定数幅量子ブランチングプログラムの計算能力

中西 正樹[†]，浜口 清治[‡]，柏原 敏伸[‡]
[†] 奈良先端科学技術大学院大学 情報科学研究科，[‡] 大阪大学大学院基礎工学研究科

On the Power of Bounded Width Quantum Branching Programs

Masaki Nakanishi[†], Kiyoharu Hamaguchi[‡], Toshinobu Kashiwabara[‡]
[†]Graduate School of Information Science, Nara Institute of Science and Technology,
[‡]Gracuate School of Engineering Science, Osaka University

概 要

近年，量子計算機に関する研究が盛んに行われており，従来の計算機に比べて，能力が高い可能性があることが報告されている．また，NMR を用いた量子計算機の実現が報告されるなど，注目を集めている．しかしながら，現在のところ，扱うことのできる量子ビット数は少数であり，したがって，少量子ビットを用いた量子計算機の能力の解析が重要性を増している．

本稿では，少数 (1 ~ 数個) の量子ビットを用いた計算機の能力を解析するための計算モデルとして，定数幅量子ブランチングプログラムを導入し，また，定数幅確率可逆ブランチングプログラムを対比する従来の計算モデルとして用いることにより，量子計算モデルの優位性を証明する．

1 はじめに

近年，量子計算機に関する研究が盛んに行われており，従来の計算機に比べて，能力が高い可能性があることが報告されている [1, 4]．また，NMR を用いた量子計算機の実現が報告されるなど，注目を集めている．しかしながら，現在のところ，扱うことのできる量子ビット数は少数であり，したがって，少量子ビットを用いた量子計算機の能力の解析が重要性を増している．

一方で，従来の計算モデルとしてブランチングプログラムと呼ばれる計算モデルについて研究が行われてきており [2]，筆者らはこれを量子計算可能なモデルに拡張した量子ブランチングプログラムを提案している [3]．

本稿では [3] で導入された定数幅量子ブランチングプログラムにさらに適切な制約を加えた定数幅量子ブランチングプログラムを導入する．この定数幅量子ブランチングプログラムは定数個の量子ビットを用いて実現可能な計算モデルである．この計算モデルを用いて少数 (1 ~ 数個) の量子ビットを用いた計算機の能力を解析する．定数幅確率可逆ブランチングプログラムを対比する従来の計算モデルとして用いることにより，量子計算モデルの優位性を証明する．

以下，2 節で各種ブランチングプログラムの定義を行い，3 節で定数幅量子ブランチングプログラムと定数幅確率可逆ブランチングプログラムの計算能力の比較を行う．

2 ブランチングプログラム

以下に各種ブランチングプログラムの定義を示す．

定義 1 確率ブランチングプログラム

確率ブランチングプログラム (Probabilistic Branching Program ,PBP) は 0 もしくは 1 のラベルを持つ終端節点と，内部節点を持つ根付き非巡回有向グラフである．根は複数存在して良いこととする．内部節点は論理変数でラベル付けされており，0-枝，1-枝と呼ばれる出る辺が接続している．また各辺 e には $0 \leq w(e) \leq 1$ なる重み $w(e)$ が付加されている．今，節点 v に接続している 0-枝，1-枝の集合をそれぞれ $E_0(v)$，$E_1(v)$ とする．このとき，

$$\sum_{e \in E_0(v)} w(e) = 1, \sum_{e \in E_1(v)} w(e) = 1$$

である．辺の重みは次のステップでその辺をたどるという事象に対する確率を表す．根である節点の中にソースと呼ばれる他の節点と識別可能な節点がある．各節点にはレベルと呼ばれる自然数が対応付けられており，レベル i の節点から出る辺はレベル $i+1$ 以上の節点をさす．

入力 $\boldsymbol{x}$ が与えられたとき，ソースを初期節点として，節点にラベル付けされた変数の値が 0 ならば 0-枝を，1 ならば 1-枝を重みとして与えられた確率にしたがってたどる．0 がラベル付けされた終端節点にたどり着いたなら 0 を，1 がラベル付けされた終端節点にたどり着いたなら 1 を出力する．

入力 $\boldsymbol{x}$ に対して関数 $f(\boldsymbol{x})$ の値を誤り確率 $\frac{1}{2} - \varepsilon (\frac{1}{2} \geq \varepsilon > 0)$ 以下で出力する確率ブランチングプログラムを，f を計算する確率ブランチングプログラムと呼ぶ．

