VII.
ALGORYTMY KWANTOWE

Janusz Adamowski

1 Obliczanie wartości funkcji za pomocą komputera kwantowego

Rozważmy funkcję $y = f(x)$, która elementom x zbioru liczb całkowitych

$$x \in \{0, 1, \ldots, 2^m - 1\}$$

przyporządkowuje elementy y innego zbioru liczb całkowitych

$$y \in \{0, 1, \ldots, 2^n - 1\} \ ,$$

gdzie m i n są liczbami całkowitymi dodatnimi.

Klasyczny komputer oblicza wartości $y = f(x)$ przyporządkowując każdemu wskaźnikowi $x \in \{0, 1, \ldots, 2^m - 1\}$ na wejściu odpowiednio indeksowaną wartość funkcji na wyjściu, czyli

$$y \in \{f(0), f(1), \ldots, f(2^m - 1)\} \ .$$

Komputer kwantowy używa operacji unitarnej U_f do obliczenia wartości funkcji $y = f(x)$. Każda możliwa wartość wejściowa x jest reprezentowana za pomocą wektora stanu $|x\rangle$ pierwszego rejestru (rejestru wejściowego). Podobnie każda możliwa wartość wyjściowa $y = f(x)$ jest reprezentowana za pomocą wektora stanu $|y\rangle$ drugiego rejestru (rejestru wyjściowego).

Wektory stanu odpowiadające różnym wartościom wejściowym i różnym wartościom wyjściowym są ortonormalne, czyli

$$\langle x|x'\rangle = \delta_{xx'} \ , \quad \langle y|y'\rangle = \delta_{yy'} \ .$$

Operacja obliczania wartości funkcji $y = f(x)$ przez komputer kwantowy zdefiniowana jest za pomocą operatora unitarnego U_f, który działa na dwa rejestry

$$U_f|x\rangle|0\rangle = |x\rangle|f(x)\rangle \ . \tag{1}$$

Pierwszy rejestr ($|x\rangle$) przechowuje wartości wejściowe, natomiast stan rejestru drugiego ulega przekształceniu w stan wyjściowy ($|0\rangle \longrightarrow |y = f(x)\rangle$).

Przy obliczaniu wartości funkcji $y = f(x)$ w ciekawy sposób ujawnia się **kwantowa natura obliczeń**.

Możemy tak spreparować stan rejestru wejściowego, aby był on superpozycją wszystkich stanów jednokubitowych występujących z jednakowymi amplitudami, czyli

$$|\psi_{input}\rangle = \frac{1}{2^{m/2}} \sum_{x=0}^{2^m-1} |x\rangle \ . \tag{2}$$

Za pomocą operatora unitarnego U_f wykonujemy obliczenie wartości funkcji $y = f(x)$ tylko raz i otrzymujemy wszystkie 2^m wartości funkcji $f(0), f(1), \ldots, f(2^m - 1)$, czyli

$$|\psi_{output}\rangle = U_f \left(\frac{1}{2^{m/2}} \sum_{x=0}^{2^m-1} |x\rangle \right) |0\rangle = \frac{1}{2^{m/2}} \sum_{x=0}^{2^m-1} |x\rangle|f(x)\rangle \ . \tag{3}$$

Przedyskutujmy teraz zawartość informacyjną stanu wyjściowego (3).

Stan ten zawiera superpozycję wszystkich 2^m stanów $|f(0)\rangle, |f(1)\rangle, \ldots |f(2^m-1)\rangle$, jednak w żadnym pomiarze nie otrzymamy informacji o wszystkich tych stanach.

Pojedynczy pomiar wykonany w stanie (3) pozwala na uzyskanie następujących informacji:

(1) Każdy ze stanów wyjściowych $|x\rangle|f(x)\rangle$ może być otrzymany z jednakowym prawdopodobieństwem równym $1/2^m$, a zatem każda z wartości funkcji $f(0), f(1), \ldots, f(2^m-1)$ może wystąpić z tym samym prawdopodobieństwem $1/2^m$.

(2) Jeżeli w wyniku pomiaru otrzymaliśmy np. stan $|\tilde{\psi}_{output}\rangle = |\tilde{x}\rangle|f(\tilde{x})\rangle$, to wynikiem kolejnego pomiaru, wykonanego bezpośrednio po pierwszym pomiarze, będzie ten sam stan otrzymany z prawdopodobieństwem równym 1, co oznacza, że otrzymamy znowu wartość funkcji $f(\tilde{x})$, czyli nie otrzymamy żadnej dodatkowej informacji o nowej wartości funkcji $y = f(x)$.

2 Algorytm Deutscha

Algorytm Deutscha jest przykładem realizacji **wyroczni kwantowej**. Wyroczni jest urządzeniem, które odpowiada na stawiane pytania jedynie **"tak"** lub **"nie"**. Pytania mogą być bardzo skomplikowane, procedura przygotowywania odpowiedzi może być bardzo złożona i może wymagać wielu obliczeń. Jednak wyrocznia podaje wyłącznie jedną z odpowiedzi "tak" lub "nie".

Pomimo prostoty tych odpowiedzi, wyrocznia może być użyteczna przy rozwiązywaniu bardzo złożonego problemu, jeżeli tylko potrafimy dokonać dekompozycji tego problemu na problemy proste, tak sformułowane, że wystarczy dla nich jedna z odpowiedzi "tak" lub "nie".

Algorytm Deutscha jest wyrocznią kwantową, która daje odpowiedź na następujące pytanie.

Powiedzmy, że mamy określoną funkcję f, która odwzorowuje zbiór liczb $\{0,1\}$ na zbiór $\{0,1\}$, przy czym funkcja ta jest albo **stała**, jeżeli $f(0) = f(1)$, albo **zrównoważona**, jeżeli $f(0) \neq f(1)$.

Wyrocznia Deutscha znajduje odpowiedź na pytanie, czy funkcja $f(x)$ jest stała czy zrównoważona.

Komputer klasyczny daje odpowiedź na to pytanie po obliczeniu wartości funkcji $f(0)$ i $f(1)$, czyli po dwukrotnym obliczeniu funkcji.

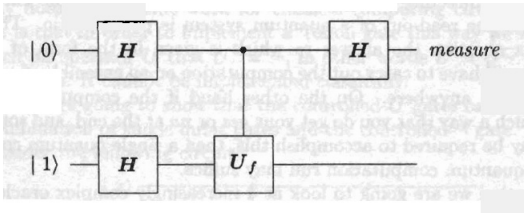
Kwantowy algorytm Deutscha odpowiada na to pytanie po jednokrotnym obliczeniu funkcji $f(x)$.

Implementacja algorytmu Deutscha odbywa się za pomocą obwodu kwantowego o schemacie pokazanym na rysunku 1.

Stanami wejściowymi są:

$|0\rangle$ w pierwszym rejestrze (górna linia),

$|1\rangle$ w dolnym rejestrze (dolna linia).



Rysunek 1: Schemat obwodu kwantowego do implementacji algorytmu Deutscha.

H jest bramką Hadamarda, a U_f jest sterowana bramką, która służy do obliczania wartości funkcji f. W jawnej postaci

$$U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle \ , \tag{4}$$

gdzie $\oplus$ jest operacją dodawania modulo 2.

Przypominam działanie bramki Hadamarda na stany bazy obliczeniowej, czyli stany wejściowe w algorytmie Deutscha.

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \ , \tag{5}$$

$$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \ . \tag{6}$$

Przeanalizujemy teraz dokładnie działanie algorytmu. Algorytm Deutscha wykonywany jest w trzech krokach.

Krok (1)

Pierwsza para bramek Hadamarda przekształca kubity wejściowe $|0\rangle$ i $|1\rangle$ w stan

$$\frac{1}{2}(|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle) \ , \tag{7}$$

gdzie symbol $\otimes$ oznacza iloczyn tensorowy wektorów stanu rejestru pierwszego (górnego) i drugiego (dolnego).

Krok (2)

Stan (7) otrzymany w kroku pierwszym poddany jest działaniu bramki U_f. W celu obliczenia wyniku tego działania wykorzystamy wzór (4)

$$U_f|x\rangle \otimes (|0\rangle - |1\rangle) = |x\rangle \otimes (|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle) \ . \tag{8}$$

Jeżeli $f(x) = 0$, to wyrażenie w nawiasie po prawej stronie równania (8) przyjmuje postać

$$|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle = |0\rangle - |1\rangle = (-1)^0(|0\rangle - |1\rangle) = (-1)^{f(x)}(|0\rangle - |1\rangle) \ . \tag{9}$$

Jeżeli $f(x) = 1$, to wyrażenie to przyjmuje postać

$$|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle = |0\rangle - |1\rangle = (-1)^1(|0\rangle - |1\rangle) = (-1)^{f(x)}(|0\rangle - |1\rangle) \; . \quad (10)$$

Ze wzorów (9) i (10) wynika, że ten sam wzór jest słuszny dla wszystkich wartości $f(x)$, czyli

$$U_f|x\rangle \otimes (|0\rangle - |1\rangle) = (-1)^{f(x)}|x\rangle \otimes (|0\rangle - |1\rangle) \; . \quad (11)$$

Stosując (11) do stanu (7) otrzymanego w wyniku kroku pierwszego otrzymujemy

$$U_f \frac{1}{2}(|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle) = \frac{1}{2}\left[(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\right] \otimes (|0\rangle - |1\rangle) \; . \quad (12)$$

Krok (3)

