## § 8.1 RSA

Designers: Ron Rivest, Adi Shamir, and Leonard Adleman, 1977.

Basic assumption. Factorization of $N = pq$ for two prime numbers $p$ and $q$ are hard to do.

Public key crypto-system. Bob (the bank, VISA card co.) can announce a public key for customers (Alice) to encrypt their message and send it to Bob via a public channel, and Bob can easily decrypt the message.

**Example** Factor the following numbers into $N = pq$.

45878443254366745

7536576836238936804738907362515346578697687343

753657683628743673389368047389675407362518902115346578697687

**Mathematical Background**

- **Euclidean Algorithm** Let $a, b$ be positive integer. There are unique pair of integers $(q, r)$ such that $b = aq + r$.

- **Fermet's Little Theorem** If $p$ is an odd prime, and $a \in \mathbb{Z}$ is not a multiple of $p$, then

$$a^{p-1} \equiv 1 \ (\bmod p).$$

$p \mid (a^{p-1} - 1)$

- Use the notation $\mathbb{Z}_p = \{[0], \ldots, [p-1]\}$. Then $[j]^p = [1]$ whenever $[j] \neq [0]$.

  See the nice proof in the textbook.

- Let $N = pq$ for two odd primes $p, q$. Then for every $[e]$ in the set

$$\mathbb{Z}^*_{pq} = \{[j] : j \text{ is not a multiple of } p \text{ or } q\},$$

  there is a unique $[d] \in \mathbb{Z}^*_{pq}$ such that $[e][d] = [1]$.

  As a result, for any $m \in \{0, \ldots, N-1\}$, if $m^e$ is given, then $[m]^{(ed)} = [m]$.

$\mathbb{Z}_{2\cdot 3} = \mathbb{Z}_6$

$0 \le m < N = \{[0], \ldots [5]\}$

$[m]$ $N = pq$

$pq \mid (e \cdot d - 1)$

$\{[1], [5]\}$

- So, Bob can announce $e$ in public. If Alice wants to send $m$, she can send $c = m^e$. Bob can then recover $m$ by computing $[c]^d = [m]^{(ed)} = [m]$.

$= [1]$

$[1][1]$

$[5][5]$

$= [25] = [1]$

**RSA Scheme**

**Step 1** Bob: Let $N = pq$, and let $e < N$ be relatively prime to $(p-1)(q-1)$. Here $e$ is known as the exponent, and release $N$ and $e$. Then compute the modular inverse $d$ of $e$ and keeps $d$ secret.

**Step 2** Alice: To send Bob a message represented as a number $m$, encode the message $m$ by $m^e$ and send it through an open/public channel.

**Step 3** Bob: Decode the message by applying $(m^e)^d = m \pmod{N}$.

**Example** 1. Let $(p, q) = (61, 53)$ and $N = 3233$.

2. The groups of units has $(p-1)(q-1) = 780$ elements.

3. For instance $e = 17$ is a unit, and $d = 413$ satisfies
   $ed \equiv 1 \pmod{N}$.

4. Public key $(N, e) = (3233, 17)$.

5. Alice sends a number (message) $m$ as
   $c(m) = m^e \pmod{3233}$ with $c(m) < 3233$.

6. Bob decrypts $c(m)$ as $m = c(m)^d \pmod{3233}$.

   For instance if $m = 65$, then $c = 65^{17} = 2790((\text{mod } 3233)$.

   Then Bob computes $2790^{413} = 65((\text{mod } 3233)$.

## § 8.2 Factorization Algorithm

**Step 1** Let $N$ be given. Take a random $m < N$ and compute $\gcd(m, N) = g$ by the Euclidean Algorithm.
If $g > 1$, we are extremely lucky! If not, go to Step 2.

**Step 2** Define $f_N : \mathbb{N} \to \mathbb{N}$ by $a = m^a \pmod N$.
Find the smallest $P$ such that $m^P = 1 \pmod N$.
(That is, finding the order/period of $m$ in $U_N^*$.)
(This is the quantum part!)

**Step 3** If $P$ is odd, it cannot be used. Go back to Step 1.
Else, go to Step 4.

**Step 4** If $P$ is even, then
$(m^{P/2} - 1)(m^{P/2} + 1) = m^P - 1 = 0 \pmod N$.
If $m^{P/2} + 1 = 0 \pmod N$, then $\gcd(m^{P/2} - 1, N) = 1$;
go back to Step 1.
If $m^{P/2} + 1 \neq 0 \pmod N$, then
$m^{P/2} - 1$ has a prime factor of $N$.
[Note that $m^{P/2} \neq 1 \pmod N$ as $P$ is the order of $m$.]
Proceed to Step 5.