□

図 1 に確率ブランチングプログラムの例を示す．

定義 2 可逆ブランチングプログラム

ブランチングプログラムの各節点について，入る 0-枝，1-枝の本数がそれぞれ高々 1 本のとき，可逆ブランチングプログラム (Reversible Branching Program，RBP) と呼ぶ．

□

定義 3 量子ブランチングプログラム

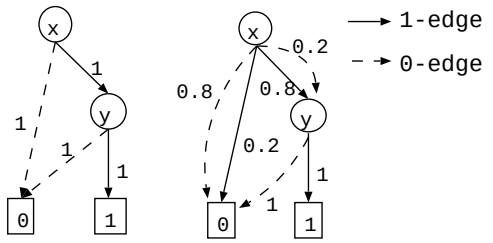

図 1: $f = xy$ をそれぞれ誤り確率 0, 0.2 で計算する確率ブランチングプログラム

量子ブランチングプログラム (Quantum Branching Program，QBP) は 0 もしくは 1 のラベルを持つ終端節点と，内部節点を持つ根付き非巡回有向グラフである．根は複数存在して良いこととする．内部節点は論理変数でラベル付けされた変数節点と，論理変数でラベル付けされていないダミー節点がある．各内部節点には 0-枝，1-枝と呼ばれる出る辺が接続している．また各辺 e には $0 \leq |w(e)| \leq 1$ なる複素数の重み $w(e)$ が付加されている．ダミー節点においては，その 0-枝，1-枝は重みも含めて同じものである．今，節点 v に接続している 0-枝，1-枝の集合をそれぞれ $E_0(v)$，$E_1(v)$ とする．このとき，

$$\sum_{e \in E_0(v)} |w(e)|^2 = 1, \sum_{e \in E_1(v)} |w(e)|^2 = 1$$

である．辺の重みは次のステップでその辺をたどるという事象に対する確率振幅を表す[1] ．根である節点の中にソースと呼ばれる他の節点と識別可能な節点がある．また，節点の集合 V は 1 でラベル付けされた終端節点からなる受理状態集合 (V_{acc})，0 でラベル付けされた終端節点からなる非受理状態集合 (V_{rej})，内部節点からなる非停止状態集合 (V_{non}) に分割される．QBP の様相は節点のみによって表されるため，以後，QBP の様相と QBP の節点を同一視する．各節点にはレベルと呼ばれる自然数が対応付けられており，レベル i の節点から出る辺はレベル $i+1$ 以上の節点をさす．

[1] 確率振幅の絶対値の 2 乗が確率となる．

QBP P の重ね合わせ状態は $l_2(V)$ の任意のノルム 1 の要素である．各様相 $v \in V$ に対して，列ベクトル $|v\rangle$ を次のように定義する．

- $|V|$ 行の列ベクトルである．
- v に対応する行が 1，他の行は 0 である．

状態遷移関数 $\delta : (V \times \{0,1\} \times V) \longrightarrow \mathbb{C}$ を以下のように定義する．

$$\delta(v, a, v') = w(e)$$

ここで $a \in \{0,1\}$ であり，$a = 1$ のとき $\delta(v, a, v')$ は v から v' への 1-枝の重みとし，$a = 0$ のとき $\delta(v, a, v')$ は v から v' への 0-枝の重みとする．該当する枝が存在しない場合は $\delta(v, a, v') = 0$ とする．

状態遷移行列 $U_\delta^{\boldsymbol{x}}$ を以下のように定義する．

$$U_\delta^{\boldsymbol{x}}(|v\rangle) = \sum_{v' \in V} \delta(v, x(v), v') \, |v'\rangle$$

ここで $\boldsymbol{x}$ は入力を表し，v が変数節点のとき，$x(v)$ は $\boldsymbol{x}$ における，節点 v にラベル付けされた変数に対する値を表す．v がダミー節点のときは，$\delta(v, 0, v') = \delta(v, 1, v')$ であるため，$x(v)$ は 0, 1 どちらの値でも良い．ここでは 1 と定義する．

$U_\delta^{\boldsymbol{x}}$ がユニタリ行列の時，対応する QBP は well-formed であるという．つまりこのとき，QBP は量子理論に従うと言える．well-formed な QBP にするために，終端節点からの遷移を適切に定義する必要がある．以降では図中に終端節点からの遷移も 0-枝，1-枝として表記する．また，以降では，well-formed である量子ブランチングプログラムのみを考える．