Działamy bramką Hadamarda na stan pierwszego rejestru w (12), czyli na wektor $(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle$, i otrzymujemy

$$\begin{aligned} & \tfrac{1}{2} \quad \left[(-1)^{f(0)}H|0\rangle + (-1)^{f(1)}H|1\rangle\right] \otimes (|0\rangle - |1\rangle) \\ = & \quad \frac{1}{2}\left[(-1)^{f(0)}\frac{1}{\sqrt{2}}(|0\rangle|1\rangle) + (-1)^{f(1)}\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right] \\ & \quad \otimes(|0\rangle - |1\rangle) \\ = & \quad \frac{1}{2}\left\{|0\rangle\left[(-1)^{f(0)} + (-1)^{f(1)}\right] + |1\rangle\left[(-1)^{f(0)} - (-1)^{f(1)}\right]\right\} \\ & \quad \otimes\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \; . \end{aligned} \quad (13)$$

Przeanalizujmy teraz wynik zapisany w ostatnich dwóch wierszach (13).
Jeżeli funkcja $f(x)$ jest stała, to zachodzi

$$(-1)^{f(0)} - (-1)^{f(1)} = 0 \; .$$

W tym przypadku stan końcowy górnego rejestru (stan górnej linii schematu) przyjmuje postać

$$\frac{1}{2}\left\{|0\rangle\left[(-1)^{f(0)} + (-1)^{f(1)}\right]\right\} = \pm|0\rangle \; . \quad (14)$$

Jeżeli funkcja $f(x)$ jest zrównoważona, to zachodzi

$$(-1)^{f(0)} + (-1)^{f(1)} = 0 \; .$$

W tym przypadku stan końcowy górnego rejestru (stan górnej linii schematu) przyjmuje postać

$$\frac{1}{2}\left\{|1\rangle\left[(-1)^{f(0)} - (-1)^{f(1)}\right]\right\} = \pm|1\rangle \; . \quad (15)$$

Wniosek

W celu określenia, czy funkcja $f(x)$ jest stała czy zrównoważona, wystarczy – po wykonaniu kroków (1-3) – dokonać pomiaru stanu wyjściowego górnego rejestru (stanu końcowego górnej linii). Jeżeli stanem końcowym jest $|0\rangle$, to funkcja $f(x)$ jest stała, a jeżeli stanem końcowym jest $|1\rangle$, to funkcja $f(x)$ jest zrównoważona.

A zatem algorytm Deutscha odpowiada na pytanie, czy funkcja $f(x)$ jest stała czy zrównoważona, obliczając wartość funkcji $f(x)$ tylko jeden raz (operacja U_f została zastosowana tylko raz).

Dyskusja

- Kubity w dolnej linii podlegają ewolucji czasowej, w której wyniku osiągany jest albo stan $|0\rangle$ albo $|1\rangle$. Wskutek splątania stanów dolnej i górnej linii powoduje to odpowiednią zmianę stanów linii górnej.
- Obliczenie dwóch wartości funkcji $f(x)$ za pomocą jednej operacji jest ilustracją **kwantowej równoległości obliczeń**.
- Całość działania algorytmu opiera się na **paradoksie EPR**.

  Signum temporis: **Obecnie to jest właśnie paradoks: istota działania komputera kwantowego opiera się na paradoksie EPR**.

Zastosowanie algorytmu Deutscha

Metody różnicowe fizyki obliczeniowej wymagają dokonania dyskretyzacji zmiennych i funkcji, czyli tabelaryzacji wartości funkcji. W wyniku dyskretyzacji otrzymujemy funkcję przedziałami stałą. W tym celu możemy zastosować algorytm Deutscha.

A zatem algorytm ten może być stosowany do odpowiedniego przyspieszenia implementacji metod różnicowych, używanych np. w różniczkowaniu numerycznym lub numerycznym całkowaniu (za pomocą metody prostokątów).

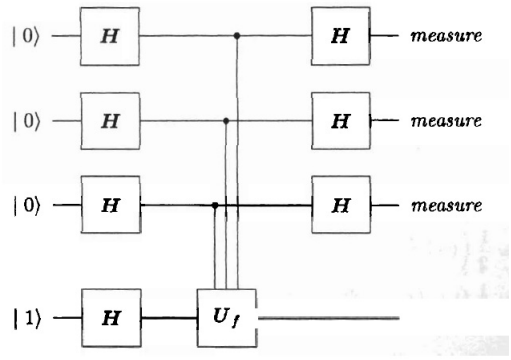
3 Algorytm Deutscha-Jozsy

Algorytm Deutscha-Jozsy jest wyrocznią, będącą uogólnieniem wyroczni Deutscha. Badana jest funkcja określona na n parach zmiennych.

$$f(x) : \{0,1\}^n \to \{0,1\} \, . \tag{16}$$

Algorytm ten odpowiada na pytanie, czy funkcja ta stała czy zrównoważona, przy funkcja (16) jest zrównoważona, jeżeli $f(x)$ przyjmuje wartość 0 dla jednej połowy swoich argumentów, a wartość 1 dla drugiej połowy.

W algorytmie Deutscha-Jozsy bramki Hadamarda działają analogicznie jak w algorytmie Deutscha, natomiast bramka U_f jest teraz kontrolowana przez stany n rejetrów (n linii). Funkcja f odwzorowuje zbiór 2^n liczb $\{0,1\}^n$ na parę liczb $\{0,1\}$. Funkcja ta jest albo stała albo zrównoważona.



Rysunek 2: Schemat obwodu kwantowego do implementacji algorytmu Deutscha-Jozsy.

Zadaniem algorytmu (wyroczni) Deutscha-Jozsy jest odpowiedzieć na pytanie, który z tych dwóch przypadków zachodzi dokonując wyłącznie jednego obliczenia wartości funkcji.

Klasyczny algorytm musi obliczyć funkcję f 2^n razy, aby odpowiedzieć na to pytanie.

Analizujemy poszczególne kroki algorytmu Deutscha-Jozsy.

Krok (1)

Działamy bramką H na każdy z n kubitów wejściowych $|0\rangle$.

W celu znalezienia wyniku tej operacji obliczamy

$$\begin{aligned} H \qquad & |0\rangle \;\; H\,|0\rangle \;\; \ldots \;\; H\,|0\rangle \\ = \;\; & \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\ldots\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ = \;\; & \frac{1}{2^{n/2}}(|00\ldots0\rangle + |00\ldots1\rangle + \ldots|11\ldots1\rangle) \\ = \;\; & \frac{1}{2^{n/2}}(|\mathbf{0}\rangle + |\mathbf{1}\rangle + \ldots + |\mathbf{2^n-1}\rangle) \\ = \;\; & \frac{1}{2^{n/2}}\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}|\mathbf{x}\rangle\,. \end{aligned} \tag{17}$$

W tym kroku przekształcany jest jeszcze kubit $|1\rangle$ w dolnym rejestrze (dolnej linii), co prowadzi do wyniku

$$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\,. \tag{18}$$

Ostatecznie po wykonaniu pierwszego kroku cały układ przechodzi do stanu

$$|\psi^{(1)}\rangle = \frac{1}{2^{n/2}}\left(\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}|\mathbf{x}\rangle\right)\otimes\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\,. \tag{19}$$

Krok (2)

W kroku tym zastosowana jest n-wierszowa kontrolowana operacja U_f. Uogólniając dla tej operacji wynik otrzymany poprzednio dla algorytmu Deutscha otrzymujemy stan układu po wykonaniu drugiego kroku

$$|\psi^{(2)}\rangle = \frac{1}{2^{n/2}} \left(\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}} (-1)^{f(\mathbf{x})} |\mathbf{x}\rangle \right) . \tag{20}$$

Krok (3)

W kroku końcowym poddajemy każdy z kubitów w n górnych liniach działaniu bramki Hadamarda. Jednak teraz stanem wejściowym każdego z tych rejestrów nie jest stan $|0\rangle$, a zatem transformacja Hadamarda staje się bardziej skomplikowana. W celu wyznaczenia jej wyniku przekształcamy znany wynik działania transformacji Hadamarda

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}} \left[(-1)^{0\times 0}|0\rangle + (-1)^{0\times 1}|1\rangle \right] , \tag{21}$$

$$H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}} \left[(-1)^{1\times 0}|0\rangle + (-1)^{1\times 1}|1\rangle \right] , \tag{22}$$

gdzie symbol $\times$ oznacza zwykłe mnożenie liczb ($x \times y = xy$).

Wzory (21) i (22) można zapisać w jednolity sposób jako

$$H|x\rangle = \frac{1}{\sqrt{2}} \sum_{y=0}^{1} (-1)^{x\times y} |y\rangle . \tag{23}$$

Zastosujemy teraz formułę (23) do iloczynu tensorowego n kubitów.

$$\begin{aligned}
|\psi^{(n)}\rangle = \quad &= \prod_{j=1}^{n} \otimes H|x_j\rangle \\
&= H|x_1\rangle \otimes H|x_2\rangle \otimes \ldots \otimes H|x_n\rangle \\
&= \left[\frac{1}{\sqrt{2}} \sum_{y_1=0}^{1} (-1)^{x_1\times y_1} |y_1\rangle \right] \otimes \left[\frac{1}{\sqrt{2}} \sum_{y_2=0}^{1} (-1)^{x_2\times y_2} |y_2\rangle \right] \\
&\quad \otimes \ldots \otimes \left[\frac{1}{\sqrt{2}} \sum_{y_n=0}^{1} (-1)^{x_n\times y_n} |y_n\rangle \right] \\
&= \frac{1}{2^{n/2}} \sum_{y_1 y_2 \ldots y_n=0}^{1} (-1)^{x_1\times y_1} (-1)^{x_2\times y_2} \ldots (-1)^{x_n\times y_n} \\
&\quad |y_1 y_2 \ldots y_n\rangle \\
&= \frac{1}{2^{n/2}} \sum_{\mathbf{y=0}}^{\mathbf{2^n-1}} (-1)^{\mathbf{x\cdot y}} |\mathbf{y}\rangle .
\end{aligned} \tag{24}$$

Iloczyn $\mathbf{x}\cdot\mathbf{y}$ we wzorze (24) ma następujące znaczenie:

$$\mathbf{x}\cdot\mathbf{y} = x_1 \times y_1 \oplus x_2 \times y_2 \oplus \ldots \oplus x_n \times y_n \ ,$$

gdzie $\oplus$ oznacza dodawanie modulo 2.

