることができる．Brassard らは解の個数 t を数え上げる問題を全体の要素を N として $O(\sqrt{tN})$ で行うアルゴリズムを考案した [3]．これは Shor のアルゴリズムと Grover のアルゴリズムを両方用いている点で興味深い．

また，Bennett らによりブラックボックス関数を用いた振幅増幅の計算量の下限は $\Omega(\sqrt{N})$ であることが示されていることも重要である [1]．これから，Grover のアルゴリズムはこの手法による最適なアルゴリズムであることがいえる．

本研究では，Grover のアルゴリズムの応用として最小値探索，数え上げを取り上げる．我々は Dürr と Høyer による最小値探索のアルゴリズムを応用して，その近似アルゴリズムを考案した．これは N 個の相異なるランダムに並んだ要素のうち，下から k 番目までの要素を $O(\sqrt{N/k})$ で探索する．さらに我々は Brassard らによる数え上げのアルゴリズムの評価を改良し，Mosca [6] が示している計算量 $\Theta(\sqrt{(t+1)(N-t+1)}\ )$ と等しいことを示し，最適性を証明した．またこれらのアルゴリズムに対して平均時間の解析を行った．そして徳永，長井，今井 [9] による量子計算機シミュレータを用いてシミュレーション実験を行い計算量の係数，解の分散を検証した．その結果，理論的解析とよく似た振る舞いが見られた．

本論文の構成は次の通りである．まず，2 節で量子計算の一般的な原理を説明する．3 節で Grover の量子探索アルゴリズムを解説し，条件を満たす要素数が不明の場合についての解析結果を示す．4 節で最小値探索のアルゴリズムの解説とその近似アルゴリズムを提案する．5 節で数え上げアルゴリズムの解説とその最適性の証明を示す．3，4，5 節ではさらにそれぞれ平均時間や分散などの解析とシミュレーションによる検証を行う．

2 量子計算の原理

古典計算機と量子計算機の違いを理解するためには，まず，1 ビットを考えるとよい．古典ビットは “真” と “偽” の 2 状態のうちのどちらか一つをとる．古典的な “確率的” ビットは確率 α で真をとり，確率 β で偽をとり，$\alpha+\beta=1$ という性質をみたす．量子ビット (qubit) は後者に非常によく似ている．量子ビットに対しては，α,β は任意の複素数をとることができ，$\|\alpha\|^2+\|\beta\|^2=1$ という性質をみたす．量子ビットを観測すると，確率的ビットのように確率 $\|\alpha\|^2$ で真をとり，確率 $\|\beta\|^2$ で偽をとる．しかし，量子計算機をモデル化したとき使用可能な変換の集合は確率的計算機に比べて大きい．ここに量子計算機の能力の所以がある．

より一般的に n ビットを考える．古典 n ビットは $N=2^n$ 個の状態のうちのどれか一つをとりうる．量子 n ビットは N 個の基底状態をとる．基底状態を $|q_1\rangle,|q_2\rangle,\ldots,|q_N\rangle$ と記す．ψ を複素数の係数をもつこれらの線形結合とする．

$$\psi=\alpha_1|q_1\rangle+\alpha_2|q_2\rangle+\ldots+\alpha_N|q_N\rangle.$$

ψ の l_2 ノルムは，

$$||\psi||=\sqrt{|\alpha_1|^2+|\alpha_2|^2+\ldots+|\alpha_N|^2}.$$

量子計算の状態は $\|\psi\|=1$ をみたす任意の ψ をとりうる．ψ は $|q_1\rangle,|q_2\rangle,\ \ldots,\ |q_N\rangle$ の基底状態の “重ね合わせ” と呼ばれ，$\alpha_1,\ldots,\alpha_N$ は “振幅” と呼ばれる．$l_2(Q)$ を $|q_1\rangle,|q_2\rangle,\ldots,|q_N\rangle$ で張られる複素内積空間とする．

任意の複素数の振幅を用いられることは量子計算の有用性の本質となっている．例えば “負” の実数を振幅に用いることによって，“正” の実数との “打ち消しあい” が起こる．これは確率的計算機にはなかったことである．