オブザーバブル $\mathcal{O} = E_{non} \oplus E_{acc} \oplus E_{rej}$ を以下のように定義する．

$$\begin{aligned} E_{non} &= span\{|v\rangle \, | v \in V_{non}\} \\ E_{acc} &= span\{|v\rangle \, | v \in V_{acc}\} \\ E_{rej} &= span\{|v\rangle \, | v \in V_{rej}\} \end{aligned}$$

また，オブザーバブル $\mathcal{O}$ について観測を行なった際の出力を E_{non}，E_{acc}，E_{rej} それぞれに対して “non”，“acc”，“rej” とする．

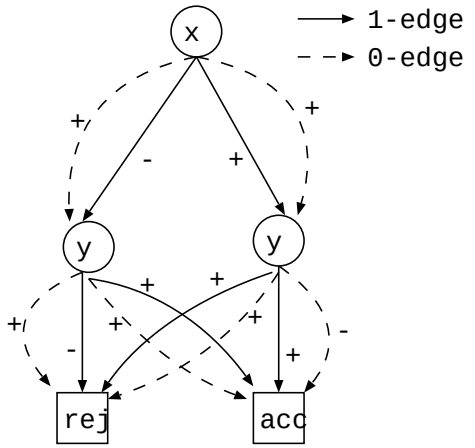

図 2: $f = xy\oplus$ をそれぞれ誤り確率 0 で計算する量子ブランチングプログラム．各辺の重みは $1/\sqrt{2}$ または $-1/\sqrt{2}$ であり，図中では符号のみを記してある．終端節点からの遷移は省略してある．

量子ブランチングプログラムが計算する関数を以下のように定義する．

ソースである節点を v_s ．初期状態を $|\psi_0\rangle = |v_s\rangle$ とし，以下の操作を行なう．

1. $|\psi_{i+1}\rangle = U_\delta^{\boldsymbol{x}} \, |\psi_i\rangle$ とする．
2. $|\psi_{i+1}\rangle$ をオブザーバブル $\mathcal{O}$ で観測する．観測によって $|\psi_{i+1}\rangle$ は観測された部分空間への射影へと収縮する．“acc” が観測されれば 1 を出力する．“rej” が観測されれば 0 を出力する．“non” が観測されれば (1) を繰り返す．

上記 (1)，(2) の操作を合わせて 1 ステップ分の操作と呼ぶことにする．

入力 $\boldsymbol{x}$ に対して関数 $f(\boldsymbol{x})$ の値を誤り確率 $\frac{1}{2} - \varepsilon(\frac{1}{2} \geq \varepsilon > 0)$ 以下で出力する量子ブランチングプログラムを，f を計算する量子ブランチングプログラムと呼ぶ．

□

図 2 に量子ブランチングプログラムの例を示す．

ブランチングプログラム中の任意の節点について，レベル i の節点から出る辺がレベル $i+1$ の節点を指すとき，そのブランチングプログラムを標準型ブランチングプログラムと呼ぶ．後に述べる条件 1 を満たす量子ブランチングプログラムは標準型ブランチングプログラムである．また，任意の確率ブランチングプログラムは標準型ブランチングプログラムに変換できる．以下では標準型ブランチングプログラムのみを考える．

定義 4 定数幅ブランチングプログラム

入力変数の個数が n 個の関数 f_n に対してその関数を計算する標準型ブランチングプログラム P_n を考える．f_n，P_n それぞれに対して，関数の属 $\{f_n\}$ とブランチングプログラムの属 $\{P_n\}$ 考える．ある定数 k が存在して，任意の $P \in \{P_n\}$ なるブランチングプログラム P に対して，各レベルについて，そのレベルに属する節点の個数が k 以下であるとき，そのブランチングプログラムの属を定数幅ブランチングプログラム，または，幅-k ブランチングプログラムと呼ぶ．以降，混乱が生じない場合，定数幅ブランチングプログラムに属する個々のブランチングプログラムも定数幅ブランチングプログラムと呼ぶ．

□

定義 5 順序付ブランチングプログラム

ブランチングプログラム P と変数順序 $\pi = (x_{k_1} < x_{k_2} < \ldots < x_{k_n})$ に対して，根から終端節点までの任意の経路について変数の出現順序が π に従う時，つまり，$i < j$ なら，x_{k_i} が x_{k_j} より先に出現する時，P は順序付ブランチングプログラムであるという．

□

量子ブランチングプログラムは次の条件を満たすとき定数個の量子ビットを用いて実現可能である．

条件 1

- 定数幅ブランチングプログラムである．
- 同じレベルの変数節点には同じ変数がラベル付けされている．
- レベル i の節点から出る辺はレベル $i+1$ の節点を指す．