Wstawiamy wyrażenie (24) do stanu wynikowego drugiego kroku (20) i otrzymujemy wynik trzeciego kroku

$$\begin{aligned}
|\psi^{(3)}\rangle &= \frac{1}{2^{n/2}}\left[\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}(-1)^{f(\mathbf{x})}\prod_{j=1}^{n}\otimes H|x_j\rangle\right]\otimes\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \\
&= \frac{1}{2^{n/2}}\left[\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}(-1)^{f(\mathbf{x})}\frac{1}{2^{n/2}}\sum_{\mathbf{y=0}}^{\mathbf{2^n-1}}(-1)^{\mathbf{x}\cdot\mathbf{y}}|\mathbf{y}\rangle\right]\otimes\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \\
&= \left[\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}\sum_{\mathbf{y=0}}^{\mathbf{2^n-1}}(-1)^{f(\mathbf{x})}(-1)^{\mathbf{x}\cdot\mathbf{y}}|\mathbf{y}\rangle\right]\otimes\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle)\ . \qquad (25)
\end{aligned}$$

Przeanalizujemy teraz stan końcowy (25).

(I) Jeżeli funkcja $f(\mathbf{x})$ jest stała, to wyrażenie $(-1)^{f(\mathbf{x})}$ możemy wyciągnąć przed sumę podwójną, która przybiera postać

$$(-1)^{f(\mathbf{x})}\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}\sum_{\mathbf{y=0}}^{\mathbf{2^n-1}}(-1)^{\mathbf{x}\cdot\mathbf{y}}|\mathbf{y}\rangle\ . \qquad (26)$$

Ustalmy $|\mathbf{y}\rangle$ i rozważmy możliwe stany

$$|\psi_{\mathbf{y}}\rangle = \left[\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}(-1)^{\mathbf{x}\cdot\mathbf{y}}\right]|\mathbf{y}\rangle\ . \qquad (27)$$

Dla $\mathbf{y}\neq\mathbf{0}$ suma w wyrażeniu (27) musi się zerować, czyli

$$\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}(-1)^{\mathbf{x}\cdot\mathbf{y}} = 0\ , \qquad (28)$$

ponieważ w sumie tej wyrażenie $\mathbf{x}\cdot\mathbf{y}$ przyjmuje taką samą liczbę razy wartość 0 oraz 1. Zatem w wyrażeniu (27) pozostaje jedynie wyraz odpowiadający $\mathbf{y}=\mathbf{0}$.

W tym przypadku stan końcowy (25) przyjmuje postać

$$\begin{aligned}
|\psi^{(3)}\rangle &= \frac{1}{2^n}(-1)^{f(\mathbf{x})}\left[\sum_{\mathbf{x=0}}^{\mathbf{2^n-1}}(-1)^{\mathbf{x}\cdot\mathbf{0}}|\mathbf{0}\rangle\right]\otimes\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \\
&= (-1)^{f(\mathbf{x})}|\mathbf{0}\rangle\otimes\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle)\ . \qquad (29)
\end{aligned}$$

Wynik (29) oznacza, że dla funkcji $f(\mathbf{x})$, będącej funkcją stałą, pomiar stanu końcowego dowolnego rejestru górnego da zawsze stan $|0\rangle$.

(II) Jeżeli funkcja $f(\mathbf{x})$ jest zrównoważona, to np. dla $|\mathbf{y}\rangle = |\mathbf{0}\rangle$ otrzymujemy

$$\sum_{\mathbf{x}=\mathbf{0}}^{\mathbf{2^n-1}} (-1)^{f(\mathbf{x})}(-1)^{\mathbf{x}\cdot\mathbf{0}}|\mathbf{0}\rangle = \sum_{\mathbf{x}=\mathbf{0}}^{\mathbf{2^n-1}} (-1)^{f(\mathbf{x})}|\mathbf{0}\rangle = 0 \,, \tag{30}$$

ponieważ zrównoważona funkcja $f(\mathbf{x})$ tyle samo razy przyjmuje wartości 0 lub 1.

Wynika stąd, że – w odróżnieniu od przypadku (I) – prawdopodobieństwo znalezienia stanu $|\mathbf{y}\rangle = |\mathbf{0}\rangle$ jest równe zero.

Podsumowanie własności stanu końcowego otrzymanego po wykonaniu algorytmu Deutscha-Jozsy:

(I) Jeżeli stan wynikowy każdego rejestru kontrolnego jest stanem $|0\rangle$, to funkcja $f(\mathbf{x})$ jest stała.

(II) Jeżeli nie zachodzi ten przypadek, to funkcja $f(\mathbf{x})$ jest zrównoważona.

4 Algorytm Simona

Algorytm (wyrocznia) Simona stanowi dalsze uogólnienie algorytmu Deutscha. Rozważamy funkcję

$$\mathbf{f}(\mathbf{x}) : \{0,1\}^n \to \{0,1\}^n \,, \tag{31}$$

gdzie $\mathbf{x}$ i $\mathbf{f}$ są dwuwymiarowymi wektorami o składowych przyjmujących wartości należące do zbioru $\{0,1\}^n$. Zakładamy, że funkcja (31) jest **dwuznaczna**, czyli dla każdej wartości funkcji $\mathbf{y} = \mathbf{f}(\mathbf{x})$ zawsze istnieją dwa wektory $\mathbf{x}_1$ i $\mathbf{x}_2$ takie, że $\mathbf{f}(\mathbf{x}_1) = \mathbf{f}(\mathbf{x}_2)$. Ponadto zakładamy, że funkcja (31) jest **periodyczna**, co oznacza, że istnieje taki wektor binarny $\mathbf{a}$, że

$$\mathbf{f}(\mathbf{x} \oplus \mathbf{a}) = \mathbf{f}(\mathbf{x}) \,. \tag{32}$$

Algorytm Simona znajduje wektor $\mathbf{a}$, czyli **okres** funkcji $\mathbf{f}(\mathbf{x})$ w $\mathcal{O}(n)$ próbach.

Dla porównania algorytm klasyczny znajduje okres funkcji przy użyciu L prób, których liczba rośnie eksponencjalnie z n, czyli $L \sim \mathcal{O}(\exp(n))$.

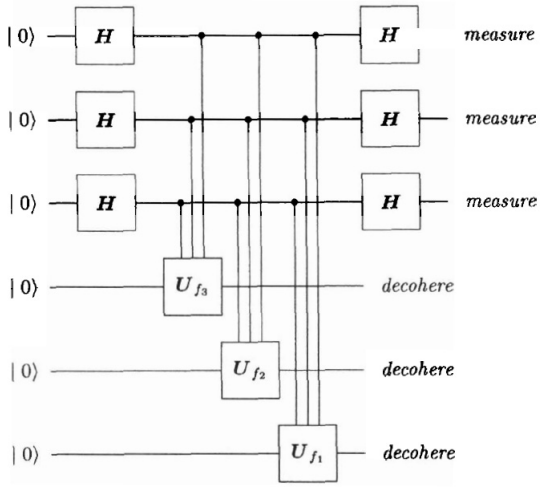
Obwód kwantowy do implementacji algorytmu Simona zawiera w ogólnym przypadku n linii górnych, które wyglądają tak samo jak górne linie w obwodzie kwantowym algorytmu Deutscha-Jozsy, oraz n linii dolnych. Każda z linii dolnych odpowiada sub-funkcji

$$f_j : \{0,1\}^n \to \{0,1\} \,,$$

gdzie $j = 1, \ldots, n$. Z n sub-funkcji f_j tworzymy funkcję $\mathbf{f}$.

Zaznaczone na schemacie operacje U_{f_j} są bramkami kontrolowanymi, przy czym sterowanie odbywa się za pomocą funkcji f_j.

Najpierw sformułujemy zadanie, które algorytm Simona ma rozwiązać.



Rysunek 3: Schemat obwodu kwantowego do implementacji algorytmu Simona na przykładzie funkcji $f : \{0,1\}^3 \to \{0,1\}^3$.

Algorytm ten testuje funkcję wektorową

$$\mathbf{f} : \{0,1\}^n \to \{0,1\}^n \ , \tag{33}$$

która w obwodzie kwantowym zostaje poddana dekompozycji na n funkcji skalarnych

$$f_j : \{0,1\}^n \to \{0,1\} \ , \tag{34}$$

gdzie $j = 1, \ldots, n$.

