

Algorithms for Public Key Cryptography

Computing Square Roots Modulo a Prime

We have already seen how to compute square roots modulo primes of the form $p = 4k + 3$:

Let α be a quadratic residue modulo p . Then

$$\beta \equiv \alpha^{\frac{p+1}{4}} \equiv \alpha^{k+1} \pmod{p}$$

is a square root of α :

$$\beta^2 \equiv \alpha^{\frac{p+1}{2}} \equiv \alpha \alpha^{\frac{p-1}{2}} \equiv \alpha 1 \equiv \alpha \pmod{p}.$$

Note that $-\beta$ is also a square root of α .

Example: Compute the square root of $\alpha = 3$ modulo $p = 11$.

$$\beta \equiv \alpha^{\frac{p+1}{4}} \equiv 3^3 \equiv 27 \equiv 5 \pmod{11}$$

Computing Square Roots Modulo a Prime (cont.)

We now show a probabilistic algorithm to compute square roots modulo primes of the form $p = 4k + 1$.

Theorem: -1 is a quadratic residue modulo $p = 4k + 1$.

Proof: (already given in the course) The Legendre symbol $\left(\frac{-1}{p}\right)$ is

$$\begin{aligned} \left(\frac{-1}{p}\right) &\equiv (-1)^{(p-1)/2} \equiv (-1)^{(4k+1-1)/2} \equiv \\ &\equiv (-1)^{2k} \equiv 1^k \equiv 1 \pmod{p} \end{aligned}$$

QED

Computing Square Roots Modulo a Prime (cont.)

Claim: For any a , both a and $-a$ have the same Legendre symbol modulo $p = 4k + 1$ (thus they are both quadratic residues or both quadratic non-residues).

Proof: By Legendre we get

$$\left(\frac{-a}{p}\right) = \left(\frac{-1}{p}\right) \cdot \left(\frac{a}{p}\right) = 1 \cdot \left(\frac{a}{p}\right) = \left(\frac{a}{p}\right).$$

QED

Computing Square Roots Modulo a Prime (cont.)

Let m be a quadratic residue modulo p and let $r^2 \equiv m \pmod{p}$.

Assume WLG that $m \not\equiv 0 \pmod{p}$ (otherwise $r \equiv 0 \pmod{p}$). Then, $r \not\equiv 0 \pmod{p}$.

The solutions of $x^2 \equiv m \pmod{p}$ are $x \equiv \pm r \pmod{p}$.

Computing Square Roots Modulo a Prime (cont.)

Fact: Let $0 \leq \delta < p$, $\delta \neq r$. Then $\delta + r$ and $\delta - r$ have the same Legendre symbol iff

$$(\delta + r)/(\delta - r) \triangleq (\delta + r)(\delta - r)^{-1}$$

is a quadratic residue modulo p .

Claim: When δ gets all its possible values $0 \leq \delta < p$, except $\delta \equiv r$, the ratio $(\delta + r)/(\delta - r)$ gets all the values $0 \leq \gamma < p$, except for $\gamma \equiv 1$.

Computing Square Roots Modulo a Prime (cont.)

Proof:

- (a) Assume that some γ is received from two distinct δ 's: $\delta_1 \neq \delta_2 \pmod{p}$. Then,

$$(\delta_1 + r)/(\delta_1 - r) \equiv (\delta_2 + r)/(\delta_2 - r) \pmod{p}$$

From which the following equations are derived:

$$\begin{aligned} (\delta_1 + r)(\delta_2 - r) &\equiv (\delta_2 + r)(\delta_1 - r) \pmod{p} \\ \delta_1\delta_2 + r\delta_2 - r\delta_1 - r^2 &\equiv \delta_1\delta_2 + r\delta_1 - r\delta_2 - r^2 \pmod{p} \\ r(\delta_2 - \delta_1) &\equiv -r(\delta_2 - \delta_1) \pmod{p} \\ 2r(\delta_2 - \delta_1) &\equiv 0 \pmod{p} \end{aligned}$$

Since $r \not\equiv 0 \pmod{p}$, we get:

$$\delta_1 \equiv \delta_2 \pmod{p}.$$

Contradiction. Thus, all the received γ 's are distinct.