□

幅-k 量子ブランチングプログラムが上記条件を満たすとき，各ステップにおいて，ブランチングプログラムの状態は同じレベルの節点の重ね合わせとなっている．そのため，各ステップにおいて高々 k 個の様相が重ね合わさっていることになり，$\lceil log_2 k \rceil$ 個の量子ビットを用いて，この重ね合わせを表現することが可能である．この k 個の様相の集合を $Q = \{q_1, \ldots, q_k\}$ とする．

幅-k 量子ブランチングプログラムの節点の集合を V，受理状態集合，非受理状態集合，非停止状態集合をそれぞれ V_{acc}，V_{rej}，V_{non} とする．さらに，レベル i の節点を $V_i = \{v_{i,1}, v_{i,2}, \ldots, v_{i,k}\} \subseteq V$ とする．また，$V_{i,acc} = \{v_{i,j} | 1 \leq j \leq k, v_{i,j} \in V_{acc}\}$，$V_{i,rej} = \{v_{i,j} | 1 \leq j \leq k, v_{i,j} \in V_{rej}\}$，$V_{i,non} = \{v_{i,j} | 1 \leq j \leq k, v_{i,j} \in V_{non}\}$ とする．$\delta_{i,a}(q_s, q_t) = \delta(v_{i,s}, a, v_{i+1,t})$ とし，$\delta_{i,a}$ を元に Q に対する状態遷移行列 $U_{i,a}$ を構成する．ここでオブザーバブル $\mathcal{O}_i = E_{i,non} \oplus E_{i,acc} \oplus E_{i,rej}$ を以下のように定義する．

$$
\begin{aligned}
E_{i,non} &= span\{|q_j\rangle \,|v_{i,j} \in V_{i,non}\} \\
E_{i,acc} &= span\{|q_j\rangle \,|v_{i,j} \in V_{i,acc}\} \\
E_{i,rej} &= span\{|q_j\rangle \,|v_{i,j} \in V_{i,rej}\}
\end{aligned}
$$

元の幅-k 量子ブランチングプログラムのソースを $v_{1,j}$ とする．このとき，$|\psi_1\rangle = |q_j\rangle$ を初期状態とし，各ステップ i において，以下の操作を行うことにより，$\lceil log_2 k \rceil$ 個の量子ビットを用いて元の定数幅量子ブランチングプログラムの動作を模倣できる．

- 元の幅-k 量子ブランチングプログラムは，各レベルにおいて，読む必要のある変数は高々一つである．つまり，各ステップにおいて，重ね合わせ中に存在する全ての

様相で，同じ変数を読むことになる．そこで，レベル i で読む変数に対する入力 a に対して，$|\psi_{i+1}\rangle = U_{i,a}\,|\psi_i\rangle$ とする．

- $|\psi_{i+1}\rangle$ をオブザーバブル $\mathcal{O}_i$ で観測する．観測によって $|\psi_{i+1}\rangle$ は観測された部分空間への射影へと収縮する．“acc” が観測されれば 1 を出力する．“rej” が観測されれば 0 を出力する．“non” が観測されれば上記操作を繰り返す．

元のブランチングプログラムにおいて，i ステップ後の重ね合わせ状態が

$$|\psi_i'\rangle = \alpha_1\,|v_{i,1}\rangle + \cdots + \alpha_k\,|v_{i,k}\rangle$$

のとき，対応する $\lceil log_2 k \rceil$ 個の量子ビットの重ね合わせは

$$|\psi_i\rangle = \alpha_1\,|q_1\rangle + \cdots + \alpha_k\,|q_k\rangle$$

であることに注意する．

□

3 順序付き定数幅量子ブランチングプログラムの計算能力

以下に，定数幅量子ブランチングプログラムとその対比する従来の計算モデルである定数幅確率可逆ブランチングプログラムについて，いくつかの定理を示す．

定理 1 順序付き幅-3 量子ブランチングプログラムは

$$(x_1 + x_2 + \ldots + x_m)(y_1 + y_2 + \ldots + y_n)$$

を計算可能である．

(証明)

$1/2 > \varepsilon > 1/3$ とすることにより，図 3 に示す順序付き幅-3 量子ブランチングプログラムで計算できる．以下にこのブランチングプログラムを説明する．

図 3 の (1) において，$x_1 = 1$ のとき，

$$\frac{1}{\sqrt{2}}\,|v_1\rangle + \frac{1}{\sqrt{2}}\,|v_2\rangle \xrightarrow{U_\delta^{\boldsymbol{x}}} \frac{1}{\sqrt{2}}\,|u_1\rangle + \frac{1}{\sqrt{2}}\,|u_2\rangle$$