**Step 5** Compute $d = \gcd(m^{P/2} - 1, N)$ to get $p$ or $q$.

**Example** Let $N = 799$.
Step 1. Choose $m = 7$.
Step 2. Then (by quantum computer or conventional computer) that $P = 368$ is the smallest positive number such that $7^P = 1$ ((mod 799)).
Step 3. Set $P/2 = 184$. Then
$(7^{184} - 1)(7^{184} + 1) = 0 \pmod{799}$.
Step 4. Now, $\gcd(7^{184} + 1, 799) = 17 \neq 1$.
So, we are good and done, namely, $799 = 17 \cdot 47$.
[In fact, $\gcd(7^{184} - 1, 799) = 47$.]

$f(x)$

$\gcd(a, r_1) = \gcd(r_2, r_1)$

$m^P \equiv 1$

$799 \mid (7^9 \equiv 1)$ $\quad 7^9 \equiv 1 \pmod{799}$

$7^{368} \equiv 1 \pmod{799}$

$\gcd(S, N) \neq 1$

## § 8.3 - 8.5 Shor's Algorithm

designed by Peter Shor (1994).

Complexity: The time taken is polynomial in $\log N$, which is the size of the input). Specifically it takes quantum gates of order $O((logN)^2(loglogN)(logloglogN))$ using fast multiplication.

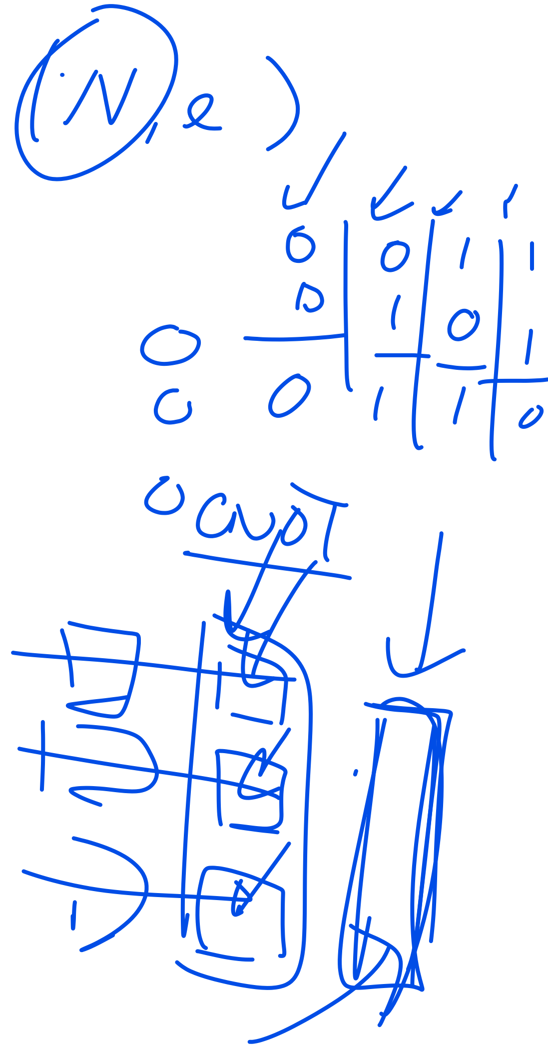
This is almost exponentially faster than the most efficient known classical factoring algorithm, the general number field sieve: $\left(e^{1.9(\log N)^{1/3}(\log\log N)^{2/3}}\right)$.

- In 2001, Shor's algorithm was demonstrated by a group at IBM, who factored 15 into $3 \times 5$, using an NMR implementation of a quantum computer with 7 qubits.

- After IBM's implementation, two independent groups implemented Shor's algorithm using photonic qubits, emphasizing that multi-qubit entanglement was observed when running the Shor's algorithm circuits.

- In 2012, the factorization of 15 was performed with solid-state qubits. Also in 2012, the factorization of 21 was achieved, setting the record for the largest number factored with Shor's algorithm.

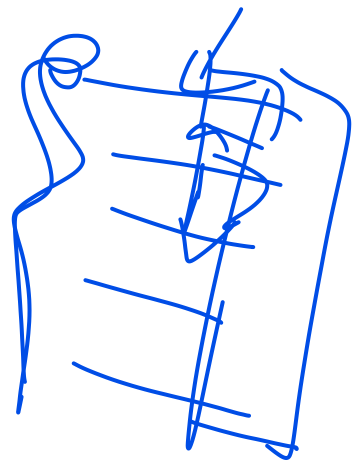
- In 2019 an attempt was made to factor the number 35 using Shor's algorithm on an IBM Q System One, but the algorithm failed due to cumulating errors.