Computing Square Roots Modulo a Prime (cont.)

- (b) It remains only to show that $\gamma \not\equiv 1 \pmod{p}$:

But, if $(\delta + r)/(\delta - r) \equiv 1 \pmod{p}$ then $(\delta + r) \equiv (\delta - r) \pmod{p}$, and thus $r \equiv 0 \pmod{p}$. Contradiction.

QED

Computing Square Roots Modulo a Prime (cont.)

Conclusion: Exactly half of the values of δ satisfy that $(\delta + r)$ and $(\delta - r)$ have the same Legendre symbol.

Proof: Exactly half of the values $\gamma = 1, \dots, p-1$ are quadratic residues, and all of them, except 1 are received by various δ 's. The value 1 is a quadratic residue that is not received, but instead the quadratic residue 0 is received. QED

Computing Square Roots Modulo a Prime (cont.)

The Algorithm:

Concentrate on the polynomial

$$f(x) \equiv x^2 - m \equiv (x+r)(x-r) \pmod{p}.$$

Then

$$f(x-\delta) \equiv (x+r-\delta)(x-r-\delta) \equiv (x-(\delta-r))(x-(\delta+r)) \pmod{p}.$$

Exactly for half of the values of δ , only one of $\delta+r$ and $\delta-r$ is a quadratic residue, and the other is a quadratic non-residue. From now on, we concentrate only on these values of δ . Thus, only one of the roots $\delta+r$ and $\delta-r$ of $f(x-\delta)$ is a quadratic residue.

Computing Square Roots Modulo a Prime (cont.)

The polynomial $x^{(p-1)/2} - 1 \pmod{p}$ is of degree $(p-1)/2$, and whose roots are exactly all the quadratic residues modulo p . By denoting all the quadratic residues by $\rho_1, \rho_2, \dots, \rho_{(p-1)/2}$, we get

$$x^{(p-1)/2} - 1 \equiv (x - \rho_1)(x - \rho_2) \dots (x - \rho_{(p-1)/2}) \pmod{p}.$$

Since only one of the roots of $f(x-\delta)$ is a quadratic residue, only this root is also a root of $x^{(p-1)/2} - 1 \pmod{p}$ — thus only one of $\delta \pm r$ is one of the ρ_i 's.

We can find it by computing gcd of polynomials:

$$\gcd(x^{(p-1)/2} - 1, f(x-\delta)) = x - \rho_i = x + r - \delta \text{ or } x - r - \delta.$$

On average, two trials of δ are required to find the square root.

Computing Square Roots Modulo a Prime (cont.)

Example: Compute the square root of 3 modulo 13.

- Choose $\delta = 7$: Then

$$\begin{aligned} f(x-\delta) &\equiv (x-7)^2 - 3 \equiv x^2 - 14x + 49 - 3 \equiv \\ &\equiv x^2 - x + 7 \pmod{13} \\ x^{(p-1)/2} - 1 &\equiv x^6 - 1 \pmod{13} \end{aligned}$$

By computing the gcd we get:

$$\gcd(x^2 - x + 7, x^6 - 1) = x - 3$$

Thus,

$$\begin{aligned} x - \delta \pm r &\equiv x - 3 \\ \pm r &\equiv -3 + \delta \equiv 4 \pmod{13} \\ r &\equiv \pm 4 \pmod{13} \end{aligned}$$

Computing Square Roots Modulo a Prime (cont.)

- If we choose $\delta = 5$ we get

$$\begin{aligned} f(x-\delta) &\equiv (x-5)^2 - 3 \equiv x^2 - 10x + 25 - 3 \equiv \\ &\equiv x^2 - 10x - 4 \pmod{13} \end{aligned}$$

By computing the gcd we get:

$$\gcd(x^2 - 10x - 4, x^6 - 1) = x^2 - 10x - 4$$

so that both roots are quadratic residues, and really $5+r = 9$ and $5-r = 1$ (we already found that $r = \pm 4$).

Computing Square Roots Modulo a Prime (cont.)