Funkcja $\mathbf{f}$ musi spełniać następujące warunki:

(1) Funkcja $\mathbf{f}$ jest **funkcją dwuznaczną**, tzn. każdej wartości $\mathbf{f}$ zawsze odpowiadają dwa wektory $\mathbf{x}_1$ i $\mathbf{x}_2$ takie, że

$$\mathbf{f}(\mathbf{x}_1) = \mathbf{f}(\mathbf{x}_2) \ .$$

(2) Funkcja $\mathbf{f}$ jest **funkcją periodyczną**, tzn. istnieje taki wektor $\mathbf{a}$, że

$$\mathbf{f}(\mathbf{x} \oplus \mathbf{a}) = \mathbf{f}(\mathbf{x}) \ .$$

Przeanalizujmy działanie algorytmu Simona.

Krok (1)

Transformacja Hadamarda zastosowana w n górnych liniach obwodu kwantowego działa tak samo jak w przypadku algorytmu Deutscha-Jozsy. Użyjemy

zatem gotowych wyników otrzymanych w kroku (1) tego algorytmu. Otrzymujemy następujący stan wynikowy po pierwszym kroku:

$$\begin{aligned} |\psi^{(1)}\rangle &= \frac{1}{2^{n/2}} \left(\sum_{\mathbf{x}=\mathbf{0}}^{\mathbf{2^n-1}} |\mathbf{x}\rangle \right) \otimes |0\rangle|0\rangle \ldots |0\rangle \\ &= \frac{1}{2^{n/2}} \left(\sum_{\mathbf{x}=\mathbf{0}}^{\mathbf{2^n-1}} |\mathbf{x}\rangle \right) \otimes |\mathbf{0}\rangle \ . \end{aligned} \tag{35}$$

Krok (2)

W kroku tym stosowane są sterowane bramki U_{f_j}, które w rezultacie przekształcają stany dolnych n linii z $|\mathbf{0}\rangle$ w $|f_j(\mathbf{x})\rangle$.

Wynika to z następującego rozumowania:

Dla każdej dolnej linii w stanie $|0\rangle$ odpowiadająca jej funkcja $f_j(\mathbf{x})$ może przyjąć albo wartość 1 i wtedy stan linii jest zmieniany na stan $|1\rangle$, albo wartość 0 i wtedy stan linii pozostaje niezmieniony, a zatem – po tej operacji – każda dolna linia znajdzie się w stanie $|f_j(\mathbf{x})\rangle$.

Stanem układu po wykonaniu kroku (2) jest stan

$$|\psi^{(2)}\rangle = \frac{1}{2^{n/2}} \left(\sum_{\mathbf{x}=\mathbf{0}}^{\mathbf{2^n-1}} |\mathbf{x}\rangle \right) \otimes |\mathbf{f}(\mathbf{x})\rangle \ . \tag{36}$$

Krok (3)

Pozwalamy teraz, aby stany dolnych linii ewoluowały w kontakcie z otoczeniem, co oznacza dekoherencję tych stanów. W wyniku dekoherencji zostanie osiągnięty pewien stan, który oznaczamy jako $|\mathbf{f}(\mathbf{x}_0)\rangle$. Ze powodu periodyczności funkcji $\mathbf{f}$ z jednakowym prawdopodobieństwem może zostać osiągnięty stan $|\mathbf{f}(\mathbf{x}_0 \oplus \mathbf{a})\rangle$, a zatem stan n dolnych linii znajdzie się w stanie

$$|\mathbf{f}(\mathbf{x}_0)\rangle = |\mathbf{f}(\mathbf{x}_0 \oplus \mathbf{a})\rangle \ .$$

Korzystamy teraz z tego, że stany linii dolnych i górnych są splątane, a zatem z superpozycji 2^n stanów (36) wybrane zostaną tylko dwa stany i stan linii górnych staje się ich kombinacją liniową (z jednakowymi amplitudami).

Ostatecznie stanem układu po wykonaniu kroku (3) jest stan

$$|\psi^{(3)}\rangle = \frac{1}{\sqrt{2}}(|\mathbf{x}_0\rangle + |\mathbf{x}_0 \oplus \mathbf{a}\rangle) \otimes |\mathbf{f}(\mathbf{x}_0)\rangle \ . \tag{37}$$

Krok (4)

Zgodnie ze schematem obwodu kwantowego (por. rysunek) n górnych linii zastaje poddanych działaniu bramek Hadamarda, co prowadzi do wyniku

$$|\psi^{(4)}\rangle = \frac{1}{\sqrt{2^{n+1}}} \left\{ \sum_{\mathbf{y}=\mathbf{0}}^{\mathbf{2^n-1}} \left[(-1)^{\mathbf{x_0 \cdot y}} + (-1)^{(\mathbf{x}_0 \oplus \mathbf{a}) \cdot \mathbf{y}} \right] |\mathbf{y}\rangle \right\} \otimes |\mathbf{f}(\mathbf{x}_0)\rangle \ . \tag{38}$$

Uwaga Wielkości typu $\mathbf{x}$ są tutaj traktowane jak macierze $1 \times n$, których elementami są liczby 0 lub 1. Wynika stąd, że iloczyny $\mathbf{x} \cdot \mathbf{y}$ we wzorze (38) mają następujące znaczenie:

$$\mathbf{x} \cdot \mathbf{y} = x_0 y_0 \oplus x_1 y_1 \oplus x_2 y_2 \oplus \ldots \oplus x_{n-1} y_{n-1} \; . \tag{39}$$

Zbiór wektorów $\{\mathbf{y}\}$ można podzielić na dwie klasy:

(1) dla pierwszej klasy $\mathbf{y} \cdot \mathbf{a} = 1$,

(2) dla drugiej klasy $\mathbf{y} \cdot \mathbf{a} = 0$.

Dla pierwszej klasy przed każdym z wektorów stanu $|\mathbf{y}\rangle$ w rozwinięciu (38) występuje amplituda

$$(-1)^{\mathbf{x_0} \cdot \mathbf{y}} - (-1)^{\mathbf{x_0} \cdot \mathbf{y}} = 0 \; .$$

Wynika stąd, że wektory $|\mathbf{y}\rangle$ należące do klasy (1) dają zerowy wkład do rozwinięcia (38).

A zatem jedynie wektory $\mathbf{y}$ należące do klasy (2) (prostopadłe do wektora $\mathbf{a}$) wnoszą niezerowe przyczynki do sumy (36). Prowadzi to do następującej postaci stanu końcowego

$$|\psi^{(4)}\rangle = \frac{1}{\sqrt{2^{n+1}}} \left[\sum_{\mathbf{y} \cdot \mathbf{a} = 0} (-1)^{\mathbf{x_0} \cdot \mathbf{y}} |\mathbf{y}\rangle \right] \otimes |\mathbf{f}(\mathbf{x}_0)\rangle \; . \tag{40}$$

Otrzymany stan końcowy (40) oznacza, że dokonując pomiaru n górnych linii dostaniemy zawsze wektor $\mathbf{y}$, który jest prostopadły do szukanego wektora $\mathbf{a}$. To jeszcze nie określa jednoznacznie wektora $\mathbf{a}$. Jeżeli jednak przeprowadzimy pomiar stanu górnych linii n razy, to otrzymamy układ n równań

$$\begin{aligned} \mathbf{y_1} \cdot \mathbf{a} &= 0 \\ \mathbf{y_2} \cdot \mathbf{a} &= 0 \\ &\ldots \\ \mathbf{y_n} \cdot \mathbf{a} &= 0 \end{aligned} \tag{41}$$

na n niewiadomych $\{a_1, a_2, \ldots, a_n\}$. Ten układ równań może być rozwiązany metodami klasycznej algebry liniowej.

W ten sposób znajdujemy szukany okres funkcji $\mathbf{f}$.

Dyskusja algorytmu Simona

(1) Zastosowanie transformacji Hadamarda do stanów początkowych $|0\rangle$ n górnych linii wytwarza superpozycję $2^n - 1$ stanów w pojedynczym rejestrze. Dzięki sprzężeniu z dolnymi liniami obliczana jest wartość funkcji $\mathbf{f}$ w tej samej chwili czasu dla wszystkich argumentów funkcji. Ten pojedynczy krok w obliczeniach kwantowych jest równoważny $2^n - 1$ krokom w obliczeniach klasycznych. Ujawnia się tutaj równoległość obliczeń kwantowych.

(2) Zastosowanie sterowanych bramek U_{f_j} wytwarza stan splątany całego układu.

(3) Korelacja kwantowa powoduje, że ewolucja czasowa (dekoherencja) n stanów linii dolnych pociąga za sobą przekształcenie stanów n linii górnych, splątanych ze stanami linii dolnych, do superpozycji stanów $|\mathbf{x}_0\rangle$ i $|\mathbf{x}_0 \oplus \mathbf{a}\rangle$.

(4) Ostatnia operacja kwantowa prowadzi do oddzielenia wektora $\mathbf{x}_0$ z tej superpozycji i pozostawienia superpozycji wektorów prostopadłych do $\mathbf{a}$.