量子計算は 2 種類の変換をする．一つはユニタリ変換である．ユニタリ変換は $l_2(Q)$ 上の線形な変換 U であり，l_2 ノルムを保存する．(これは ψ が ψ' に写されたとき $\|\psi\|=\|\psi'\|=1$ ということである．)

二つめは観測である．観測は $\psi=\alpha_1|q_1\rangle+\alpha_2|q_2\rangle+\ldots+\alpha_N|q_N\rangle$ という重ね合わせのとき，確率 $\|\alpha_i\|^2$ で $|q_i\rangle$ を与える．($\|\psi\|=1$ により異なる出力の確率の和は 1 であることが保証される．) 観測後，状態は $|q_i\rangle$ に写る．

3 Grover の量子探索アルゴリズム

3.1 解の個数が既知の場合

ブラックボックス論理関数 F が定められているとする．$A=\{a|F(a)=1\}$, $B=\{b|F(b)=0\}$, $|A|=t$, $|B|=N-t$ のように，N 個の基底状態の

うち，真となる解がt個存在し，tがいくつか知っているとする．目的は論理関数Fが真となる解$a \in A$をひとつ探索することである．Groverのアルゴリズムは真となる解の振幅を増幅し，観測確率を高くすることによって探索をする．

aの振幅がk，bの振幅がlとすると量子状態は$tk^2 + (N-t)l^2 = 1$をみたし以下のようになる．

$$|\psi(k,l)\rangle = \sum_{a \in A} k|a\rangle + \sum_{b \in B} l|b\rangle.$$

Groverの量子探索アルゴリズムは次の通りである．

アルゴリズム 1 **QSearch (Grover)**

1. $F(x) = 1$をみたす基底状態$|x\rangle$の振幅にマイナスをかける．
2. Walsh-Hadamard変換をかける．これはすべての量子ビットに対して2×2ユニタリ行列

$$W = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

をかける操作である．

3. 基底状態$|0\rangle$の振幅にマイナスをかける．
4. Walsh-Hadamard変換をかける．
5. 全基底状態の振幅にマイナスをかける．

この量子探索アルゴリズム1回により，$F(x) = 1$となる基底状態$|x\rangle$の振幅が増幅し状態は以下のように遷移する．

$$|\psi(k,l)\rangle \mapsto \left|\psi\left(\frac{N-2t}{N}k + \frac{2(N-t)}{N}l, -\frac{2t}{N}k + \frac{N-2t}{N}l\right)\right\rangle$$

Groverのアルゴリズムは等しい重みの重ねあわせ状態

$$|\psi_0\rangle = |\psi(1/\sqrt{N}, 1/\sqrt{N})\rangle = \sum_{a=0}^{N-1} \frac{1}{\sqrt{N}}|a\rangle$$

から始める．この状態は$|0\rangle$の基底状態の振幅を1，他を0にとり，すべての量子ビットにWalsh-Hadamard変換をかけることで得られる．j回の繰り返し後の状態$|\psi_j\rangle = |\psi(k_j, l_j)\rangle$において，$k_j, l_j$は$k_0 = j_0 = 1/\sqrt{N}$で，$\sin^2\theta = t/N$と$\theta$を定義すると，

$$\left.\begin{aligned} k_j &= \tfrac{1}{\sqrt{t}}\sin((2j+1)\theta) \\ l_j &= \tfrac{1}{\sqrt{N-t}}\cos((2j+1)\theta) \end{aligned}\right\} \quad (1)$$

と書き表せる．式(1)から$(2m+1)\theta = \pi/2$のとき$k_m = 1$となり，振幅が最大になる．このとき$m = (\pi - 2\theta)/4\theta$である．よって

$$m = \lfloor \pi/4\theta \rfloor \leq \frac{\pi}{4}\sqrt{\frac{N}{t}} \quad (2)$$