であり，(2) において，

$$\frac{1}{\sqrt{2}}\,|v_3\rangle + \frac{1}{\sqrt{2}}\,|v_4\rangle \xrightarrow{U_\delta^{\boldsymbol{x}}} |u_3\rangle$$

である．

Case 1 $(x_1+\ldots+x_m) = 0, (y_1+\ldots+y_n) = 0$ のとき

r_1 にて “rej” を得る確率が ε，r_2 にて “rej” を得る確率が $\frac{1}{2}(1-\varepsilon)$，$r_3$ にて “rej” を得る確率が $\frac{1}{2}\cdot\frac{1}{2}(1-\varepsilon)$．よって “rej” を得る確率は $\frac{3+\varepsilon}{4} > \frac{1}{2}$

Case 2 $(x_1+\ldots+x_m) = 0, (y_1+\ldots+y_n) = 1$ のとき

r_1 にて “rej” を得る確率が ε，r_2 にて “rej” を得る確率が $\frac{1}{2}(1-\varepsilon)$，$r_3$ にて “rej” を得る確率が 0．よって “rej” を得る確率は $\frac{1}{2} + \varepsilon > \frac{1}{2}$

Case 3 $(x_1+\ldots+x_m) = 1, (y_1+\ldots+y_n) = 0$ のとき

r_1 にて “rej” を得る確率が ε，r_2 にて “rej” を得る確率が 0，r_3 にて “rej” を得る確率が $\frac{1}{2}\cdot\frac{1}{2}(1-\varepsilon)$．よって “rej” を得る確率は $\frac{1}{4} + \frac{3}{4}\varepsilon$ である．$\varepsilon > \frac{1}{3}$ より，この確率は $\frac{1}{2}$ より大きい．

Case 4 $(x_1+\ldots+x_m) = 1, (y_1+\ldots+y_n) = 1$ のとき

r_1 にて “rej” を得る確率が ε，r_2 にて “rej” を得る確率が 0，r_3 にて “rej” を得る確率が 0．よって “rej” を得る確率は $\varepsilon < \frac{1}{2}$ である．

□

上記ブランチングプログラムは条件 1 を満たす．つまり，定数個の量子ビットを用いて実現可能である．

定理 2 順序付き幅-2 量子ブランチングプログラムは $\bmod k$ を計算可能である．

つまり，入力変数 $\{x_1, \ldots, x_n\}$ (x_i：論理変数) に対して，値 1 が割り当てられている変数

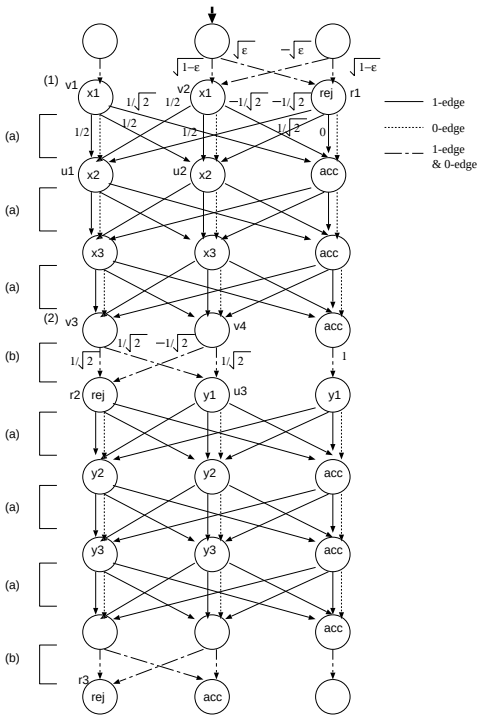

図 3: $(x_1 + x_2 + x_3 + x_4)(y_1 + y_2 + y_3 + y_4)$ を計算する量子ブランチングプログラム．(a)，(b) はそれぞれ同じ確率振幅が重みとして付加されている．

の個数が $k \cdot m$ (m:整数) のときかつそのときのみ 1 を返す関数を誤り確率 $\frac{1}{2} - \varepsilon$ $(\frac{1}{2} \geq \varepsilon > 0)$ 以下で計算する順序付き幅-2 量子ブランチングプログラムが存在する．

(証明)