W algorytmie Simona uwidocznione są podstawowe cechy obliczeń kwantowych:

- **superpozycja ≡ równoległość**
- **splątanie ≡ nielokalność**
- **korelacja kwantowa**

5 Kwantowa transformata Fouriera

Kwantowa transformata Fouriera (QFT) jest operacją unitarną zdefiniowaną na n kubitach w sposób następujący:

$$\mathbf{F} : |\mathbf{x}\rangle \rightarrow \frac{1}{\sqrt{2^n}} \sum_{\mathbf{y}=0}^{2^n-1} e^{2\pi i x y/2^n} |\mathbf{y}\rangle \ , \tag{42}$$

gdzie xy jest zwykłym mnożeniem dwóch liczb. Liczby te wygodnie jest zapisać w **reprezentacji binarnej**, czyli

$$x = x_0 + x_1 2 + x_2 2^2 + x_3 2^3 + \ldots + x_{n-1} 2^{n-1} \ , \tag{43}$$

$$y = y_0 + y_1 2 + y_2 2^2 + y_3 2^3 + \ldots + y_{n-1} 2^{n-1} \ . \tag{44}$$

Liczbom x i y odpowiadają stany n-kubitowe

$$|\mathbf{x}\rangle = |x_0\rangle \otimes |x_1\rangle \otimes \ldots \otimes |x_{n-1}\rangle \ , \tag{45}$$

$$|\mathbf{y}\rangle = |y_0\rangle \otimes |y_1\rangle \otimes \ldots \otimes |y_{n-1}\rangle \ , \tag{46}$$

gdzie $|x_j\rangle$ i $|y_j\rangle$ są pojedynczymi kubitami, a liczby x_j, y_j przyjmują wartości 0, 1.

Transformatę Fouriera dowolnego wektora $|\mathbf{w}\rangle$ wyznaczymy, jeżeli będziemy znali wynik działania QFT na wektory bazy $|\mathbf{x}\rangle$, ponieważ dowolny wektor $|\mathbf{w}\rangle$ można zapisać w postaci

$$|\mathbf{w}\rangle = \sum_{\mathbf{x}} f(\mathbf{x}) |\mathbf{x}\rangle \ . \tag{47}$$

Na podstawie (42) i (47) otrzymujemy następujący wzór na kwantową transformatę Fouriera funkcji $f(\mathbf{x})$:

$$\begin{aligned}\mathbf{F}\left[\sum_{\mathbf{x}=0}^{N-1} f(\mathbf{x})|\mathbf{x}\rangle\right] &= \sum_{\mathbf{x}=0}^{N-1} f(\mathbf{x})\mathbf{F}(|\mathbf{x}\rangle) \\ &= \frac{1}{\sqrt{N}}\sum_{\mathbf{x}=0}^{N-1} f(\mathbf{x})\sum_{\mathbf{y}=0}^{N-1} e^{2\pi i xy/N}|\mathbf{y}\rangle \\ &= \frac{1}{\sqrt{N}}\sum_{\mathbf{y}=0}^{N-1}\sum_{\mathbf{x}=0}^{N-1} f(\mathbf{x})e^{2\pi i xy/N}|\mathbf{y}\rangle\ , \end{aligned} \tag{48}$$

gdzie N jest liczbą elementów bazy $|\mathbf{x}\rangle$ (niekoniecznie równą 2^n).

Na podstawie wzoru (48) otrzymujemy wzór na składową $\mathbf{y}$ transformaty Fouriera funkcji f

$$\mathbf{F}_{\mathbf{y}}(f) = \frac{1}{\sqrt{N}}\sum_{\mathbf{x}=0}^{N-1} f(\mathbf{x})e^{2\pi i xy/N}\ . \tag{49}$$

Wzór (49) jest wzorem na klasyczną **dyskretną transformatę Fouriera**.

Konstrukcja obwodu kwantowego realizującego QFT opiera się na metodzie **szybkiej transformaty Fouriera (FFT)**.

Rozważmy wyrażenie

$$e^{2\pi i xy/2^n} \tag{50}$$

występujące we wzorze (42).

Wyrażenie (50) jest funkcją periodyczną iloczynu zmiennych xy o okresie 2^n. W oparciu o tę periodyczność stosujemy podstawowy trik metody FFT, który polega na obliczaniu wyłącznie wyrazów pierwszego okresu, czyli takich, dla których $xy/2^n < 1$. Natomiast wyrazy odpowiadające okresowi drugiemu i wyższym są obcinane.

A zatem obliczamy

$$\begin{aligned}\frac{xy}{2^n} &= \frac{1}{2^n}(x_0 + x_1 2 + x_2 2^2 + \ldots + x_{n-1}2^{n-1}) \\ &\times \quad (y_0 + y_1 2 + y_2 2^2 + \ldots + y_{n-1}2^{n-1}) = \ldots \end{aligned} \tag{51}$$

W równaniu (51) każda z liczb x i y została przedstawiona za pomocą reprezentacji binarnej, a zatem składowe x_j i y_j przyjmują wartości 0 lub 1.

W dalszym ciągu przekształcamy prawą stronę równania (51) stosując re-

prezentację binarną x_j i y_j.

$$\begin{aligned}
= \quad & \tfrac{1}{2^n} \quad \big[y_0(x_0 + x_1 2^1 + x_2 2^2 + \ldots + x_{n-1}2^{n-1}) \\
& \quad +y_1 2^1(x_0 + x_1 2^1 + x_2 2^2 + \ldots + x_{n-1}2^{n-1}) \\
& \quad +\ldots \\
& \quad +y_{n-1}2^{n-1}(x_0 + x_1 2^1 + x_2 2^2 + \ldots + x_{n-1}2^{n-1})\big] \\
\overset{obc}{=} \quad & \tfrac{1}{2^n} \quad \big[y_0(x_0 + x_1 2^1 + x_2 2^2 + \ldots + x_{n-1}2^{n-1}) \\
& \quad +y_1(x_0 2^1 + x_1 2^2 + x_2 2^3 + \ldots + x_{n-2}2^{n-1}) \\
& \quad +y_2(x_0 2^2 + x_1 2^3 + x_2 2^4 + \ldots + x_{n-3}2^{n-1}) \\
& \quad +\ldots \\
& \quad +y_{n-1}x_0 2^{n-1}\big] = \ldots
\end{aligned} \tag{52}$$

Symbol $\overset{obc}{=}$ oznacza równość wyrazów pierwszego okresu. Natomiast wyrazy odpowiadające okresom drugiemu i wyższym zostały obcięte.

Otrzymujemy ostatecznie

$$\begin{aligned}
= \quad & y_0\left(\tfrac{x_0}{2^n} + \tfrac{x_1}{2^{n-1}} + \ldots + \tfrac{x_{n-1}}{2}\right) \\
& +y_1\left(\tfrac{x_0}{2^{n-1}} + \tfrac{x_1}{2^{n-2}} + \ldots + \tfrac{x_{n-2}}{2}\right) \\
& +y_2\left(\tfrac{x_0}{2^{n-2}} + \tfrac{x_1}{2^{n-3}} + \ldots + \tfrac{x_{n-3}}{2}\right) \\
& \qquad +\ldots \\
& \qquad +y_{n-1}\tfrac{x_0}{2}\ .
\end{aligned} \tag{53}$$

Wyrażenia w nawiasach w (53) można zapisać w postaci **binarnego ułamka** z zastosowaniem następującej notacji:

$$\begin{aligned}
\frac{x_0}{2} \quad & \to \quad (.x_0) \\
\frac{x_0}{2^2} + \frac{x_1}{2} \quad & \to \quad (.x_0x_1) \\
\frac{x_0}{2^3} + \frac{x_1}{2^2} + \frac{x_2}{2} \quad & \to \quad (.x_0x_1x_2) \\
& \ldots \quad .
\end{aligned} \tag{54}$$

Przy użyciu tej notacji otrzymujemy

$$\frac{xy}{2^n} = y_0(.x_0x_1\ldots x_{n-1}) + y_1(.x_0x_1\ldots x_{n-2}) + \ldots y_{n-1}(.x_0)\ . \tag{55}$$

Korzystając z (55) przepisujemy QFT jako

$$\begin{aligned}
\mathbf{F}|\mathbf{x}\rangle &= \frac{1}{\sqrt{2^n}} \sum_{\mathbf{y}=0}^{2^n-1} e^{2\pi i x y/2^n} |\mathbf{y}\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{\mathbf{y}=0}^{2^n-1} e^{2\pi i [y_0(.x_0x_1\ldots x_{n-1})+y_1(.x_0x_1\ldots x_{n-2})+\ldots+y_{n-1}(.x_0)]} |\mathbf{y}\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{\mathbf{y}=0}^{2^n-1} e^{2\pi i y_0(.x_0x_1\ldots x_{n-1})} |y_0\rangle \otimes e^{2\pi i y_1(.x_0x_1\ldots x_{n-2})} |y_1\rangle \otimes \\
\ldots &\otimes e^{2\pi i y_{n-1}(.x_0)} |y_{n-1}\rangle \, . \qquad (56)
\end{aligned}$$

Korzystamy teraz z tego, że każda z liczb y_j jest równa 0 lub 1.
Jeżeli np. $y_1 = 0$, to otrzymujemy odpowiedni wyraz rozwinięcia (56) równy

$$e^{2\pi i 0(.x_0x_1\ldots x_{n-1})}|0\rangle = |0\rangle \, .$$

Jeżeli $y_1 = 1$, to odpowiedni wyraz rozwinięcia (56) jest równy

$$e^{2\pi i 1(.x_0x_1\ldots x_{n-1})}|1\rangle = e^{2\pi i(.x_0x_1\ldots x_{n-1})}|1\rangle \, .$$

Wysumowanie po wszystkich możliwych wartościach $\mathbf{y}$ w równaniu (56) powoduje przypisanie każdej wartości y_j zarówno 0 jak i 1, a zatem otrzymujemy

$$\begin{aligned}
\mathbf{F}|\mathbf{x}\rangle &= \frac{1}{\sqrt{2}} \left[|0\rangle + e^{2\pi i(.x_0x_1\ldots x_{n-1})}|1\rangle \right] \\
&\otimes \frac{1}{\sqrt{2}} \left[|0\rangle + e^{2\pi i(.x_0x_1\ldots x_{n-2})}|1\rangle \right] \\
&\otimes \ldots \otimes \frac{1}{\sqrt{2}} \left[|0\rangle + e^{2\pi i(.x_0)}|1\rangle \right] \, . \qquad (57)
\end{aligned}$$