の整数回の繰り返しにより非常に高い確率で単一の解を観測できる．ここで$\theta \geq \sin\theta = \sqrt{t/N}$を用いた．また，約50%の成功確率でよいならば，$\frac{\pi}{8}\sqrt{\frac{N}{t}}$の繰り返しで十分である．そして，$\frac{\pi}{2}\sqrt{\frac{N}{t}}$の繰り返しを行うと成功確率がほぼ0になることにも注意する．つまり，式(1)からも見てとれるように，Groverのアルゴリズムにより振幅の大きさは周期的に変化している．この周期性は5節の数え上げに効果的に用いられている．

3.2 解の個数が不明の場合

通常，解の個数を事前に知っているということはまれである．Boyerらは解の個数tが不明でも$O(\sqrt{N/t})$で十分高い確率で一つの解を探索できることを示した[2]．また，計算時間の上限は約$\frac{9}{2}\sqrt{\frac{N}{t}}$であることも示した．よって計算時間が$\frac{9}{2}\sqrt{N}$を越えたときは$t = 0$として終了する．

以下にtが不明のときに解を見つけるBoyerらのアルゴリズムを示す．簡単のために$1 \leq t \leq 3N/4$を仮定する．

アルゴリズム 2 **t-Unknown (Boyer *et al.*)**

1. $m = 1, \lambda = 1.2$と初期設定する．(ここでλの値は$1 < \lambda \leq 4/3$の任意の数でよい．)
2. $0 \leq j \leq \lfloor m \rfloor$から自然数$j$を均一の割合でランダムに選ぶ．
3. 初期状態$|\psi_0\rangle = \sum_a \frac{1}{\sqrt{N}}|a\rangle$から始めてGroverのアルゴリズムを$j$回繰り返す．
4. レジスタを観測する．aをその出力とする．
5. もし$F(a) = 1$ならば問題は解けて，ここで終了．

6. そうでなければ，m を $\min(\lambda m, \sqrt{N})$ にセットしてステージ2に戻る．

我々はBoyerらの論文をもとにして，このアルゴリズムの平均時間の解析を行った．

補題 3.1 (Boyer *et al.*) t を不明な解の個数とし，$\sin^2\theta = t/N$ とする．m を任意の正整数とし，j を0から $m-1$ の中からランダムに選んだ整数とする．初期状態 $|\psi_0\rangle = \sum_a \frac{1}{\sqrt{N}}|a\rangle$ から始めて j 回の繰り返しのあと観測して解を得られる確率は

$$P_m = \frac{1}{2} - \frac{\sin(4m\theta)}{4m\sin(2\theta)}. \tag{3}$$

特に，$m \geq 1/\sin(2\theta)$ のとき，$P_m \geq 1/4$ となる．

Boyerらのアルゴリズムと補題3.1から成功確率が1/2以上となるのは

$$\begin{aligned}
&\sum_{s=1}^{x}(1-P_m)^{s-1}P_m \\
=\ & 1 - \prod_{s=1}^{x}(1-P_m) \\
=\ & 1 - \prod_{s=1}^{x}\left(\frac{1}{2} + \frac{\sin(4m\theta)}{4m\sin(2\theta)}\right) \\
=\ & 1 - \prod_{s=1}^{x}\left(\frac{1}{2} + \frac{\sin(4\cdot\lfloor(1.2)^{s-1}\rfloor\theta)}{4\cdot\lfloor(1.2)^{s-1}\rfloor\sin(2\theta)}\right) \\
\geq\ & \frac{1}{2}
\end{aligned} \tag{4}$$

となる最小の x 回，Boyerらのアルゴリズムを繰り返したときである．

Boyerらのアルゴリズムの s 回目のループのとき j を $0 \leq j \leq \lfloor m \rfloor$ からランダムに選んだときのGroverのアルゴリズムの繰り返し回数の期待値は $\frac{1}{2}\lfloor m \rfloor$ である．よってBoyerらのアルゴリズムを x 回繰り返したときのGroverのアルゴリズムの繰り返し回数の期待値 $iter$ は

$$iter = \frac{1}{2}\sum_{s=1}^{x} m = \frac{1}{2}\sum_{s=1}^{x}(1.2)^{s-1}$$

で計算できる．

(4)式の最小の x を数式で導くのは困難なため，数式処理ソフトMapleを用いて数値計算により解いた．それを表1に記す．計算時間のカウントは全体のループを1，Groverのアルゴリズムの繰り返しを1としている．$(x+iter)/\sqrt{N/t}$ が計算量の係数を表している．N が小さいときは若干大きいが1.4から1.5で安定した．