- If we choose $\delta = 2$ we get

$$\begin{aligned} f(x-\delta) &\equiv (x-2)^2 - 3 \equiv x^2 - 4x + 4 - 3 \equiv \\ &\equiv x^2 - 4x + 1 \pmod{13} \end{aligned}$$

By computing the gcd we get:

$$\gcd(x^2 - 4x + 1, x^6 - 1) = 1$$

and thus both roots are quadratic non-residues.

Computing Square Roots Modulo $n = pq$

Example: Compute the square root of 3 modulo $11 \cdot 13$.

We have seen that:

- ± 5 are the square roots of 3 (mod 11).
- ± 4 are the square roots of 3 (mod 13).

The 4 solutions of:

$$\begin{cases} u \equiv \pm 5 \pmod{11} \\ u \equiv \pm 4 \pmod{13} \end{cases}$$

are the square roots of 3 modulo $11 \cdot 13$.

Computing Square Roots Modulo $n = pq$ (cont.)

by using the Chinese remainder theorem:

$$\begin{aligned} u_1 &\equiv 4 \cdot 6 \cdot 11 + 5 \cdot 6 \cdot 13 \equiv 82 \pmod{11 \cdot 13} \\ u_2 &\equiv -4 \cdot 6 \cdot 11 + 5 \cdot 6 \cdot 13 \equiv 126 \pmod{11 \cdot 13} \\ u_3 &\equiv -u_2 \equiv 4 \cdot 6 \cdot 11 - 5 \cdot 6 \cdot 13 \equiv 17 \pmod{11 \cdot 13} \\ u_4 &\equiv -u_1 \equiv -4 \cdot 6 \cdot 11 - 5 \cdot 6 \cdot 13 \equiv 61 \pmod{11 \cdot 13} \end{aligned}$$

Note that:

$$\begin{aligned} 13^{-1} &\equiv 6 \pmod{11} \\ 11^{-1} &\equiv 6 \pmod{13} \end{aligned}$$

The Density of Prime Numbers

For many applications, we need to find large “random” primes. Fortunately, large primes are not too rare, so it is not too time consuming to test random integers of the appropriate size until a prime is found.

The **prime number function** $\pi(n)$ specifies the number of primes that are less than or equal n .

Examples: $\pi(10) = 4$.

The Density of Prime Numbers (cont.)

Prime Number Theorem:

$$\lim_{n \rightarrow \infty} \frac{\pi(n)}{n/\ln n} = 1$$

We can use the prime number theorem to estimate the probability that a randomly chosen integer n is a prime as $\frac{1}{\ln n}$. Thus, we need to examine approximately $\ln n$ integers chosen randomly near n in order to find a prime that is of the same length as n (this figure can be cut in half by choosing only odd integers).

Primality Tests

We want to know whether a given number n is prime.

$$\text{Primes} = \{n : n \text{ is a prime number in binary representation}\}$$

- It is easy to show that $\text{Primes} \in \text{coNP}$.
Primes $\in \text{NP}$ (Pratt 75).
- Primes $\in \text{coRP}$ (Solovay-Strassen 77, Rabin 80).
Primes $\in \text{RP}$.
Thus, Primes $\in \text{ZPP} = \text{RP} \cap \text{coRP}$.

In 2002, Agrawal, Kayal and Saxena have shown that $\text{Primes} \in P$. However, the time complexity of their algorithm is $O(\log^{12}(n))$.

Note:

Monte Carlo algorithms - BPP ($\text{RP}, \text{coRP} \subseteq \text{BPP}$).
Las Vegas algorithms - ZPP.

Primality Tests (cont.)

The following is a simple primality test, based on Euler's theorem.

Choose some $0 < a < n$, and test whether

$$a^{n-1} \equiv 1 \pmod{n}.$$

By Fermat's theorem, the equation holds for any prime number n , and for any a .

Thus, if this equation does not hold: n is composite. If the equation holds: try another a .

Primality Tests (cont.)

Does such a test suffice? Can we conclude that if we even tried many a 's in $0 < a < n$, and the equations hold, then n is a prime?

No!