Implementacja algorymu QFT

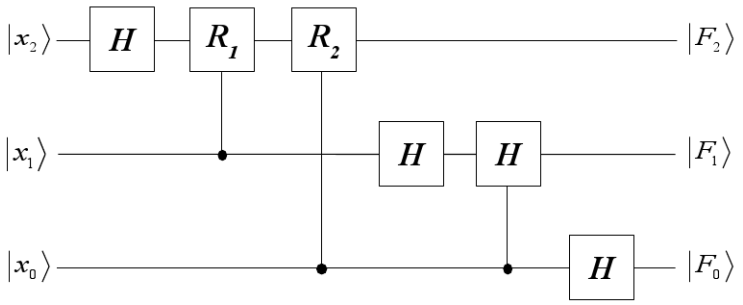
Na schemacie obwodu kwantowego H oznacza bramkę Hadamarda. Natomiast dla każdej linii obwodu kwantowego R_d oznacza bramkę zmiany fazy zdefiniowaną jako

$$R_d = \begin{pmatrix} I & 0 \\ 0 & e^{i\pi/2^d} \end{pmatrix} , \qquad (58)$$

gdzie $d = 1, 2, \ldots n$, a I jest macierzą jednostkową 3×3.

Zgodnie ze schematem obwodu kwantowego, że bramka R_d działa zawsze na dwa kubity. Np. R_1 działa na kubity górny i środkowy, czyli na $|x_2\rangle$ i $|x_1\rangle$, przy czym kubit środkowy, czyli w tym przypadku $|x_1\rangle$, jest kubitem sterującym Jest to zatem sterowana bramka dwukubitowa CR_d, której reprezentacja macierzową jest macierz 4×4 zapisana teraz w postaci

$$CR_d = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\pi/2^d} \end{pmatrix} \, . \qquad (59)$$



Rysunek 4: Schemat obwodu kwantowego do implementacji algorytmu QFT.

Zgodnie ze schematem obwodu algorytm QFT wykonywany jest w 5 krokach.

Krok (1)

Pierwsza bramka Hadamarda działająca na stan $|x_2\rangle$ rejestru górnego (górnej linii) daje w wyniku

$$\begin{aligned} H|x_2\rangle &= \frac{1}{\sqrt{2}}\sum_{y=0}^{1}(-1)^{x_2 y}|y\rangle \\ &= \frac{1}{\sqrt{2}}\sum_{y=0}^{1}e^{2\pi i x_2 y/2}|y\rangle \\ &= \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_2)}|1\rangle\right] . \end{aligned} \tag{60}$$

Krok (2)

Działamy bramką R_1 na stan górnego rejestru, otrzymany w wyniku 1. kroku (60), przy czym rejestr środkowy (stan $|x_1\rangle$ linii środkowej) pełni rolę sterującą.

Korzystamy z reprezentacji macierzowych bramki sterowanej CR_1 [wzór (59)] i dwukubitowej bazy obliczeniowej [por. wykład 5., wzory (32-35)]. Otrzymujemy następujące wyniki działania bramki CR_1 na stany $|x_1\rangle|x_2\rangle \equiv |x_1 x_2\rangle$.

Dla dowolnego kubitu sterującego $|x_1\rangle$ stan $|0\rangle$ rejestru górnego nie ulega zmianie, czyli

$$CR_1|x_1\rangle|0\rangle = |x_1\rangle|0\rangle \ . \tag{61}$$

Natomiast stan $|1\rangle$ rejestru górnego doznaje zmiany fazy w zależności od stanu kubitu sterującego.

Jeżeli kubit sterujący $|x_1\rangle = |1\rangle$, to

$$CR_1|x_1\rangle|1\rangle = e^{i\pi/2}|x_1\rangle|1\rangle \ , \tag{62}$$

a jeżeli $|x_1\rangle = |0\rangle$, to

$$CR_1|x_1\rangle|1\rangle = |x_1\rangle|1\rangle \ . \tag{63}$$

Wzory (62) i (63) można zapisać w jednolitej postaci

$$CR_1|x_1\rangle|1\rangle = e^{i\pi x_1/2}|x_1\rangle|1\rangle \ , \tag{64}$$

która jest słuszna dla dowolnego kubitu sterującego $|x_1\rangle$.

Stosujemy teraz (64) do stanów rejestru górnego (60) i środkowego $|x_1\rangle$.

$$\begin{aligned} CR_1 \quad & |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_2)}|1\rangle\right] \\ = \quad & |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_2)}e^{2\pi i x_1/2}|1\rangle\right] \\ = \quad & |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(x_2/2+x_1/4)}|1\rangle\right] \\ = \quad & |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_1x_2)}|1\rangle\right] \ . \end{aligned} \tag{65}$$

Krok (3)

Działamy bramką CR_2 w górnej linii, ale teraz kubitem sterującym jest kubit zapisany w rejestrze dolnym, czyli $|x_0\rangle$. W tym przypadku operator CR_2 nie zmienia kubitu $|0\rangle$, natomiast zmienia czynnik fazowy kubitu $|1\rangle$ o wartość $\exp(i\pi x_0/4)$. Operacja ta przekształca stan całego układu w sposób następujący:

$$\begin{aligned} CR_2 \quad & |x_0\rangle \otimes |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(x_1x_2)}|1\rangle\right] \\ = \quad & |x_0\rangle \otimes |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(x_1x_2)}e^{\pi i x_0/4}|1\rangle\right] \\ = \quad & |x_0\rangle \otimes |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(x_0/8+x_1/4+x_2/2)}|1\rangle\right] \\ = \quad & |x_0\rangle \otimes |x_1\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0x_1x_2)}|1\rangle\right] \ . \end{aligned} \tag{66}$$

Krok (4)

Na podstawie podobnych obliczeń jak w krokach (1-3) pokazujemy, że stan rejestru środkowego, czyli stan $|x_1\rangle$, w wyniku działania bramek H i CR_1 ulega przekształceniu w stan

$$\frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0x_1)}|1\rangle\right] \ . \tag{67}$$

A zatem stan całego układu przechodzi w stan

$$|x_0\rangle \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0x_1)}|1\rangle\right] \otimes \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0x_1x_2)}|1\rangle\right] \ . \tag{68}$$

Krok (5)

Stosujemy pojedynczą bramkę Hadamarda w dolnej linii do stanu rejestru dolnego, czyli $|x_0\rangle$. Bramka Hadamarda przekształca ten stan następująco:

$$H|x_0\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0)}|1\rangle\right] . \tag{69}$$

Ostatecznie otrzymujemy następujący stan końcowy układu:

$$\begin{aligned} & \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0)}|1\rangle\right] \\ \otimes \quad & \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0x_1)}|1\rangle\right] \\ \otimes \quad & \frac{1}{\sqrt{2}}\left[|0\rangle + e^{2\pi i(.x_0x_1x_2)}|1\rangle\right] . \end{aligned} \tag{70}$$

Stan (70) jest stanem wynikowym kwantowej transformaty Fouriera 3-kubitowego stanu początkowego [por. (57)].

Przedstawiona powyżej QFT dla stanu 3-kubitowego może być uogólniona na dowolny stan n-kubitowy. W tym celu należy zmodyfikować schemat obwodu QFT wprowadzając n rejestrów wejściowych i n linii. W liniach tych działamy następującymi iloczynami operatorów $HCR_1CR_2 \times \ldots \times CR_{n-1}$ w 1. linii, $HCR_1CR_2 \times \ldots \times CR_{n-2}$ w 2. linii, $\ldots$ H w linii n-tej.

W wyniku otrzymujemy stan końcowy (57) realizujący QFT dla dowolnego stanu n-kubitowego.

6 Algorytm Shora

Algorytm Shora znajduje okres funkcji $y = f(x)$, która jest określona na zbiorze liczb całkowitych $x \in \mathcal{N}$ i przyjmuje wartości całkowite $y \in \mathcal{N}$. W odróżnieniu od algorytmu Simona zarówno funkcja jak i jej okres są zdefiniowane na zbiorze liczb całkowitych $\mathcal{N}$.

Rozważamy funkcję okresową $f(x)$ o okresie r, tzn.

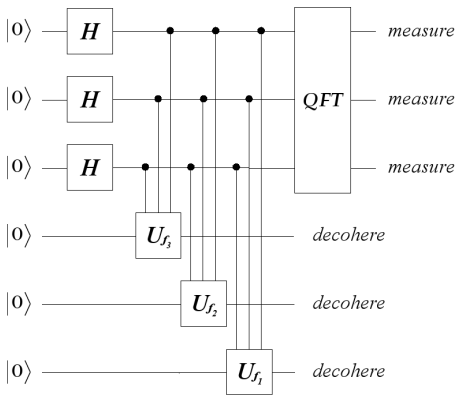
$$f(x) = f(x + kr) , \tag{71}$$

gdzie k jest liczbą całkowitą.

Schemat (5) do implemetacji algorytmu Shora skonstruowany został na podstawie schematu algorytmu Simona w ten sposób, że w górnym prawym rogu schematu – zamiast kolumny bramek Hadamarda – wstawiony został obwód kwantowy do implementacji kwantowej transformaty Fouriera.