- In April 2012, the factorization of $143 (= 11 \times 13)$ was achieved, although this used **adiabatic quantum computation** rather than Shor's algorithm.

- In November 2014, it was discovered that this 2012 adiabatic quantum computation had also factored larger numbers, the largest being $56153 = 233 \times 241$.

D wave



- Using additional mathematical idea, Gröbner basis, researchers managed to factor $223357 = 401 \times 557$ in 2017

- In 2018, in the paper "Quantum Annealing for Prime Factorization", researchers showed how to factor 15, 143, $59989 = 239 \times 251$, and $376289 = 571 \times 659$ using 4, 12, 59, and 94 logical qubits.

Let $N = pq$, and choose $n$ so that $N^2 \leq 2^n < 2N^2$ so that $S_n = \{0, \ldots, Q-1\}$ with $Q = 2^n$. Define $f : S_n \to \mathbb{Z}/N\mathbb{Z}$ by $f(a) = m^a \pmod N$. Apply the following.

**Step 2.0** Set up $|\psi_0\rangle = |0\rangle|0\rangle$ in $S_n \otimes S_n$.

**Step 2.1** Apply QFT to the first register to get

$$|\psi_1\rangle = T|0\rangle \otimes |0\rangle.$$

**Step 2.2** Apply $f$ using the unitary $U_f$ so that

$$U_f|\psi_1\rangle = |\psi_1\rangle = \frac{1}{\sqrt{Q}}\sum_{x=0}^{Q-1}|x\rangle|f(x)\rangle.$$

**Step 2.3** Apply QFT to the first register to get

$$\Upsilon(y) = \sum_{x=0}^{Q-1} w_n^{-xy}|f(x)\rangle$$

and

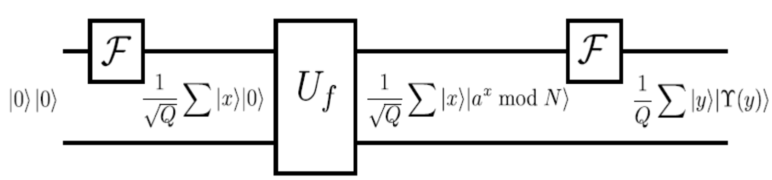
$$|\psi_3\rangle = \frac{1}{Q}\sum_{x=0}^{Q-1} T|x\rangle|f(x)\rangle = \frac{1}{Q}\sum_{y=0}^{Q-1}\|\Upsilon(y)\|\,|y\rangle\frac{|\Upsilon(y)\rangle}{\|\Upsilon(y)\|}.$$

**Step 2.4** Measure the first register. The probability of $y \in S_n$ will be

$$\mathrm{Prob}(y) = Q^{-2}\|\Upsilon(y)\|^2 = Q^{-2}|\sum_b w^{bPy}|^2,$$

and the state collapses to $|y\rangle(\|\Upsilon(y)\|/Q)$, where $w = e^{2\pi i/Q}$.

**Step 2.5** Find the order $P$ from the measurement outcome.




Quantum circuit to find the order of $f(x) = m^x \bmod N$.

**Remarks**

1. If $f$ is periodic, $f(x) = f(x+P)$ we see that $\|\Upsilon(y)\|^2/Q^2$ is larger.

2. In general, if $(w^{Py})$ is near $\pm 1$, i.e., $yP/Q$ is close to an integer $c$.

3. By the theory of of continued fractions of rational number, we need to find $d/s$ such that

$$|d/s - y/Q| \leq 1/(2Q), \qquad \gcd(d,s) = 1, \; s < N.$$

4. So, if $f(x) = f(x+s)$ then $s = P$. If not, try $m^x$ to get other fraction $d'/s'$ to approximate $y/Q$.

## §8.4 Probability Distribution (Details)

**Proposition 8.1** Let $Q = 2^n = Pq + r$ with $0 \le r < P$, and let $Q_0 = Pq$.

(a) If $Py$ is not a multiple of $Q$, then

$$\boxed{\operatorname{Prob}(y) = \frac{1}{Q^2}\|\Upsilon(y)\|^2}$$

$$= \frac{r\sin^2\left(\frac{\pi P y}{Q}\left(\frac{Q_0}{P}+1\right)\right) + (P-r)\sin^2\left(\frac{\pi P y}{Q}\cdot\frac{Q_0}{P}\right)}{Q^2\sin^2\left(\frac{\pi P y}{Q}\right)}.$$