徳永，長井，今井は量子計算の汎用的シミュレータを作成した[9]．このシミュレータはC++言語により書かれており，Sunのワークステーション上で動作する．メモリを2GB積んでいるマシンで行うと25量子ビットの動作が可能になる．速度の参考としては以下の環境で25量子ビットの離散フーリエ変換が約10分で可能である．本研究の計算機環境はSunOS, UltraSPARK II 360MHz, メモリ2GBである．ただし今回は計算時間の関係上16量子ビットまでしか扱っていない．

このシミュレータを用いて，$iter$ 等について実際に得た値を表2に記す．ブラックボックス関数 F は以下のように定めた．k はある自然数の定数，そして $T[1..N] \to [1..N]$ は相異なるランダムな順の自然数を返すテーブルで，

$$F(x) = \begin{cases} 1 & \text{if } T[x] \leq k \\ 0 & \text{if } T[x] > k \end{cases}$$

シミュレーションは100回繰り返し平均を取っている．時間の関係で $N = 65536$ までしかできなかったが，計算量の係数は約1.5に収まっていることが確認できた．また計算時間の散らばり具合いの尺度として $N = 1024, t = 1$ について統計をとったところ，平均48.5に対して標準偏差は23.83であった．

N	t	x	$iter$	$(x+iter)/\sqrt{N/t}$
256	1	12	16	1.75
1024	1	16	43	1.84
4096	1	19	77	1.50
16384	1	23	163	1.45
65536	1	27	340	1.43
262144	1	31	709	1.44
1048576	1	35	1474	1.47

表1: 平均時間解析 (t-Unknown)

4 最小値探索

N 個の未整列の要素のテーブル $T[0..N-1]$ があるとする．簡単のため，すべて異なる要素が入っているとする．このとき $T[y]$ が最小となるような y を1/2より大きい確率で $O(\sqrt{N})$ で求める量子アルゴリズムをDürr と Høyerは考案した[4]．

以下にアルゴリズムを記す．

N	t	x	$iter$	$(x+iter)/\sqrt{N/t}$
256	1	13.4	10.9	1.52
1024	1	14.7	33.8	1.51
4096	1	18.8	81.4	1.56
16384	1	22.9	172.2	1.52
65536	1	26.8	371.5	1.55

表 2: シミュレーション結果 (t-Unknown)

アルゴリズム 3 Min (Dürr, Høyer)

1. $0 \le y \le N$ からランダムに閾値 y を選ぶ.
2. 以下を計算時間が $22.5\sqrt{N} + 1.4\lg^2 N$ となるまで繰り返す.
   (a) 状態を $\sum_j \frac{1}{\sqrt{N}}|j\rangle|y\rangle$ に初期化する.
   (b) Boyer らによる解の個数が不明のときの探索アルゴリズムを用いて, $T[j] < T[y]$ となる j を探索する.
   (c) 出力を y' とする. もし $T[y'] < T[y]$ ならば閾値を y から y' に変更する.
3. y を返す.

ここで計算時間のカウントはステージ 2(a) を $\lg(N)$, ステージ 2(b) の中の Grover のアルゴリズムの 1 回の繰り返しを 1, その他は 0 としている. lg は 2 を底とする対数である. ステージ 2(b) の中の総計算時間の期待値が高々$22.5\sqrt{N}$ でステージ 2(a) の総計算時間の期待値が高々$1.4\lg^2 N$ となっている.

我々は Dürr と Høyer のアルゴリズムを応用して, 以下の命題を導いた.

命題 4.1 下から k 番目までのいずれかの値を得ることを最小値の k 近似と定める. 最小値の k 近似は 1/2 より大きい確率で $O(\sqrt{N/k})$ で求められる.

まず, Dürr と Høyer による補題を引用する.

補題 4.2 $p(t,r)$ を上記のアルゴリズムを用いたとき t 個の要素の中から r 番目の大きさの要素を閾値としていつか選択される確率とすると, $p(t,r) = 1/r$ である.

