Algorytm faktoryzacji Shora

Maciej Dudek, Michał Kaczmarczyk

Politechnika Krakowska im. Tadeusza Kosciuszki, wydział Fizyki, Matematyki i Informatyki

15 czerwca 2018

- Kwantowy algorytm Shora jest algorytmem kwantowym pozwalającym na rozłożenie liczby naturalnej N na czynniki pierwsze w czasie $O((\log N)^3)$ wykorzystując przy tym pamięć równą $O(\log N)$.
- Algorytm ten stanowi problem dla używanego w internecie kryptosystemu RSA który wykorzystuje duże liczby pierwsze.
- Opublikowany przez Petera Shora w 1994 r.
- Jest to algorytm probabilistyczny lecz dzięki szybkiemu działaniu możemy go powtarzać w optymalnym czasie

- Na początku dostajemy liczbę naturalna N
- Musimy znaleźć takie p które dzieli liczbę N bez reszty znajdujące się w zakresie $1 < p < N$

Algorytm możemy podzielić na dwie części:

1. Sprawdzenie problemu faktoryzacji do problemu znalezienia rzędu elementu w grupie (realizacja na komputerze klasycznym)
2. Znalezienie rzędu elementu za pomocą algorytmu kwantowego

1. Losujemy liczbę $a < N$
2. Obliczany NWD(a, N) – Największy wspólny dzielnik, np. Euklidesem
3. Jeśli NWD$(a, N) \neq 1$, to znaleźliśmy nietrywialny dzielnik N i kończymy część klasyczną
4. W innym wypadku używamy procedury znajdującej r, które jest okresem funkcji: $f(x) = a \times (\mathrm{mod} N)$, czyli znajdujemy najmniejsze naturalne r, takie, że $f(f + r) = f(x)$
5. Jeśli r jest nieparzyste, wracamy do pierwszego kroku
6. Jeśli $a^{\frac{r}{2}} \equiv -1(\mathrm{mod} N)$, wracamy do pierwszego kroku
7. Dzielnikiem N jest NWD$(a^{\frac{r}{2}} \pm 1, N)$.

Koniec części klasycznej

1. Przygotowujemy dwa rejestry kwantowe: wejściowy i wyjściowy, każdy z $\log_2 N$ kubitów i inicjujemy je na stan:

$$N^{-\frac{1}{2}} \sum_x |x\rangle|0\rangle$$

$$\text{dla } x \in (0, N-1)$$

2. Następnie konstruujemy układ realizujący funkcję f(x) w postaci kwantowej i aplikujemy do powyższego stanu otrzymując:

$$N^{-\frac{1}{2}} \sum_x |x\rangle|f(x)\rangle$$

3. Aplikujemy odwróconą kwantową transformatę Fouriera do rejestru wejściowego.

$$U_{QFT}|x\rangle = N^{-\frac{1}{2}} \sum_{x} e^{-\frac{2\pi i x y}{N}} |y\rangle$$

Wynikiem tej operacji jest stan:

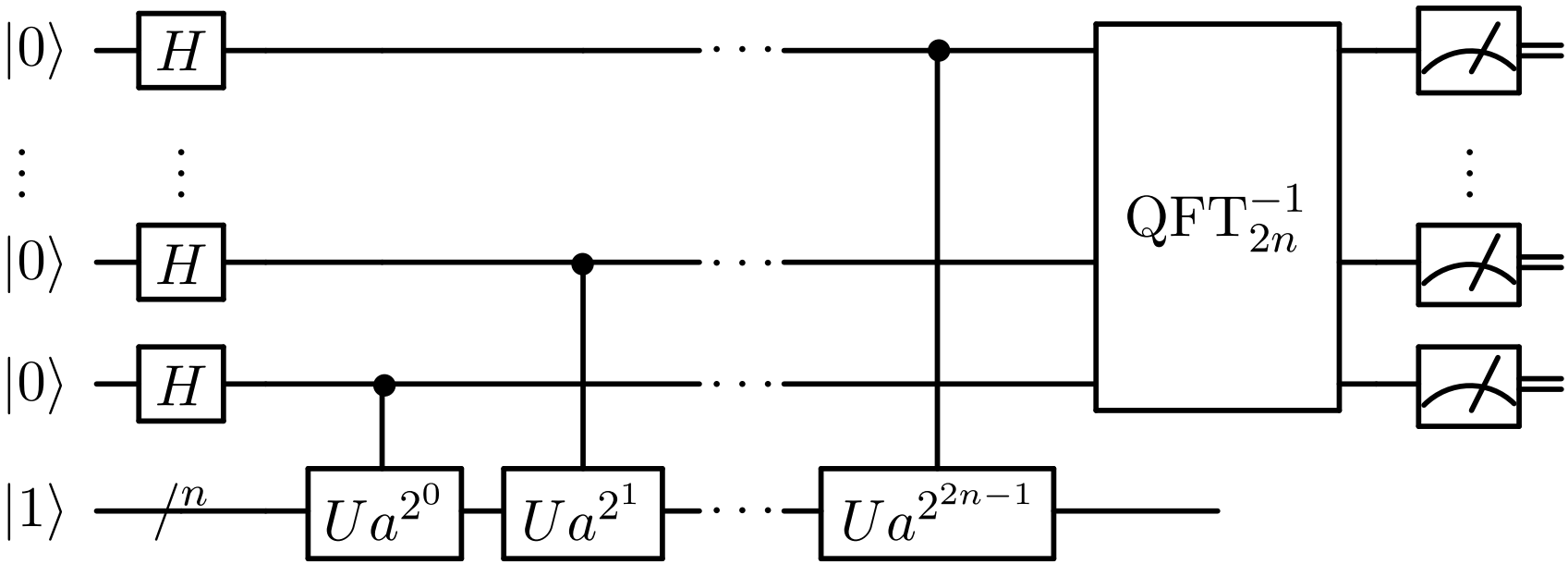
$$N^{-1} \sum_{x} \sum_{y} e^{-\frac{2\pi i x y}{N}} |y\rangle |f(x)\rangle$$

4. Teraz dokonujemy pomiaru otrzymując y w rejestrze wejściowym i $f(x_0)$ w rejestrze wyjściowym. Z powodu, że f jest okresowa, prawdopodobieństwo na otrzymanie par $y, f(x_0)$ wynosi:

$$\left| N^{-1} \sum_{x:f(x)=f(x_0)} e^{-\frac{2\pi i x y}{N}} \right|^2 = N^{-2} \left| \sum_b e^{-\frac{2\pi i (x_0 + rb) y}{N}} \right|^2$$

Prawdopodobieństwo to jest tym większe im wartość $\frac{yr}{N}$ jest bliższa liczbie całkowitej

5. Przekształcamy y/N w nieskracalny ułamek i bierzemy jego mianownik r' jako kandydata na r.
6. Sprawdzamy czy $f(x) = f(x + r')$. Jeśli tak kończymy algorytm.
7. Jeśli nie to sprawdzamy innych kandydatów na r przy użyciu wartości bliskiej y albo wielokrotności liczby r'. Jeśli któryś pasuje to algorytm jest zakończony.
8. Jeśli nie znajdziemy dobrego r to wracamy do punktu pierwszego



Rysunek: Obwód kwantowy

Faktoryzacja liczby 21

1. Losujemy liczbę x:
   - Jeśli nasze a nie jest względnie pierwsze z N np. $x = 6$ $\mathrm{NWD}(x, N) = \mathrm{NWD}(6, 21) = 3 \rightarrow 21/3 = 7$, skończyliśmy część klasyczną
   - Jeśli jest odwrotnie i np. $x = 11$ to $\mathrm{NWD}(11, 21) =$, przechodzimy do kolejnego kroku

2. Znajdujemy okres funkcji N.

Szukamy najmniejszej potęgi liczby 2, q z zakresu
$N^2 < q < 2N^2$
$N^2 = 441 < q = 2 < 2N^2 = 882 \rightarrow q = 512 = 2^9$
Dostajemy stan początkowy składający się z 2 rejestrów o długości l:

$$|\Phi\rangle = |0\rangle_{r_1}|0\rangle_{r_2} = |0\rangle^{\otimes 2^l}$$

3. Inicjujemy pierwszy rejestr z superpozycją wszystkich stanów a

$$|\Phi_0\rangle = \frac{1}{\sqrt{512}} \sum_{a=0}^{511} |a\rangle|0\rangle$$

Odpowiada to $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ na wszystkich bitach