(b) If $Py$ is a multiple of $Q$, then

$$\operatorname{Prob}(y) = \frac{1}{Q^2}\|\Upsilon(y)\|^2 = \frac{r(Q_0+P)^2 + (P-r)Q_0^2}{Q^2P^2}.$$

*Proof.* See pp. 145-146. $\square$

**Corollary** If $Q/P \in \mathbb{N}$, then

$$\operatorname{Prob}(y) = \begin{cases} 0 & \text{if } \; Py \ne 0 \;(\bmod Q), \\ 1/P & \text{if } \; Py = 0 \;(\bmod Q). \end{cases}$$

**Remark** Only those $y \in \{0, \ldots, Q-1\}$ satisfying $y = Pr$ has high $\operatorname{Prob}(y)$. (cf. Exercise 6.3.)

**Limitation** One may do a number of measurements to determine $P$ by finding the minimum distance between those $|y\rangle$ with high probability. But this is impractical if $N$ is large.

## §8.5 Continued Fractions and Order Finding (Details)

We use the continued fractions representation of a rational number $x = [a_0, \ldots, a_q]$.

$$x = a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\cdots + \frac{1}{a_q}}}}.$$

Example: $\frac{17}{47} = [0, 2, 1, 3, 4]$.

Think about the gcd calculation of (17/47).

**An algorithm for finding the order $P$ of $m^x \pmod{N}$.**

1. Find the continued fraction expansion $[a_0, a_1, \ldots a_M]$ of $y/Q$.

   We always have $a_0 = 0$ since $y/Q < 1$.

2. Let $p_0 = a_0$ and $q_0 = 1$.

3. Let $p_1 = a_1 p_0 + 1$ and $q_1 = a_1 q_0$.

4. Let $p_i = a_i p_{i-1} + p_{i-2}$ and $q_i = a_i q_{i-1} + q_{i-2}$ for $2 \leq i \leq M$.

   We obtain the sequence $(p_0, q_0), (p_1, q_1), \ldots, (p_M, q_M)$.

   It can be shown that $p_j/q_j$ is the $j$th convergent of $y/Q$.

5. Find the smallest (unique) $k$ with $0 \leq k \leq M$ such that $|p_k/q_k - y/Q| < 1/(2Q)$.

6. The order is found as $P = q_k$.

**Example 8.2** Let $N = 799, Q = 2^{20} = 1048576$ and $m = 7$. The error bound is $1/(2Q) = 4.76837 * 10^{-7}$. Suppose we obtain $y = 8548$ as a measurement outcome of the first register. We expect that $y/Q$ is an approximation of $n/P$ for some $n \in \mathbb{N}$.

1. The continued fraction expansion of 8548/1048576 is

   $[0, 122, 1, 2, 44, 5, 3]$ and $M = 6$.

2. Let $p_0 = a_0 = 0$ and $q_0 = 1$.

3. We obtain $p_1 = a_1 p_0 + 1 = 122 * 0 + 1 = 1$ and $q_1 = a_1 q_0 = 122 * 1 = 122$. We have

   $|p_1/q_1 - y/Q = |1/122 - 8548/1048576| = 4.47133 * 10^{-5} > 1/(2Q).$

4. Let $p_2 = a_2 p_1 + p_0 = 1$, $q_2 = a_2 q_1 + q_0 = 123$ and

   $|p_2/q_2 - y/Q| = |1/123 - 8548/1048576| = 2.1 * 10^{-5} > 1/(2Q).$

5. Ler $p_3 = a_3 p_2 + p_1 = 3$, $q_3 = a_3 q_2 + q_1 = 368$ and

   $|p_3/q_3 - y/Q| = |3/368 - 8548/1048576| = 1.65856 * 10^{-7} \leq 1/(2Q).$

   We have obtained $k = 3$.

6. The order is found to be $P = q_3 = 368$.

**Proposition 8.2** If $y \in \{0, \ldots, Q-1\}$ satisfies $|d/P - y/Q| \leq 1/2Q$ with $\gcd(P, d) = 1\}$, then the algorithm will determine $P$.

## §8.6 Modular Exponential Function

To that the Shor's algorithm is polynomial time, one needs to implement the computation of $f(x) = m^x$ efficiently using quantum gates. This can be done as shown in Section 8.6. The implementation is done in the following steps.

1. Adder, which outputs $a + b$ given non-negative integers $a$ and $b$.

2. Modular adder, which outputs $a + b \pmod N$.

3. Modular multiplexer, which outputs $ab \pmod N$.

4. Modular exponential function, which outputs $m^x \pmod N$.