There are composite numbers for which for any a coprime to n , $a^{n-1} \equiv 1 \pmod{n}$. These numbers are called **Carmichael numbers**.

The smallest Carmichael number is $561 = 3 \cdot 11 \cdot 17$, for which

$$\text{lcm}(3-1, 11-1, 17-1) = 80 | 560 = 561 - 1.$$

Indeed,

$$\begin{cases} a^{560} \equiv 1 \pmod{3} \\ a^{560} \equiv 1 \pmod{11} \\ a^{560} \equiv 1 \pmod{17} \end{cases}$$

Solovay-Strassen Primality Test

Ref: A Fast Monte-Carlo Test for Primality, SIAM Journal of Computing, V. 6, No. 1, March 1977. Correction in V. 7, No. 1, February 1978.

The Algorithm:

1. Let n be some odd number. We wish to test whether n is prime.
2. Choose some random number a , $1 < a < n$. If $\text{gcd}(a, n) \neq 1$ then n is not prime.
3. Compute the values

$$\begin{aligned} \epsilon &\equiv a^{(n-1)/2} \pmod{n} \\ \delta &\equiv \left(\frac{a}{n}\right) \quad (\text{Jacobi symbol}) \end{aligned}$$

4. If $\text{gcd}(a, n) > 1$ or $\epsilon \neq \delta$ then n is necessarily composite.

Solovay-Strassen Primality Test (cont.)

5. Otherwise n is probably a prime with probability $\geq 1/2$.
6. Execute the above test m times:
 - (a) If the algorithm outputs 'Composite' at least once: output 'Composite'.
 - (b) If the algorithm output 'Possibly Prime' in all the m trials: output 'Prime'.

Solovay-Strassen Primality Test (cont.)

Theorem: If n is an odd prime, the algorithm always outputs 'Prime', i.e., for any a

$$a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n}.$$

Proof: By Euler's criterion and the definition of Legendre's symbol. QED

Solovay-Strassen Primality Test (cont.)

The following theorem states that at least half of the a 's are witnesses to the fact that n is composite.

Theorem: If n is an odd composite, at most half of the numbers $a \in Z_n^*$ satisfy

$$a^{(n-1)/2} \equiv \left(\frac{a}{n}\right) \pmod{n}.$$

Proof: First we show that there exists some b such that

$$b^{(n-1)/2} \not\equiv \left(\frac{b}{n}\right) \pmod{n}.$$

Solovay-Strassen Primality Test (cont.)

1. If n is divisible by some prime power p^e ($p > 2$, $e \geq 2$, $p^{e+1} \nmid n$), we choose

$$b = 1 + \frac{n}{p}.$$

Note that $p \mid \varphi(n)$ because $\varphi(p^e) = (p-1)p^{e-1}$.

Also note that $\gcd(b, \frac{n}{p}) = 1$, which implies $\gcd(b, n) = 1$.

Denote $n = p^e q_1 q_2 \dots q_k$, where the q_i 's are not necessarily distinct.

Then,

$$\left(\frac{b}{n}\right) = \left(\frac{b}{n/p^e}\right) \left(\frac{b}{p}\right)^e = \left(\frac{b}{q_1}\right) \left(\frac{b}{q_2}\right) \dots \left(\frac{b}{q_k}\right) \left(\frac{b}{p}\right)^e$$

but $b \equiv 1 \pmod{q_i}$ for any q_i , and $b \equiv 1 \pmod{p}$. Thus,

$$\left(\frac{b}{n}\right) = 1.$$

Solovay-Strassen Primality Test (cont.)

On the other hand,

$$b^{(n-1)/2} \not\equiv 1 \pmod{n}$$

since if we assume the contrary, and denote the order of b modulo p^e by d ($b^d \equiv 1 \pmod{p^e}$), then $d \mid \frac{(n-1)}{2}$, and

$$d \mid n - 1.$$

Denoting $b \equiv 1 + kp^{e-1} \pmod{p^e}$, for $k = n/p^e$, we get by the Binom that

$$1 \equiv b^d \equiv 1 + dkp^{e-1} + \text{Some multiple of } p^e \pmod{p^e}.$$

Therefore, $dkp^{e-1} \equiv 0 \pmod{p^e}$, from which we get $p \mid dk$. Since $\gcd(k, p) = 1$, we conclude that $p \mid d$. Recall that $d \mid n - 1$, therefore,

$$p \mid n - 1$$

Solovay-Strassen Primality Test (cont.)

