Advanced cryptology

Quantum cryptography

prof. Ing. Róbert Lórencz, CSc.

České vysoké učení technické v Praze, Fakulta informačních technologií
Katedra informační bezpečnosti

Faktorization, Shor's algorithm 1 – Introduction

- Shor's algorithm for integer factorization uses quantum paralelism.
- On a quantum computer, it runs in $\mathcal{O}(L^2 \log\ L\ \log\log\ L)$, where L is the bit length of the number to be factored. The upper bound for the run time is a polynomial.
- The algorithm does not search for factors directly, but transforms factorization into searching for a **period of a certain periodic function.**
- For the factored number n, create a periodic function

$$f_{y,n}(a) = y^a \bmod\ n,$$

where y is a random integer coprime to n.

Faktorization, Shor's algorithm 2 – Introduction

- This function is interesting for its periodicity. Its period modulo n is usually denoted r. Since every r-th value is equal ($f_{y,n}(a) = f_{y,n}(a + r)$), it holds

$$y^r \equiv 1 \pmod{n}.$$

- After some adjusting,

$$(y^{r/2} - 1)(y^{r/2} + 1) \equiv 0 \pmod{n},$$

- where r is the even period (if odd, choose a different y).
- Left hand side product is divisible by n. Therefore, if it is not trivially $y^{r/2} \equiv \pm 1 \pmod{n}$, then one of the left hand side factors must have a common divisor with n.
- This way, the task is transformed to finding the greatest common divisor (gcd) of $(y^{r/2} - 1, n)$ and $(y^{r/2} + 1, n)$.
  EA solves this problem efficiently on a classical computer.

Example: Factorize $n = 21$ into a product of its prime factors. Thus, we choose $1 < y < 21$ so that $\gcd(y, 21) = 1$.

- Then $y \in \{2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20\}$.
- From it, randomly choose $y = 10$.
- Now we want to find the period of $f_{y,n}(a) = f_{10,21}(a) = 10^a \bmod 21$.
- The function values for integer $a = 1, 2, \ldots$ are $10, 16, 13, 4, 19, 1, 10, 16, \ldots$.
- This function has even period $r = 6$ and does not return trivial factors.
- Since $y^{r/2} = 1000$, we want to verify if $1000 \stackrel{?}{\equiv} \pm 1 \pmod{21}$. It is not, since $999 \nmid 21$ and $1001 \nmid 21$. If it would, we would have to choose a different y.
- In conclusion, we find the factors using $\gcd(1001, 21) = 7$ and $\gcd(999, 21) = 3$.

- On the other hand, if $y = 20$, the algorithm fails, because the period $r = 2$ (20,1,20,1,...). We want to know if $20 \stackrel{?}{\equiv} \pm 1 \pmod{21}$ and we see that $21 \mid 21$.
- A problem remains – **how to efficiently compute the period** r of a given function.
- This problem is not classically solvable in polynomial time. However, Shor showed that on a quantum computer, the period can be efficiently found using quantum parallelism.

Algorithm

- Prepare a quantum register consisiting or 2 parts called $R1$ and $R2$, and whose state we will denote $|r1, r2\rangle$.
- Step 1: Choose a random y coprime to n and choose q, for which $2n^2 \leq q \leq 3n^2$.