4. Inicjujemy drugi rejestr z superpozycją wszystkich stanów x^a

$$|\Phi_1\rangle = \frac{1}{\sqrt{512}} \sum_{a=0}^{511} |a\rangle |11^a (\mathrm{mod} 21)\rangle$$
$$= \frac{1}{\sqrt{512}} (|0\rangle|1\rangle + |1\rangle|11\rangle + |2\rangle|16\rangle + |3\rangle|8\rangle + \ldots)$$

a	0	1	2	3	4	5	6	7	8	9	10	...
$11^a (\mathrm{mod} 21)$	1	11	16	8	4	2	1	11	16	8	4	...

5. Używamy QFT na pierwszy rejestr

$$|\widetilde{\Phi}\rangle = \frac{1}{512}\sum_{a=0}^{511}\sum_{c=0}^{511} e^{\frac{2\pi iac}{512}}|c\rangle|11^a(\mathrm{mod}21)\rangle$$

6. Dokonujemy pomiaru prawdopodobieństwa stanu

$$|c, x^k(\mathrm{mod}n)\rangle, \text{ e.g. } k = 2 \rightarrow |c, 16\rangle$$

co daje nam:

$$p(c) = \left|\frac{1}{512}\sum_{a:11^a \atop \mathrm{mod}21=16} e^{\frac{2\pi iac}{512}}\right|^2 = \left|\frac{1}{512}\sum_{b} e^{\frac{2\pi i(6b+2)1}{512}}\right|^2$$

Największą wartością jest $c = \frac{512}{6}\cdot d, d \in Z$

7 Szukamy okresu r.
Zakładamy ze dostaliśmy 427:

$$\left|\frac{c}{q} - \frac{d}{r}\right| = \left|\frac{427}{512} - \frac{d}{r}\right| \leq \frac{1}{\mathrm{mod}24}$$

Kontynuujemy rozszerzenia:

$$\frac{c}{q} = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\cdots}}}, \qquad d_0 = a_0, \qquad d_1 = 1 + a_0 a_1, \qquad d_n = a_n d_{n-1} + d_{n-2}$$

$$r_0 = 1, \qquad r_1 = a_1, \qquad r_n = a_n r_{n-1} + r_{n-2}$$

$$\frac{427}{512} = 0 + \frac{1}{1 + \frac{1}{5 + \frac{1}{42 + \frac{1}{2}}}}, \qquad d_0 = 0, \quad d_1 = 1, \quad d_2 = 3, \quad d_3 = 5$$

$$r_0 = 1, \quad r_1 = 1, \quad r_2 = 6, \quad r_3 = 512$$

8. Jak widać $d_0/r_0 = 0$ podobnie $d_1/r_1 = 1$ nie interesują nas i próbujemy, $d_2/r_2 = 5/6 \rightarrow r = 6$
9. Sprawdzamy czy r jest parzysty. $+$
   Sprawdzamy czy $x^{r/2} \mathrm{mod} N \neq -1$. $+$
   Określamy czynniki:

$$x^{\frac{r}{2}} \mathrm{mod} n - 1 = 11^3 \quad \mathrm{mod} 21 - 1 = 7$$

$$x^{\frac{r}{2}} \mathrm{mod} n + 1 = 11^3 \quad \mathrm{mod} 21 + 1 = 9$$

Czynnikami są $\mathrm{NWD}(7, 21) = 7$ i $\mathrm{NWD}(9, 21) = 3$

- Algorytm Shora jest bardzo istotny w kryptografii jako, że może faktoryzować duże liczby dużo szybciej niż klasyczne algorytmy
- Znajduje swoje zastosowanie w symulacjach kwantowych
- Głównym problemem algorytmu jest to, że gdy okres serii okaże się nieparzysty musimy powtarzać algorytm dla nowego x

Dziękujemy za uwagę!