A-1

3関数に対する量子クロー探索アルゴリズム
Quantum Claw-finding Algorithms for Three Functions

河内 亮周* 岩間 一雄*
Akinori Kawachi Kazuo Iwama

1 はじめに

Groverの量子探索アルゴリズム [4] はShorの素因数分解アルゴリズム [6] と並んで最も賞賛されている量子アルゴリズムであり，その応用 [2, 5] や拡張 [1, 3] が活発に研究されている．その中の一つとして最近発表された要素不一致問題 (Element Distinctness) 及びクロー探索問題 (Claw-Finding) への応用 [2] はサンプリングやソーティングといった従来型のテクニックや振幅増幅 (Amplitude Amplification)[3] という量子計算のテクニックを巧みに利用しており，大変興味深い．すでにこのアルゴリズムを幾何学上の問題に応用した論文も発表されている [7].

k 個の関数 $f_1, ..., f_k$ に対して $f_1(x_1) = \cdots = f_k(x_k)$ を満たす入力の組 $(x_1, ..., x_k)$ を $f_1, ..., f_k$ のクローと呼ぶ．本稿では3関数 f, g, h のクロー探索に対して，f, g のクロー（中間解）にある制約を加えたときに高速化されることを論じる．[2] では k 個の関数の場合には関数 $f_1, ..., f_k$ の計算回数が $O(n^{1-1/2^k} \log n)$，特に $k = 3$ の場合は $O(n^{7/8} \log n)$ であるが，我々のアルゴリズムは $m \leq \sqrt{n}$ のときは $O(n^{3/4} \log n)$，$m > \sqrt{n}$ のときは $O(n^{7/12} m^{1/3} \log n)$ の計算回数により定数確率で3関数のクローを求めることができる．ここで m は中間解の個数である．つまり $m \leq n^{7/8}$ のときには我々のアルゴリズムの方が高速である．

2関数から3関数への拡張はそれ程単純ではない．例えば2関数のクローを（観測によって）求めてから，それと3番目の関数のクローを求める方法は失敗する．なぜならば2関数のクローは数多く存在する可能性があり，観測してしまうとその中の一つしか得られないためである．従ってはじめの2関数に対する中間解を上手に量子状態に乗せる事が重要になる．[2] では中間解を利用しないアルゴリズムが示されているが，我々のアイデアは次のように中間解を利用している．まず2関数の場合は f の値のランダムな集合を選び，その中から g とクローになるものをGroverの探索アルゴリズムを適用して探すが，この構造を3関数にもあてはめる．つまり我々のアルゴリズムはまず中間解の重ね合わせを複数個作っておく（これは中間解のランダムな集合に対応する.）．そしてその中で3つ目の関数とクローになる組をGroverのアルゴリズムによって探索するといったものである．2関数のクローの部分は形式的には全ての関数の値の組み合わせを含んでいるが，その内でクローとなっている組（中間解）のみ振幅が大きくなっている．従って中間解が少ないときにはその振幅が相対的に大きくなり，中間解の重ね合わせと最後の3つ目の関数とのクローを見つけるのが容易になる．

2 準備

関数 $F : \{0, ..., n-1\} \to \{0, 1\}$ が与えられたときに $F(x) = 1$ を満たす x を求めるという問題に対して解の数が m 個の場合，定数確率で解を見つけるための必要な計算回数は確率アルゴリズムでは $\Omega(n/m)$ だが，Groverの量子探索アルゴリズム [4] では $O(\sqrt{n/m})$ で可能である．このアルゴリズムが量子クロー探索アルゴリズムの基本的な構成要素となる．

また2つの関数 $f : X \to Z$, $g : Y \to Z$ ($X = Y = \{0, ..., n-1\}$, $f(x) = g(y)$ を満たす組はただ一つ) に対し $f(x) = g(y)$ を満たす組 $(x, y) \in X \times Y$ を見つける問題に対して [2] では定数確率で成功し，f, g の計算回数が $O(n^{3/4} \log n)$ の以下のようなアルゴリズムを示している．

1. X から部分集合 A をランダムに選択 ($|A| = \sqrt{n}$)
2. A を f の値に従ってソート
3. Y に対するインデックス $i \in [0, ..., n-1]$ の重ね合わせ $\frac{1}{\sqrt{n}} \sum_{i=0}^{n-1} |i\rangle$ をつくり，各 $a \in A$ に対して $f(a) = g(i)$ を満たす a を二分探索により探す．これをそのような a が見つかれば1，見つからなければ0を返すような関数を評価するとみなして，1を返すような Y の要素の振幅をGroverのアルゴリズムによって上げる．
4. 解の候補を含む A を見つけるためにステップ 1.-3. に振幅増幅を適用する．

ここで振幅増幅 [3] とは次のようなGroverのアルゴリズムを一般化したテクニックである．$\mathcal{A}$ を観測を行わない量子アルゴリズム，χ を $\chi : Z \to \{0, 1\}$，a を $\mathcal{A}$ の1回の適用で $\chi(z) = 1$ となるような z を見つける確率としたとき，$O(1/\sqrt{a})$ 回の $\mathcal{A}$ およびその逆計算 $\mathcal{A}^{-1}$ の適用で定数確率で z を見つけるアルゴリズムが構成できる．