But

$$p \mid n.$$

Therefore, $p \mid 1$, i.e., $p = 1$. Contradiction.

Solovay-Strassen Primality Test (cont.)

2. If n is a product of distinct primes, and is not divisible by any square of a prime:

Let p be any prime factor of n , and denote $n = pq_1 q_2 \dots q_k$, where p and the q_i 's are all distinct.

Choose a quadratic non-residue s modulo p , and choose b by the Chinese remainder theorem:

$$\begin{aligned} b &\equiv s \pmod{p} \\ b &\equiv 1 \pmod{n/p} \end{aligned}$$

Then,

$$\left(\frac{b}{n}\right) = \left(\frac{b}{p}\right) \left(\frac{b}{q_1}\right) \left(\frac{b}{q_2}\right) \dots \left(\frac{b}{q_k}\right) = (-1) \cdot 1 \cdot 1 \dots 1 = -1$$

Solovay-Strassen Primality Test (cont.)

On the other hand:

$$b^{(n-1)/2} \equiv 1 \pmod{n/p}$$

and thus

$$\begin{aligned} b^{(n-1)/2} &\not\equiv -1 \pmod{n/p} \\ b^{(n-1)/2} &\not\equiv -1 \pmod{n} \end{aligned}$$

We conclude that for any modulo n there is some b for which the equation does not hold, and

$$b^{(n-1)/2} \not\equiv \left(\frac{b}{n}\right) \pmod{n}.$$

Solovay-Strassen Primality Test (cont.)

3. We now show that at least half of the numbers do not satisfy the equation.

Let w_1, w_2, \dots, w_t all the numbers in Z_n^* that satisfy

$$w_i^{(n-1)/2} \equiv \left(\frac{w_i}{n}\right) \pmod{n}.$$

Define u_1, u_2, \dots, u_t by

$$u_i \equiv bw_i \pmod{n}, \quad i = 1, \dots, t.$$

All the numbers u_1, u_2, \dots, u_t are distinct, and all of them are coprime to n and in the range $0 < u_i < n$.

We claim that all the u_i 's do not satisfy the equation, i.e., for any u_i :

$$u_i^{(n-1)/2} \not\equiv \left(\frac{u_i}{n}\right) \pmod{n}.$$

Solovay-Strassen Primality Test (cont.)

Assume the contrary that the equation holds for some u_i :

$$u_i^{(n-1)/2} \equiv \left(\frac{u_i}{n}\right) \pmod{n}.$$

Then,

$$b^{(n-1)/2} w_i^{(n-1)/2} \equiv \left(\frac{b}{n}\right) \left(\frac{w_i}{n}\right) \pmod{n}.$$

But

$$w_i^{(n-1)/2} \equiv \left(\frac{w_i}{n}\right) \pmod{n}.$$

and thus

$$b^{(n-1)/2} \equiv \left(\frac{b}{n}\right) \pmod{n}.$$

Contradiction for the choice of b .

Solovay-Strassen Primality Test (cont.)

Thus, all the u_i 's do not satisfy the equation. Since they are all distinct, for any number w_i which satisfy the equation, there is at least one other number which do not satisfy the equation. Thus, the probability that a random a do not satisfy the equation is at least half.

QED

Solovay-Strassen Primality Test (cont.)

Complexity of the Primality Test:

- gcd computation: $O(\log n)$ divisions.
- ϵ : $O(\log n)$ modular operations.
- δ : $O(\log n)$ divisions.
- In total: $O(\log n)$ for any choice of a .
- In order to get probability 2^{-m} for an error (output 'Prime' for a composite number) the algorithm tries m a 's. The total complexity is thus $O(m \log n)$.
- If n is a composite, it is identified on average after trying two a 's. The complexity in this case is $O(2 \log n) = O(\log n)$.