補題 4.2 を用いて命題 4.1 を証明する.

証明 アルゴリズムは Dürr と Høyer のものとほぼ同じである. ただし, ステージ 2(b) における Grover のアルゴリズムの繰り返しの回数の上限を $\frac{9}{2}\sqrt{N/k}$ と定める. この上限をこえたら終了とする. するとステージ 2(b) の総計算時間の期待値は, $T[y]$ に k 番目以下の値が入るまでの計算時間の期待値を考えて, 高々,

$$\begin{aligned}
&\sum_{r=k+1}^{N} p(N,r)\frac{9}{2}\sqrt{\frac{N}{r-1}} \\
&= \frac{9}{2}\sqrt{N}\sum_{r=k}^{N-1}\frac{1}{r+1}\frac{1}{\sqrt{r}} \\
&\le \frac{9}{2}\sqrt{N}\left(\frac{1}{(k+1)\sqrt{k}} + \sum_{r=k+1}^{N-1} r^{-\frac{3}{2}}\right) \\
&\le \frac{9}{2}\sqrt{N}\left(\frac{1}{2\sqrt{k}} + \int_{r=k}^{N-1} r^{-\frac{3}{2}}\right) \\
&= \frac{9}{2}\sqrt{N}\left(\frac{1}{2\sqrt{k}} + \left[-2r^{-\frac{1}{2}}\right]_{r=k}^{N-1}\right) \\
&\le \frac{45}{4}\sqrt{\frac{N}{k}}.
\end{aligned}$$

ステージ 2(a) の総計算時間の期待値は高々,

$$\begin{aligned}
\sum_{r=k+1}^{N} p(N,r)\lg N &= (H_N - H_k)\lg N \\
&\le (\ln N - \ln k)\lg N \\
&\le \frac{7}{10}\lg\frac{N^2}{k}.
\end{aligned}$$

□

我々は Dürr と Høyer の最小値探索についても平均時間の解析を行った. 3.2 節から Boyer のアルゴリズムの計算量の係数は 1.5 とした. すると最小値を求めるのにかかる計算時間は厳密には,

$$1.5\sqrt{N}\sum_{r=1}^{N-1}\frac{1}{r+1}\frac{1}{\sqrt{r}} + \sum_{r=2}^{N}\frac{1}{r}\lg N$$

で求まる. この平均時間解析を表 3 に表す.

繰り返し回数について, このシミュレータを用いて実際に得た値を表 4 に記す. ランダムな N 個の相異なる要素をもつテーブルから最小値を探索した. シミュレーションは 100 回繰り返し平均の値を取っている. それぞれから, 計算量の係数は約 3.0 から 5.0 に収まっていることがわかる. 2(a)+2(b)

が量子計算による実際の計算時間を表している．現状のコンピュータでは $\Omega(N)$ の計算時間が必要だが，量子計算では $\sqrt{N}$ の数倍で最小値が発見出来ていることは注目である．

N	2(a)	2(b)	$(2(a)+2(b))/\sqrt{N}$
16	9.52	8.17	4.42
32	15.29	12.79	4.96
64	22.46	19.32	5.22
128	31.03	28.56	5.26
256	40.99	41.64	5.16
512	52.34	60.13	4.97
1024	65.09	86.28	4.73
2048	79.22	123.26	4.47
4096	94.74	175.56	4.22
8192	111.64	249.52	3.99
16384	129.93	354.12	3.78
32768	149.61	502.05	3.59
65536	170.68	711.24	3.44

表 3: 平均時間解析 (Min)

N	2(a)	2(b)	$(2(a)+2(b))/\sqrt{N}$
16	9.76	8.93	4.67
32	14.55	13.73	4.99
64	22.62	22.30	5.61
128	30.45	30.14	5.35
256	39.12	44.71	5.23
512	52.29	69.54	5.38
1024	64.80	95.59	5.01
2048	80.85	146.30	5.01
4096	99.36	192.04	4.55
8192	117.65	269.09	4.27
16384	126.98	371.57	3.89
32768	148.65	561.35	3.92
65536	167.20	777.20	3.68

表 4: シミュレーション結果 (Min)

5 数え上げ

ブラックボックス論理関数 F があり，N 個の要素の中で $F(a)=1$ となる解の個数 t を高確率で求める $O(\sqrt{tN})$ のアルゴリズムを Brassard らは考案した [3]．Grover のアルゴリズムの周期性に着目し，Shor の素因数分解のアルゴリズムの本質となっている周期発見のアルゴリズムを適用している点が面白い．

以下にアルゴリズムを記す．

アルゴリズム 4 Count(F,P) (Brassard et al.)