Formułujemy problem tak, aby można było użyć komputera (klasycznego lub kwantowego) do jego rozwiązania. Musimy się zatem ograniczyć do skończonej liczby bitów (kubitów) i zapisać funkcję f w postaci binarnej. Przepisujemy więc funkcję f jako

$$f : x \in \{0,1\}^n \to y = f(x) \in \{0,1\}^n . \tag{72}$$



Rysunek 5: Schemat obwodu kwantowego do implementacji algorytmu Shora.

Szukamy okresu r funkcji $y = f(x)$, przy czym okres ten jest liczbą całkowitą należącą do zbioru

$$r \in [1, 2^n] \, .$$

Na podstawie schematu (5) przeanalizujemy działanie algorytmu Shora.

Pierwsze dwa kroki algorytmu przebiegają dokładnie tak samo jak w algorytmie Simona. Powtórzymy je w skrócie.

Pierwsza kolumna bramek Hadamarda generuje w górnych liniach superpozycję stanów numerowanych liczbami całkowitymi od 0 do $2^n - 1$, czyli

$$|\psi^{(1)}\rangle = \frac{1}{2^{n/2}} \sum_{\mathbf{x=0}}^{\mathbf{2^n-1}} |\mathbf{x}\rangle \, . \tag{73}$$

Zastosowanie w dolnych liniach kontrolowanych bramek U_{f_j} prowadzi do stanu całego układu

$$|\psi^{(2)}\rangle = \frac{1}{2^{n/2}} \sum_{\mathbf{x=0}}^{\mathbf{2^n-1}} |\mathbf{x}\rangle \otimes |f(\mathbf{x})\rangle \, , \tag{74}$$

gdzie

$$|f(\mathbf{x})\rangle = |f_{n-1}(\mathbf{x})\rangle \otimes |f_{n-2}(\mathbf{x})\rangle \otimes \ldots \otimes |f_0(\mathbf{x})\rangle \, .$$

Podobnie jak w algorytmie Simona pozwalamy, aby ewolucja czasowa (dekoherencja) doprowadziła kubity w dolnych liniach do pewnego stanu $|f(x_0)\rangle$. Ze względu na okresowość funkcji f ta sama wartość odpowiada argumentom funkcji $x_0 + r$, $x_0 + 2r$, itd. Splątanie stanów linii dolnych i górnych prowadzi do tego, że stan całego układu przechodzi w

$$|\psi^{(3)}\rangle = \frac{1}{\sqrt{M}} \left(\sum_{k=0}^{M-1} |x_0 + kr\rangle \right) \oplus |f(x_0\rangle \, , \tag{75}$$

gdzie stała normalizacyjna M jest taką liczbą całkowitą, że translacje

$$x_0 \to x_0 + r \to x_0 + 2r \to \ldots$$

nie wyprowadzą wartości argumentów poza przedział $[1, 2^n]$.

Od tego etapu zapominamy o liniach dolnych i koncentrujemy się wyłącznie na transformacji stanów linii górnych. Są to superpozycje stanów [wzór (75), wyrazy po lewej przed symbolem $\otimes$]. Stany tych linii poddawane są kwantowej transformacie Fouriera, która działa następująco:

$$\begin{aligned} F\left(\frac{1}{\sqrt{M}}\sum_{k=0}^{M-1}|x_0+kr\rangle\right) &= \frac{1}{2^{n/2}\sqrt{M}}\sum_{\mathbf{y=0}}^{\mathbf{2^n-1}}\sum_{k=0}^{M} e^{2\pi i(x_0+kr)y/2^n}|\mathbf{y}\rangle \\ &= \frac{1}{2^{n/2}\sqrt{M}}\sum_{\mathbf{y=0}}^{\mathbf{2^n-1}} e^{2\pi i x_0 y/2^n} \\ &\times \sum_{k=0}^{M} e^{2\pi i k r y/2^n}|\mathbf{y}\rangle\ . \end{aligned} \tag{76}$$

Rozważamy teraz ustalony wektor stanu $|\mathbf{y}\rangle$. Jeżeli wykonamy pomiar w stanie (76), to prawdopodobieństwo znalezienia stanu $|\mathbf{y}\rangle$ dane jest kwadratem odpowiedniej amplitudy, czyli

$$P(y) = \frac{M}{2^n}\left|\frac{1}{M}\sum_{k=0}^{M} e^{2\pi i k r y/2^n}\right|^2\ . \tag{77}$$

Zakładamy, że 2^n jest podzielne przez r bez reszty. Wtedy

$$M = \frac{2^n}{r} \Longrightarrow \frac{M}{2^n} = \frac{1}{r}$$

i prawdopodobieństwo (77) staje się równe

$$P(y) = \frac{1}{r}\left|\frac{1}{M}\sum_{k=0}^{M} e^{2\pi i k r y/M}\right|^2\ . \tag{78}$$

Dla $y = M$ otrzymujemy

$$\begin{aligned} P(M) &= \frac{1}{r}\left|\frac{1}{M}\left(e^{2\pi i 0} + e^{2\pi i 1} + e^{2\pi i 2} + \ldots\right)\right|^2 \\ &= \frac{1}{r}\left|\frac{1}{M}\underbrace{(1+1+1+\ldots)}_{M\ razy}\right|^2 \\ &= \frac{1}{r}\left|\frac{1}{M}M\right|^2 = \frac{1}{r}\ . \end{aligned} \tag{79}$$

Dla $y = 2M$ otrzymujemy

$$\begin{aligned} P(2M) &= \frac{1}{r}\left|\frac{1}{M}\left(e^{2\pi i 0} + e^{2\pi i 2} + e^{2\pi i 4} + \ldots\right)\right|^2 \\ &= \frac{1}{r}\left|\frac{1}{M}\underbrace{(1+1+1+\ldots)}_{M\ razy}\right|^2 \\ &= \frac{1}{r}\left|\frac{1}{M}M\right|^2 = \frac{1}{r}\,. \end{aligned} \tag{80}$$

Identyczne wyniki otrzymamy dla

$$y = 3M, 4M, \ldots, rM\ ,$$

czyli dla wszystkich wartości zmiennej y współmiernych z M.

Otrzymamy zatem liczby ze zbioru

$$y \in \{M, 2M, 3M, \ldots, rM\}\ . \tag{81}$$

Na postawie serii wyników (81) znajdujemy M, a stąd okres jako

$$r = \frac{2^n}{M}\ . \tag{82}$$

Jeżeli y jest **niewspółmierne** z M, to otrzymamy wyniki rozrzucone przypadkowo wokół liczb ze zbioru (81). Destruktywna interferencja odpowiadających im stanów spowoduje, że znowu otrzymamy liczby ze zbioru (81) (ale tym przypadku z prawdopodobieństwem $P \sim 1$).

Podobny efekt destruktywnej interferencji wystąpi w przypadku, gdy 2^n nie jest podzielne przez r bez reszty.

Można pokazać, że powyższe przewidywania są prawdziwe stosując **algorytm oszacowania fazy**.

7 Algorytm Grovera

Algorytm Grovera rozwiązuje problem poszukiwania elementu w nieuporządkowanej bazie danych. Np. może to być znajdowanie nazwiska osoby w książce telefonicznej, gdy znany jest wyłącznie numer telefony tej osoby.

Jeżeli baza danych zawiera N elementów (haseł), to algorytm klasyczny potrzebuje średnio $N/2$ prób na znalezienie w niej jednego określonego hasła, ponieważ algorytm ten sprawdza hasła jedno po drugim. Kwantowy algorytm poszukiwania, opracowany przez Grovera, potrzebuje na to $\mathcal{O}(\sqrt{N})$ operacji.

Przeanalizujemy działanie algorytmu Grovera.

Bazę danych zapisujemy przy użyciu $N = 2^n$ kubitów $|x\rangle$, gdzie $x = \{0, 1, \ldots, 2^n - 1\}$.

Definiujemy funkcję $f(x)$

$$f(x) = \begin{cases} 0\,, & \text{jeżeli } x \neq y \\ 1\,, & \text{jeżeli } x = y \end{cases} \tag{83}$$

A zatem problem poszukiwania elementu $|y\rangle$ w bazie danych $\{|x\rangle\}$ sprowadzamy do rozwiązania równania

$$f(x) = \delta_{xy}\ . \tag{84}$$

W celu uproszczenia rozważań, zakładamy, że istnieje tylko jedna wartość y spełniająca równanie (84).

Przypominam, że wektor stanu bazy obliczeniowej może być zapisany w zwartej formie jako

$$|x\rangle = |x_{n-1} \ldots x_1 x_0\rangle\ , \tag{85}$$

gdzie $x_{n-1}, \ldots x_,x_0$ przyjmują wartości 0 lub 1, a zatem

$$x = x_{n-1} \ldots x_1 x_0$$

stanowi binarną reprezentację liczby x.

Definiujemy operator wyroczni $\mathcal{O}$ za pomocą wyniku jego działania na wektory bazy obliczeniowej

$$\mathcal{O}|x\rangle = (-1)^{f(x)}|x\rangle\ . \tag{86}$$

Definiujemy operator Grovera jako

$$G = H^{\otimes n} V H^{\otimes n} \mathcal{O}\ , \tag{87}$$

gdzie operator V jest definiowany za pomocą relacji

$$V|x\rangle = -(-1)^{\delta_{x0}}|x\rangle\ , \tag{88}$$

a symbol $H^{\otimes n}$ oznacza n-krotne działanie operatora Hadamarda H (iloczyn tensorowy n operatorów H).