図 4 にこの関数を計算する順序付き幅-2 量子ブランチングプログラムを示す．

$\sqrt{\frac{1}{2} + \varepsilon} \cdot \cos\frac{\pi}{k} < \sqrt{\frac{1}{2}}$ となるように $\varepsilon(\frac{1}{2} \geq \varepsilon > 0)$ を定める．ブランチングプログラムの 2 番目のレベル以降は，変数に値 1 が割り当てられているとき，2 次元の状態空間上で π/k の回転を行っている．したがって，$\{x_1, \ldots, x_n\}$ のうち，値 1 が割り当てられている変数の個数が $k \cdot m$ (m:整数) のとき，“acc” を得る確率が $\frac{1}{2} + \varepsilon$ となり，それ以外のときは “acc” を得る確率が $(\frac{1}{2} + \varepsilon)\cos^2\frac{\pi}{k} (< \frac{1}{2})$ より小さくなる．

□

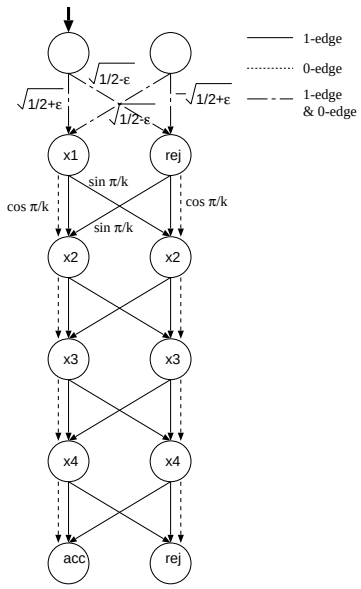

図 4: $\bmod k$ を計算する量子ブランチングプログラム．

上記ブランチングプログラムは条件 1 を満たす．つまり，定数個の量子ビットを用いて実現可能である．

定理 3 順序付き定数幅確率可逆ブランチングプログラムは

$$(x_1 + x_2 + \ldots + x_m)(y_1 + y_2 + \ldots + y_n)$$

を計算不可能である．

□

(証明)

$(x_1+x_2+\ldots+x_m)(y_1+y_2+\ldots+y_n)$ を誤り確率 $\frac{1}{2}-\varepsilon(\frac{1}{2} \geq \varepsilon > 0)$ 以下で計算する順序付き定数幅確率可逆ブランチングプログラムが存在すると仮定する．いま，$m = n$ とし，入力変数を $X = \{x_1, \ldots, x_n\}$，$Y = \{y_1, \ldots, y_n\}$ とする．L を変数順序の最初の n 個の変数からなる集合，R を変数順序の残りの n 個の変数からなる集合とし，$X_L = X \cap L$，$X_R = X \cap R$，$Y_L = Y \cap L$，$Y_R = Y \cap R$ とする．ここで，$|X_L| \geq |Y_L|$ として一般性を失わない．今，Y_L，X_R に属する変数すべてに 0 を割り当て，これ

らの変数がラベル付けされた節点を縮約したブランチングプログラムを考える．このブランチングプログラムは，$(x_1+x_2+\ldots+x_{|X_L|})(y_1+y_2+\ldots+y_{|Y_R|})$ を計算するブランチングプログラムである．いま，$|X_L|=|Y_R|$ である．ここで，$|X_L|=|Y_R|=n$，$X=X_L$，$Y=Y_R$ と置き直す．すると，このブランチングプログラムは，$(x_1+x_2+\ldots+x_n)(y_1+y_2+\ldots+y_n)$ を計算することになる．ここで，変数順序は $(x_{k_1}<\ldots<x_{k_n}<y_{l_1}<\ldots<y_{l_n})$ である．今，一般性を失うことなく変数順序を $(x_1<\ldots<x_n<y_1<\ldots<y_n)$ とし，また，レベル i の節点に変数順序が i 番目の変数がラベル付けされているとする．

可逆性を満たすため，このブランチングプログラム中に，0-枝もしくは 1-枝を複数持つ節点は高々定数個しか存在しない．いま，レベル i から j の間 $(i<j<n)$ の節点に 0-枝もしくは 1-枝を複数持つ節点が存在しないものとする．n の値を十分大きく取ることにより，$j-i$ の値を十分大きく取れる．ブランチングプログラムにおいて，入力 $\boldsymbol{x}$ にしたがって，ソースからレベル i の各節点にたどり着く確率に対する確率分布を $D_i^{\boldsymbol{x}}$ と表す．またレベル i の個々の節点 v にたどり着く確率を $D_i^{\boldsymbol{x}}(v)$ と表す．次のような相異なる 2 つの入力 $\boldsymbol{x}_1$，$\boldsymbol{x}_2$ を考える．