1. 初期状態を以下のようにとる．

$$|\psi_0\rangle = \frac{1}{\sqrt{PN}} \sum_{m=0}^{P-1} \sum_{x=0}^{N-1} |m\rangle|x\rangle$$

2. 第 2 レジスタに Grover のアルゴリズムを m 回作用させる．

$$|\psi_1\rangle = \frac{1}{\sqrt{P}} \sum_{m=0}^{P-1} \left(k_m \sum_{x\in F^{-1}(1)} |x\rangle + l_m \sum_{x\in F^{-1}(0)} |x\rangle \right)$$

3. 第 2 レジスタを観測する．ここで $F(x)=1$ となる x が観測されたとして一般性を失わない．k_m を $\sin\theta = \sqrt{t/N}$ を用いて書き換え，

$$|\psi_2\rangle = \alpha \sum_{m=0}^{P-1} \sin((2m+1)\theta)|m\rangle.$$

ここで α は θ による規格化因子である．

4. 離散フーリエ変換をかける．

$$|\psi_3\rangle = \frac{\alpha}{\sqrt{P}} \sum_{l=0}^{P-1} \sum_{m=0}^{P-1} e^{2\pi iml/P} \sin((2m+1)\theta)|l\rangle$$

5. 第 1 レジスタを観測し，その値を $\tilde{f}$ とする．

6. $\tilde{t} = N\sin^2(\tilde{f}\pi/P)$ を出力する．

離散フーリエ変換により，$f = P\theta/\pi, P(\pi-\theta)/\pi$ の付近の整数値の重みが大きくなり，$|f-\tilde{f}|<1$ となる $\tilde{f}$ を観測する確率は $8/\pi^2$ 以上であることが代数的に証明される．図 1 にシミュレーションで得た $N=64$, $P=8\sqrt{N}$, $t=10$ のときの観測確率のグラフを示す．横軸が基底状態，縦軸がその観測確率を示している．

$\sin\theta = \sqrt{t/N}$ と $f = P\theta/\pi$ から $t = N\sin^2(f\pi/P)$ が導ける．

我々はこの数え上げアルゴリズムの誤差の評価をより厳密にした．この数え上げアルゴリズムの誤差は以下のように評価される．

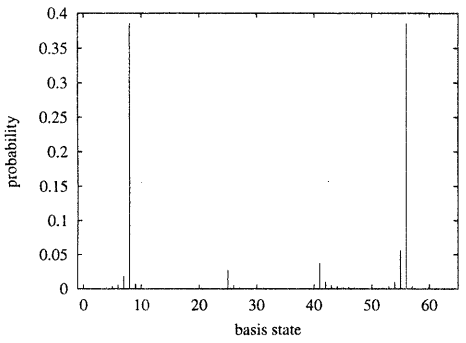

図 1: 観測確率

命題 5.1

$$|t-\tilde{t}| \leq \frac{2\sqrt{2}\pi}{P}\sqrt{t(N-t)} + \frac{\pi^2}{P^2}N. \qquad (5)$$

証明 まず Brassard らは次のような評価をした. $|f-\tilde{f}|<1$ のとき, $|\theta-\tilde{\theta}|<\pi/P$ である. よって $\sin\tilde{\theta}=\sqrt{\tilde{t}/N}$ と定めると, $|\sin\theta-\sin\tilde{\theta}|<\pi/P$ が得られる. $\sin\theta=\sqrt{t/N}$ から,

$$\begin{aligned} |t-\tilde{t}| &= |N\sin^2\theta - N\sin^2\tilde{\theta}| \\ &= N|(\sin\theta-\sin\tilde{\theta})(\sin\theta+\sin\tilde{\theta})| \\ &\leq N\left(\frac{\pi}{P}\left(\sqrt{\frac{t}{N}}+\sqrt{\frac{t}{N}}+\frac{\pi}{P}\right)\right) \\ &= \frac{2\pi}{P}\sqrt{tN}+\frac{\pi^2}{P^2}N. \qquad (6) \end{aligned}$$

以上のみだと, t が大きくなるにつれて誤差の評価が悪くなる.