Korzystając z własności operatora jednostkowego w bazie $\{|x\rangle\}$

$$I = \sum_{x=0}^{2^n-1} |x\rangle\langle x| \tag{89}$$

przekształcamy definicję (88) do postaci

$$V|x\rangle = \left(2|0\rangle\langle 0| - I\right)|x\rangle\ . \tag{90}$$

Operator (90) ma postać

$$V = 2|0\rangle\langle 0| - I = |0\rangle\langle 0| - \sum_{x\neq 0}^{2^n-1} |x\rangle\langle x|\ . \tag{91}$$

Powyższe własności operatorów pozwalają nam na przekształcenie operatora Grovera (87) do postaci

$$G = H^{\otimes n} \left(2|0\rangle\langle 0| - I\right) H^{\otimes n} \mathcal{O} \; . \tag{92}$$

Rozwinięcie dowolnego wektora stanu $|\Psi\rangle$ w bazie obliczeniowej $\{|x\rangle\}$, czyli

$$|\Psi\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle \tag{93}$$

można uzyskać za pomocą działania iloczynu tensorowego operatorów Hadamarda $H^{\otimes n}$ na iloczyn tensorowy stanów

$$|0^{\otimes n}\rangle = \underbrace{|0\rangle \otimes |0\rangle \ldots |0\rangle}_{n \; razy} \; .$$

A zatem

$$H^{\otimes n}|0^{\otimes n}\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle = |\Psi\rangle \; . \tag{94}$$

Korzystamy z tego, że dwukrotne działanie operatora Hadamarda jest równoważne działaniu operatora jednostkowego, czyli

$$H^2 = I \; ,$$

a zatem

$$H^{\otimes n} I H^{\otimes n} = H^{\otimes 2n} = I$$

Otrzymujemy stąd

$$H^{\otimes n} \left(2|0\rangle\langle 0| - I\right) H^{\otimes n} = H^{\otimes n} 2|0\rangle\langle 0|) H^{\otimes n} - I = 2|\Psi\rangle\langle\Psi| - I \; . \tag{95}$$

Natomiast operator Grovera przyjmuje postać

$$G = (2|\Psi\rangle\langle\Psi| - I)\mathcal{O} \; . \tag{96}$$

Operator Grovera (96) może być interpretowany jako operator rotacji w płaszczyźnie dwuwymiarowej. Aby to pokazać, rozważmy unormowany wektor stanu

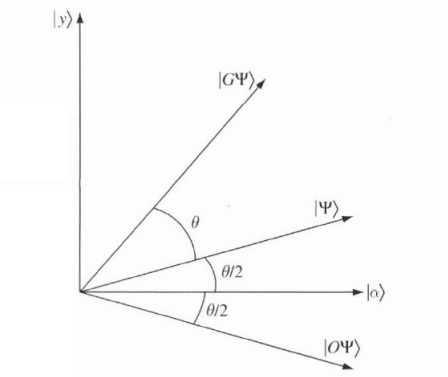
$$|\alpha\rangle = \frac{1}{\sqrt{N-1}} \sum_{x \neq y} |x\rangle \; , \tag{97}$$

gdzie $N = 2^n$. Wektor stanu $|\Psi\rangle$ możemy teraz zapisać w postaci

$$|\Psi\rangle = \sqrt{\frac{N-1}{N}}|\alpha\rangle + \frac{1}{\sqrt{N}}|y\rangle \; . \tag{98}$$

Wzór (98) możemy przepisać jako

$$|\Psi\rangle = \cos\frac{\theta}{2}|\alpha\rangle + \sin\frac{\theta}{2}|y\rangle \; , \tag{99}$$



Rysunek 6: Obrazowe przedstawienie rotacji i odbicia w algorytmie Grovera.

gdzie kąt θ dany jest wzorem

$$\cos\frac{\theta}{2} = \sqrt{1 - \frac{1}{N}} \; . \tag{100}$$

Operator wyroczni (86) działa na kombinacje liniową wektorów $|\alpha\rangle$ i $|y\rangle$ w następujący sposób:

$$\mathcal{O}(a|\alpha\rangle + b|y\rangle) = a|\alpha\rangle - b|y\rangle \; . \tag{101}$$

Na podstawie (101) operator $\mathcal{O}$ można zinterpretować jako **operator odbicia** względem osi wyznaczonej przez $|\alpha\rangle$ w płaszczyźnie Π wyznaczonej przez wektory stanu $|\alpha\rangle$ i $|y\rangle$.

Ponadto operator $\mathcal{R} = 2|\Psi\rangle\langle\Psi| - I$ jest operatorem odbicia w płaszczyźnie Π względem osi wyznaczonej przez wektor $|\Psi\rangle$. Wynika to z następującego rozumowania: jeżeli $\langle\Psi|\Phi\rangle = 0$, to

$$\mathcal{R}(a|\Psi\rangle + b|\Phi\rangle) = (2|\Psi\rangle\langle\Psi| - I)(a|\Psi\rangle - b|\Phi\rangle) = a|\Psi\rangle - b|\Phi\rangle \; . \tag{102}$$

Jednakże iloczyn dwóch operacji odbicia jest **operacją obrotu** w płaszczyźnie Π, czyli

$$G = \mathcal{R}\mathcal{O} \equiv G_{rot} \; . \tag{103}$$

Zgodnie z rysunkiem (6) otrzymujemy obrót o kąt $3\theta/2$, czyli

$$G|\Psi\rangle \equiv G_{rot}|\Psi\rangle = \cos\frac{3\theta}{2}|\alpha\rangle + \sin\frac{2\theta}{2}|y\rangle \; . \tag{104}$$

Wektor stanu $G|\Psi\rangle$ został otrzymany z wektora $|\Psi\rangle$ za pomocą obrotu o kąt θ, co prowadzi do tego, że kąt pomiędzy wektorami stanu $G|\Psi\rangle$ i $|\alpha\rangle$ staje się równy θ. Podobnie wektor stanu $G^2|\Psi\rangle$ otrzymamy z wektora $G|\Psi\rangle$ za pomocą obrotu o kąt θ.

Na podstawie indukcji wnioskujemy, że po k operacjach otrzymamy kąt pomiędzy $G^k|\Psi\rangle$ i $|\alpha\rangle$ równy $(2k+1)\theta/2$. A zatem

$$G^k|\Psi\rangle = \cos\frac{(2k+1)\theta}{2}|\alpha\rangle + \sin\frac{(2k+1)\theta}{2}|y\rangle \ . \tag{105}$$

Ze wzoru (105) wynika, że kolejne obroty coraz bardziej zbliżają wektor stanu $G^k|\Psi\rangle$ do szukanego wektora $|y\rangle$.

Oszacujemy teraz optymalną liczbę operacji $k = k_0$. Żądamy, aby

$$\cos\frac{(2k+1)\theta}{2} = 0 \ . \tag{106}$$

Oznacza to, że

$$0 = \cos k\theta\cos\frac{\theta}{2} - \sin k\theta\sin\frac{\theta}{2} = \sqrt{\frac{N-1}{N}}\cos k\theta - \frac{1}{\sqrt{N}}\sin k\theta \ . \tag{107}$$

Otrzymujemy stąd

$$\tan k\theta = \sqrt{N-1} \tag{108}$$

lub

$$\cos k\theta = \frac{1}{\sqrt{N}} \ , \tag{109}$$

Ze wzoru (109) obliczamy

$$k_0 = \left[\frac{1}{\theta}\cos^{-1}\left(\frac{1}{\sqrt{N}}\right)\right] + 1 \ , \tag{110}$$

gdzie symbol $[\xi]$ oznacza część całkowitą liczby ξ, a $\cos^{-1}$ jest funkcją odwrotną do funkcji cosinus (arc cos).

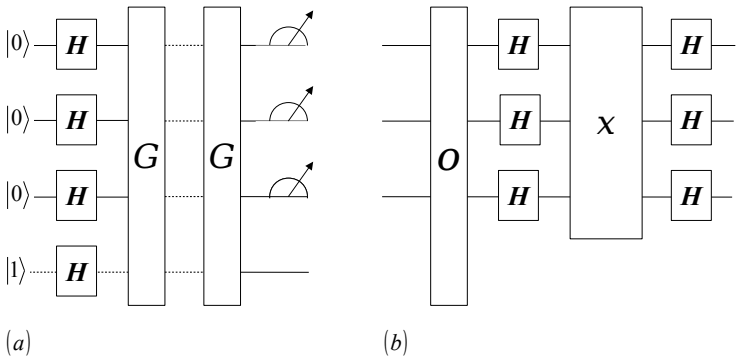
Dla dużych N z porównania (98) i (99) otrzymujemy

$$\theta \simeq \frac{2}{\sqrt{N}} \ , \tag{111}$$

czyli oszacowanie optymalnej liczby operacji ma postać

$$k_0 \simeq \frac{\sqrt{N}}{2}\cos^{-1}\left(\frac{1}{\sqrt{N}}\right) \simeq \frac{\pi\sqrt{N}}{4} \ . \tag{112}$$

Wynika stąd, że algorytm Grovera prowadzi – z prawdopodobieństwem bliskim 1 – do znalezienia pojedynczego elementu w bazie N danych w $\mathcal{O}(\sqrt{N})$ krokach.



(a) Logic circuits of the Grover algorithm for $n = 3$. (b) The circuits of G. The action of the oracle O is $O|x\rangle = (-1)^{f(x)}|x\rangle$ and that of the box X is $X|x\rangle = -(-1)^{\delta_{x0}}|x\rangle$

Rysunek 7: Schemat obwodu kwantowego do implementacji algorytmu Grovera.