Randomized Algorithms, Lecture No. 3
Russell Impagliazzo, Lecturer

Michel Abdalla, scribe

February 3, 1998

Technique: algebraic counting arguments.

Fact 1 The number of roots of a polynomial, $\mod$ a prime, is at most equal to its degree.

Fact 2 [Fermat's little theorem] If p is a prime, then, for every $g \in Z_p^\star$ ($g \not\equiv 0 \mod p$), $g^{p-1} \equiv 1 \mod p$.

Both ideas can be used to compute square roots $\mod$ a prime.

Motivation: Two commonly-used encryption schemes, RSA and Rabin's scheme, are based on factorization.

RSA uses $\mathcal{E}(x) = x^e \mod N$ as an encryption function, where N is equal to the product of two primes p and q, and e is relatively prime to $p-1$ and $q-1$. Rabin's scheme, instead, uses $\mathcal{E}(x) = x^2 \mod N$ as an encryption function.

One important point here is how these encryption mechanisms are related to the factorization problem. That is, while it is known we can break both schemes if we know how to break factorization, the other way around is only known for Rabin's scheme.

An interesting question, however, is how to decrypt a message knowing the factorization. If we can take "square roots" modulo both p and q, we can also take them mod $N = pq$, and hence decrypt Rabin's function.

Idea: if $p \mod 4 \equiv 3$, then $\left(x^{\frac{p+1}{4}}\right)^2 = x^{\frac{p+1}{2}} = x^{\frac{p-1}{2}} \cdot x \equiv \pm x$. Therefore, if $x = y^2$ for some y, then $x^{\frac{p-1}{2}} = \left(y^2\right)^{\frac{p-1}{2}} = y^{p-1} \equiv 1 \mod p$. Then

$x^{\frac{1}{2}} = x^{\frac{p+1}{4}} \mod p$, if such a square root exists.

However, if $p \mod 4 \equiv 1$, the idea presented above cannot be used, because $\frac{p+1}{4}$ is not an integer, and so the last argument doesn't work.

The following is a sketch of a probabilistic algorithm to compute the square root of a number $\mod p$ when $p \mod 4 \equiv 1$. The idea is that, although we don't know the value of $\sqrt{x}$, we can use symbolic multiplication to get around it.

Definition 1 (Symbolic Multiplication mod p)

$$(a + b\sqrt{x}) \times (c + d\sqrt{x}) = (ac + bdx) + (ad + bc)\sqrt{x}$$

Definition 2 $(r + \sqrt{x})^d = s_d(r) + t_d(r)\sqrt{x}$

Algorithm

1. Pick r at random
2. Symbolically compute $(r + \sqrt{x})^{\frac{p-1}{2}} = s + t\sqrt{x}$ using repeated squaring.
3. If $t = 0$, goto step 1.
4. If $t \neq 0$, then $s + t\sqrt{x}$ should be either 1 or $-1 \mod p$.
   - To check if $s + t\sqrt{x} = 1$, try $\sqrt{x} = (1 - s)t^{-1}$
   - To check if $s + t\sqrt{x} = -1$, then try $\sqrt{x} = (-1 - s)t^{-1}$

Fact 3 Either $t_{\frac{p-1}{2}}(r) \neq 0$ for at least half of the residues r or $t_{\frac{p-1}{2}}(r) = 0$ for every residue r.

Claim 4 *s_d and t_d are polynomials of degree at most d*

Proof: The proof is by induction. First, the base case where $d = 1$ is trivial and can be easily verified. In the inductive step, we will consider two cases depending on whether d is even or odd.

If d is even, then

$$\begin{aligned}
(r + \sqrt{x})^d &= \left((r + \sqrt{x})^{\frac{d}{2}}\right)^2 \\
&= \left(s_{\frac{d}{2}}(r) + t_{\frac{d}{2}}(r)\sqrt{x}\right)^2 \\
&= s_{\frac{d}{2}}(r)^2 + t_{\frac{d}{2}}(r)^2 x + 2 s_{\frac{d}{2}}(r) t_{\frac{d}{2}}(r)\sqrt{x} \\
&= s_d(r) + t_d(r)\sqrt{x}
\end{aligned}$$

Therefore, $s_d(r) = s_{\frac{d}{2}}(r)^2 + t_{\frac{d}{2}}(r)^2 x$ and $t_d(r) = 2s_{\frac{d}{2}}(r)t_{\frac{d}{2}}(r)\sqrt{x}$. Since the $s_{\frac{d}{2}}$ and $t_{\frac{d}{2}}$ are polynomials of degree $\frac{d}{2}$ and s_d and t_d are quardratic expressions in these polynomials, s_d and t_d are degree d polynomials.

If d is odd, then

$$\begin{aligned}(r+\sqrt{x})^d &= \left(s_{d-1}(r)+t_{d-1}(r)\sqrt{x}\right)(r+\sqrt{x}) \\ &= (rs_{d-1}(r)+xt_{d-1}(r)) + (s_{d-1}(r)+rt_{d-1}(r))\sqrt{x}\end{aligned}$$

Therefore, $s_d(r) = rs_{d-1}(r) + xt_{d-1}(r)$ and $t_d(r) = s_{d-1}(r) + rt_{d-1}(r)$. As before, since s_{d-1} and t_{d-1} are degree $d-1$, this means that s_d and t_d are degree d. ■

Claim 5 *If $t_{p-1/2}(r) \neq 0 \mod p$ for any r, then $t_{p-1/2}(r) \neq 0$ for at least half the residues r.*

Proof: Since $t_{p-1/2}$ is a polynomial of degree $p-1/2$, if it is not always 0, it has at most $p-1/2$ roots. That's less than half of the p residues $\mod p$. ■

Claim 6 ***Let** $y^2 = x$. **Then for any** r, $(r+y)^d = s_d(r) + t_d(r)y$**,and** $(r-y)^d = s_d(r) - t_d(r)y$.*

Proof: Since $y^2 = x$, $(-y)^2 = x$. Hence, the equation $(r+\sqrt(x))^d = s_d(r)+t_d(r)\sqrt(x)$ is valid replacing $\sqrt(x) = y$ and replacing $\sqrt(x) = -y$. ■.

Claim 7 ***If** $x \neq 0 \mod p$ **and** $y^2 = x$, **then** $t_d(y) \neq 0 \mod p$.*

Proof: Assume $t_d(y) = 0$. Then $0 = (y-y)^d = s_d(y) - t_d(y)y = s_d(y)$. So $(y+y)^d = 2^d y^d = s_d(y) + t_d(y)y = 0 + 0y = 0$. So $y = 0$ mod p, and $x = y^2 = 0$ mod p, a contradiction. ■.

So in particular, for $d = \frac{p-1}{2}$, we have that $t_{\frac{p-1}{2}}(r) \neq 0$ for at least one value of r, provided x has a square root. So $t_{\frac{p-1}{2}}(r) \neq 0$ with probability greater than 1/2 for a random r. Thus, our algorithm expects to run only twice before halting.