- $\boldsymbol{x}_1 : y_1=1$，その他は 0
- $\boldsymbol{x}_2 : x_{a_1}=x_{a_2}=\cdots=x_{a_s}=y_1=1$，$i\le a_t<j$，その他は 0

このとき，$D_i^{\boldsymbol{x}_1}=D_i^{\boldsymbol{x}_2}$ である．また，各レベル間の 0-枝，1-枝の接続の仕方は高々定数通りしかないため，$j-i$ の値が十分大きいとき，$D_j^{\boldsymbol{x}_1}=D_j^{\boldsymbol{x}_2}$ となるように $a_1,\ldots,a_s$ を定めることができる．しかしながら，$\boldsymbol{x}_1$ に対しては 0 が正答であり，$\boldsymbol{x}_2$ に対しては 1 が正答である．これは矛盾である．

□

定理 4 順序付き幅-2 確率ブランチングプログラムは $f=\bmod k$ を計算不可能である．つまり，入力変数 $\{x_1,\ldots,x_n\}$ (x_i：論理変数) に対して，値 1 が割り当てられている変数の個数が $k\cdot m$ (m:整数) のときかつそのときのみ 1 を返す関数を誤り確率 $\frac{1}{2}-\varepsilon$ $(\frac{1}{2}\ge\varepsilon>0)$ 以下で計算することが不可能である．

(証明)

$f=\bmod k$ を計算する順序付き幅-2 確率ブランチングプログラムが存在すると仮定する．このブランチングプログラムは，レベル i の節点に，変数順序 i 番目の節点がラベル付けされていると仮定して一般性を失わない．

入力変数を $X=\{x_1,\ldots,x_n\}$ とし，変数順序を $\pi=(x_1<\ldots<x_n)$ とする．k に対して入力変数の個数 n が十分に大きいと仮定する．レベル $k/2$ の各節点を v_1，v_2 とする．節点 v から入力 $\boldsymbol{x}$ にしたがって枝を辿ったとき，節点 u にたどり着く確率を $P(v,\boldsymbol{x},u)$ と表す．

変数順序の最初の $k/2$ 個の入力変数 $\{x_1,\ldots,x_{k/2}\}$ に対する入力 $\boldsymbol{x}_1$，$\boldsymbol{x}_2$，$\boldsymbol{x}_3$ を考え，これらは値 1 が割り当てられた変数の個数が互いに異なるものとする．$P(source,\boldsymbol{x}_1,v_1)>P(source,\boldsymbol{x}_2,v_1)>P(source,\boldsymbol{x}_3,v_1)$，$P(source,\boldsymbol{x}_1,v_2)<P(source,\boldsymbol{x}_2,v_2)<P(source,\boldsymbol{x}_3,v_2)$ として一般性を失わない．

$\boldsymbol{x}_2$ に対して，$f(\boldsymbol{x}_2\cdot\boldsymbol{y}_2)=1$ となるような変数順序の残りの変数に対する入力 $\boldsymbol{y}_2$ を考える．$P(v_1,\boldsymbol{y}_2,acc)\ge 1/2$ とする．このとき，$P(source,\boldsymbol{x}_1\cdot\boldsymbol{y}_2,acc)>1/2$ となり，$f(\boldsymbol{x}_1\cdot\boldsymbol{y}_2)=0$ に矛盾する．したがって $P(v_1,\boldsymbol{y}_2,acc)<1/2$ である．つまり，$P(v_2,\boldsymbol{y}_2,acc)>1/2$ である．ここで，

$$P(v_1,\boldsymbol{y}_2,acc)<1/2, P(v_2,\boldsymbol{y}_2,acc)>1/2$$

$$P(source,\boldsymbol{x}_3,v_1)<P(source,\boldsymbol{x}_2,v_1)$$

$$P(source,\boldsymbol{x}_3,v_2)>P(source,\boldsymbol{x}_2,v_2)$$

さらに，

$$P(source,\boldsymbol{x}_2\cdot\boldsymbol{y}_2,acc)>1/2$$

より，

$$P(source,\boldsymbol{x}_3\cdot\boldsymbol{y}_2,acc)>1/2$$

となる．これは $f(\boldsymbol{x}_3\cdot\boldsymbol{y}_2)=0$ に矛盾する．

□

定理 5 任意の和積形論理式を論理式と同じ長さの (複数回変数を読むことを許した) 幅-3 量子ブランチングプログラムで計算できる．

ただし，和項の個数が増大すると，誤り確率が増大する．しかし，和項の個数が一定であれば，誤り確率は論理式の長さに依存しない．

(証明)