我々は次の評価を加えた. $|\theta-\tilde{\theta}|<\pi/P$ から $|\cos\theta-\cos\tilde{\theta}|<\pi/P$ もいえる. $\cos\theta=\sqrt{(N-t)/N}$ から,

$$\begin{aligned} |t-\tilde{t}| &= |N\sin^2\theta - N\sin^2\tilde{\theta}| \\ &= |N\cos^2\theta - N\cos^2\tilde{\theta}| \\ &= N|(\cos\theta-\cos\tilde{\theta})(\cos\theta+\cos\tilde{\theta})| \\ &\leq N\left(\frac{\pi}{P}\left(\sqrt{\frac{N-t}{N}}+\sqrt{\frac{N-t}{N}}+\frac{\pi}{P}\right)\right) \\ &= \frac{2\pi}{P}\sqrt{(N-t)N}+\frac{\pi^2}{P^2}N. \qquad (7) \end{aligned}$$

この評価だと t が大きいときの評価が良く, 逆に t が小さいときの評価が悪い. よって (6),(7) を会わせて,

$$|t-\tilde{t}| \leq \frac{2\sqrt{2}\pi}{P}\sqrt{t(N-t)} + \frac{\pi^2}{P^2}N. \qquad \square$$

数え上げの計算時間は Mosca [6] により振幅評価を用いて $\Theta(\sqrt{(t+1)(N-t+1)})$ と示されている. 命題 5.1 の近似評価を用いると数え上げアルゴリズムはこの計算量評価と等しくなるよう改良される.[1]

系 5.2 数え上げアルゴリズム **Count**(F,P) を用いて, 2/3 以上の確率で $O(\sqrt{(t+1)(N-t+1)})$ の計算時間で厳密な解 t を得ることができる.

証明 誤差は t によっているが, t が未知であるため, まず **Count**$(F,c\sqrt{N})$ により, $|t-\tilde{t}| \leq \frac{2\sqrt{2}\pi}{c}\sqrt{t}+\frac{\pi^2}{c^2}$ で t を見積もる.

次に **Count**$(F,c\sqrt{(\tilde{t}+1)(N-\tilde{t}+1)})$ によりほぼ $|t-\tilde{t}| \leq \frac{2\sqrt{2}\pi}{c}+\frac{\pi^2}{c^2\tilde{t}}$ で t を見積もる. $c \geq 19$ と取れば, $\frac{2\sqrt{2}\pi}{c}+\frac{\pi^2}{c^2\tilde{t}} \leq 1/2$ となる. 2 回の **Count** とも成功確率は $8/\pi^2$ 以上であるので全体の成功確率は 2/3 以上である.

$\square$

最後にシミュレーションにより得た, 数え上げアルゴリズムの振る舞いをグラフで表す. 図 2, 3 は $N=64$ のとき, $t=1$ から 64 までを一回ずつシミュレートした結果である. それぞれ $P=8\sqrt{N}$, $P=32\sqrt{N}$ のときである. 横軸が実際の t, 縦軸がシミュレーションにより得られた $\tilde{t}$ を示している. $P=8\sqrt{N}$ は誤差が, 多く見られるが, $P=32\sqrt{N}$ のときはほとんど誤差はない. 大きくはずれるものは確率的に失敗した場合と考える.