3 従来のアルゴリズム

3つの関数 $f : X \to W, g : Y \to W, Z \to W$ が入力として与えられた場合，(ここで $X = Y = Z = \{0, ..., n-1\}$ であり，$f(x) = g(y) = h(z)$ を満たす組はただ一つ存在する.) $f(x) = g(y) = h(z)$ を満たす組 $(x, y, z) \in X \times Y \times Z$ を求めるという問題に対して [2] は以下の量子クロー探索アルゴリズムを示している．

1. X からサイズ $n^{3/4}$ の部分集合 A を選び，f の値でソートする．
2. Y からサイズ $n^{1/2}$ の部分集合 B を選び，g の値でソートする．
3. Y に対するインデックス $i \in [0, ..., n-1]$ の重ね合わせ $\frac{1}{\sqrt{n}} \sum_{i=0}^{n-1} |i\rangle$ をつくり，各 $(a, b) \in A \times B$ に対して $f(a) = g(b) = h(i)$ を満たす (a, b) を二分探索により探す．これをそのような (a, b) が見つかれば1，見つからなければ0を返すような関数を評価するとみなして，1を返すような Y の要素の振幅をGroverのアルゴリズムによって上げる．
4. 2.-3. に振幅増幅を適用する．
5. 1.-4. に振幅増幅を適用する．

なお f, g, h の計算回数は $O(n^{7/8} \log n)$ である．

4 新アルゴリズム

4.1 基本的な考え方

前節の結果を踏まえて3つの関数 f, g, h に対する我々のアルゴリズムを示す．前節で示した3関数に対するクロー探

*京都大学情報学研究科，ERATO 今井量子計算機構プロジェクト

索アルゴリズムはランダムサンプリングされた集合 A と B の直積を二分探索を用いた Grover のアルゴリズムを適用することで h における解の候補の確率をあげ，A と B に解の候補が含まれる確率を振幅増幅によって上げていた．つまりこのアルゴリズムでは f と g のクローに対する情報は全く利用していないことが分かる．しかし直観的には中間解の数が少なければ，ただ f,g の値のランダムな集合を取ってきてその中を探すよりも数少ない f,g の中間解と h とのクローを探した方がより容易に解が求められると考えられる．我々のアルゴリズムの構造は 2 関数のときにソートされた f の値の部分集合の重ね合わせに対してクローとなる g の値を Grover のアルゴリズムで求めるといったアイデアを再帰的に適用するものである．まず 2 関数のクロー探索を複数回実行し，複数個の f,g のクローを保持した重ね合わせを作る．この複数個の重ね合わせは微小なエラーを含む f,g の中間解のランダムな集合と解釈でき，2 関数の場合の f の値のランダムな集合に対応している．さらにこの f,g の中間解の部分集合の重ね合わせに対してクローになる h の値を Grover のアルゴリズムによって求めることで f,g,h のクローが求められる．

4.2 アルゴリズムの記述

3 関数のクロー探索問題に対する我々のアルゴリズムを構成するにあたり，仮定を設ける．，また $(x_1,y_1)\neq(x_2,y_2)$ に対して $f(x_1)=g(y_1)=a, f(x_2)=g(y_2)=b$ のとき $a\neq b$，すなわち各クローに対する f,g の値は全て異なる値を持つとする．ここでは $f(x)=g(y)$ を満たす中間解 (x,y) の個数 m が既知の場合を示しているが，解の数が未知の場合の Grover のアルゴリズムおよび振幅増幅を利用することで前節の考え方に基づいてほぼ同じ計算量で構成可能である．

1. Hadamard 変換により初期化を行う．

$$\frac{1}{\sqrt{n^l}}\sum_{i_1=0}^{n-1}\cdots\sum_{i_l=0}^{n-1}|i_1\rangle\cdots|i_l\rangle\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}|j\rangle$$

ここで最初の l 個のレジスタは f の定義域 X の要素を l 個格納するためで，最後のレジスタは g の定義域 Y の要素一つを格納するためのものである．ここでは明示していないが，実際にはこれらのレジスタに格納されている値に対する f,g の値を格納するレジスタ，及びオラクルビットも用意する．

2. f の値に従って $i_1,...,i_l$ をソートする．ソート後の列を $i'_1,...,i'_l$ とする．

$$\frac{1}{\sqrt{n^l}}\sum_{i_1=0}^{n-1}\cdots\sum_{i_l=0}^{n-1}|i_1\rangle\cdots|i_l\rangle|i'_1\rangle\cdots|i'_l\rangle\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}|j\rangle$$