図 3 の量子ブランチングプログラムを拡張することによって可能である．以下に拡張の方法を示す．

変数の出現順序は，論理式中の変数の出現順と同じとする．和項の中の各変数に対して，図 3 の (a) と同様の遷移を定義する．各和項の終わりに図 3 の (b) と同様の遷移を定義する．ブランチングプログラムの終端に図 3 と同様の終端節点を置く．図 5 に例を示す．

和項の個数を k とし，$(1-\varepsilon)(\frac{1}{2})^k+\varepsilon > \frac{1}{2}$ となるように ε を定めることにより，正答確率 $(1-\varepsilon)(\frac{1}{2})^k+\varepsilon$ のブランチングプログラムを構成できる．正答確率について以下に説明する．

与えられた論理式を $f = C_1 \cdot C_2 \cdot \cdots \cdot C_k$ (C_i は和項)，入力を $\boldsymbol{x}$ とする． $f(\boldsymbol{x}) = 1$ のとき，このブランチングプログラムは誤り確率 ε で “acc” を返す．一方，$f(\boldsymbol{x}) = 0$ のときは，$C_1 = C_2 = \cdots = C_{k-1} = 1$ かつ $C_k = 0$ のときに誤り確率が最大となる．このとき正答確率は $(1-\varepsilon)(\frac{1}{2})^k+\varepsilon$ である．

□

上記ブランチングプログラムは条件 1 を満たす．つまり，定数個の量子ビットを用いて実現可能である．

4 むすび

定数個の量子ビットで実現可能なモデルとして定数幅量子ブランチングプログラムを提案し，従来の計算モデルである定数幅確率可逆ブランチングプログラムと能力の比較を行い，量子計算モデルの優位性を示した．

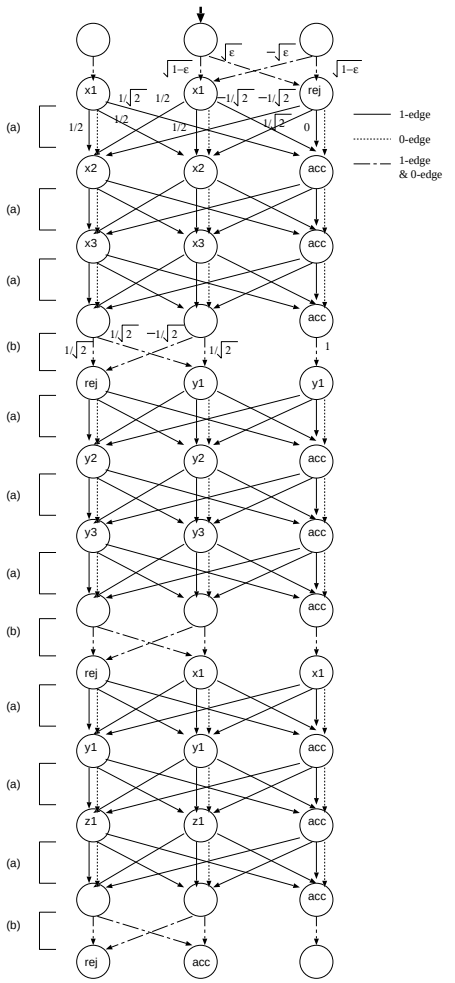

図 5: $(x_1+x_2+x_3+x_4)(y_1+y_2+y_3+y_4)(x_1+y_1+z_1)$ を計算する量子ブランチングプログラム．(a)，(b) はそれぞれ同じ確率振幅が重みとして付加されている．

多数の量子ビットを扱うことができる量子計算機が実現されるまで，こういった少量子ビット計算機の能力の解析が重要であり，今後，これらの計算機向けのさらなる有用なアルゴリズムの開発が必要である．

参考文献

[1] L.Grover, “A fast quantum mechanical algorithm for database search,” Proc. 28th Symp. on the Theory of Computing,

pp.212–219, 1996.

[2] Christoph Meinel, "Modified branching programs and their computational power," LNCS 370, Springer-Verlag, Verlin, 1989.

[3] M.Nakanishi, K.Hamaguchi, and T.Kashiwabara, "Ordered quantum branching programs are more powerful than ordered probabilistic branching programs under a bounded-width restriction," Proc. COCOON 2000, LNCS 1858, pp. 467–476, 2000.

[4] P.W.Shor, "Algorithms for quantum computation: discrete logarithms and factoring," Proc. 35th Symp. on Foundations of Computer Science, pp.124–134, 1994.