また $N=64$, $P=8\sqrt{N}$ で $t=1$ から 64 までそれぞれ 100 回繰り返し, 統計を取った. このとき任意の t に対し, $|t-\tilde{t}|<5$ であるので $|t-\tilde{t}| \geq 5$ となる $\tilde{t}$ を失敗と定めた. このときの失敗率を図 4 に示す. また成功した $\tilde{t}$ の平均誤差を図 5 に示す. N が小さいため, よく振る舞いが分からないが, ある傾向はみえる. この振る舞いに関しては今後の課題とする.

謝辞

本研究のシミュレータプログラミングに関して, 長井 歩 氏から多くの協力を得たことを感謝します.

[1] [7] によると, Brassard, Høyer, Mosca, Tapp による, より定数倍良い近似のアルゴリズムがあるようだが, 未発表のため詳細はわからない.

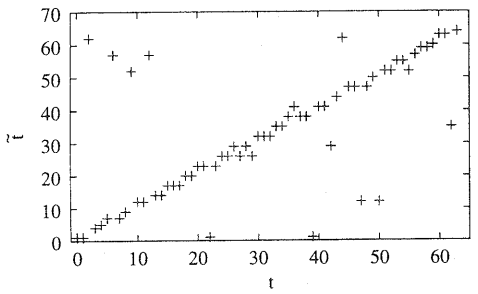

図 2: $P = 8\sqrt{N}$ のとき

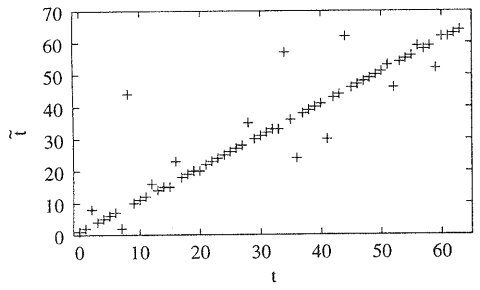

図 3: $P = 32\sqrt{N}$ のとき

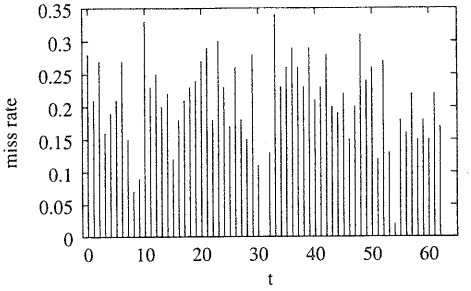

図 4: 失敗率

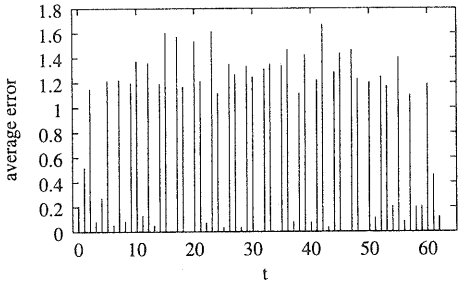

図 5: 平均誤差

参考文献

[1] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strength and weakness of quantum computing. *SIAM Journal on Computing*, Vol. 26, No. 5, pp. 1510–1523, October 1997.

[2] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Forschritte Der Physik*, Vol. 46, pp. 493–505, 1998.

[3] G. Brassard, P. Høyer, and A. Tapp. Quantum counting. In *Proceedings of the 25th International Colloquium on Automata, Languages and Programming, Lecture Notes in Computer Science*, Vol. 1443, pp. 820–831, 1998.

[4] C. Dürr and P. Høyer. A quantum algorithm for finding the minimum. Quantum Physics e-Print archive, http://xxx.lanl.gov/abs/quant-ph/960714, 1996.

[5] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of 28th ACM Symposium on Theory of Computing*, pp. 212–219, 1996.

[6] M. Mosca. Quantum searching, counting and amplitude amplification by eigenvector analysis. In *Proceedings of the Workshop on Randomized Algorithms*, 1998.

[7] A. Nayak and F. Wu. The quantum query complexity of approximating the median and related statistics. In *Proceedings of the 31th Annual ACM Symposium on Theory of Computing*, pp. 384–393, 1999.

[8] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, Vol. 26, pp. 1484–1509, 1997.

[9] 徳永裕己, 長井歩, 今井浩. 量子計算機シミュレーションシステム. 京都大学数理解析研究所講究録に掲載予定, 1999.