3. $f(i'_1),...,f(i'_l)$ から二分探索で $g(j)$ と等しいものを探す．これを Grover のアルゴリズムによって解の振幅を上げる．Grover のアルゴリズムを r ステップ動作させたときの全体の重ね合わせは [1] の解析結果を用いて

$$\frac{1}{\sqrt{n^l}}\left(\sum_{p=1}^{m}\sum_{q=1}^{m}\frac{1}{\sqrt{q}}\sin(2r+1)\theta_q|I_p^q\rangle|j^{(p)}\rangle\right)+|err\rangle$$

と表現できる．ここで $f(i^{(k)})=g(j^{(k)}),(k=1,...,m)$ を中間解とする．また $|I_p^q\rangle$ は $i^{(k)},(k=1,...,m)$ のうち $i^{(p)}$ を含む q 種類を $|i_1\rangle\cdots|i_l\rangle$ 中に含む項であり，$\theta_k,(k=1,...,m)$ は $\sin^2\theta_k=k/n$ を満たす．$|err\rangle$ は中間解を含まない項の重ね合わせである．すなわち 1.-3. で小さなエラーを含む中間解の重ね合わせを得ることができる．

4. 1.-3. に対して振幅増幅を適用する．

$$\frac{\alpha}{\sqrt{n^l}}\left(\sum_{p=1}^{m}\sum_{q=1}^{m}\frac{1}{\sqrt{q}}\sin(2r+1)\theta_q|I_p^q\rangle|j^{(p)}\rangle\right)+\beta|err\rangle$$

ここで $|J_p\rangle = \frac{\alpha}{\sqrt{n^l}}\sum_{q=1}^{m}\frac{1}{\sqrt{q}}\sin(2r+1)\theta_q|I_p^q\rangle|j^{(p)}\rangle$，$|err'\rangle=\beta|err\rangle$ とおくとこの重ね合わせは $\sum_{p=1}^{m}|J_p\rangle+|err'\rangle$ で表される．

5. 1.-4. を独立に l' 回実行する．また h のインデックスの重ね合わせを Hadamard 変換により作る．

$$\left(\bigotimes_{i=1}^{l'}\left(\sum_{p_i=1}^{m}|J_{p_i}\rangle+|err'\rangle\right)\right)\otimes\frac{1}{\sqrt{n}}\sum_{k=1}^{n}|k\rangle$$

6. g の値に従って中間解＋エラーの重ね合わせをソートする．ソート後の列を $J'_{p_1},...,J'_{p_{l'}}$ とする．($J'_{p_1},...,J'_{p_{l'}}$ の中には小さな振幅を持つエラーも含まれることになる．)

$$\left(\bigotimes_{i=1}^{l'}\left(\sum_{p_i=1}^{m}|J_{p_i}\rangle+|err'\rangle\right)\right)\otimes|J'_{p_1}\rangle\cdots|J'_{p_{l'}}\rangle\otimes\frac{1}{\sqrt{n}}\sum_{k=1}^{n}|k\rangle$$

7. 3. と同様に $|J_{p_i}{}'\rangle,(i=1,...,l')$ が含む g のインデックス j_i から $g(j_i)$ を計算し，$g(j_i),(i=1,...,l')$ から二分探索で $h(k)$ と等しいものを探す．これを Grover のアルゴリズムによって解の振幅を上げる．

8. 1.-7. に対して振幅増幅を適用する．

4.3 計算量

我々のアルゴリズムは中間解の数によって計算量が異なり，$m\leq l$ のときは $O(\sqrt{m/l'}(l'n/\sqrt{ml})\log n)$ であり，$l=\sqrt{n},l'=1$ で最適となって $O(n^{3/4}\log n)$，また $m>l$ のときは $O(\sqrt{m}((l'^{1/2}n^{2/3}/m^{1/3})\log n+\sqrt{n/l'}\log n))$ であり，$l=n^{2/3}/m^{1/3},l'=m^{1/3}/n^{1/6}$ で最適となって $O(n^{7/12}m^{1/3}\log n)$ で定数確率でクローが求められる．

参考文献

[1] M. Boyer, G. Brassard, P. Hoyer and A. Tapp, "Tight Bounds on quantum searching", Proc. of PhysComp'96, 1996.

[2] H. Buhrman, C. Durr, M. Heiligman, P. Hoyer, F. Magniez, M. Santha and R. Wolf, "Quantum Algorithm for Element Distinctness", Proc. of CCC'01, 131-137, 2001.

[3] G. Brassard, P. Hoyer, M. Mosca, A. Tapp, "Quantum Amplitude Amplification and Estimation", To appear in Quantum Computation and Quantum Information: A Millennium Volume, AMS Contemporary Mathematics Series.

[4] L. Grover, "A fast quantum mechanical algorithm for database search", Proc. of STOC'96, 212-218, 1996.

[5] L. Grover, "Rapid sampling through quantum computing", Proc. of STOC'00, 618-626, 2000.

[6] P. Shor, "Algorithm for Quantum Computation: Discrete Log and Factoring", Proc. of FOCS'94, 124-134, 1994.

[7] K. Sadakane, N. Sugawara and T. Tokuyama, "Quantum algorithms for intersection and proximity problems," Proc. of ISAAC'01, LNCS 2223, 148-159, 2001.