Podstawy kryptografii

Andrzej M. Borzyszkowski

Instytut Informatyki
Uniwersytet Gdański

sem. letni 2023/2024

inf.ug.edu.pl/~amb/

Szukanie liczb pierwszych

- Algorytm oczywisty
  – losuje się ciąg n bitów = duża liczba
  – testuje się, czy liczba jest pierwsza, jeśli nie, to powtórka
- Złożoność algorytmu
  – liczba liczb pierwszych n-bitowych $\geqslant \frac{C}{n}$ dla pewnej stałej C
  – a więc średnio n iteracji starcza do znalezienia l.pierwszej
  – problem testowania pierwszości liczby
  – oczywiście nie poprzez próbę rozkładu, ten problem oceniamy jako trudny
  – stosowane w praktyce algorytmy mogą przyjąć, że liczba jest pierwsza nawet jeśli taka nie jest (zaniedbywalne prawdopod. błędu)

Szukanie liczb pierwszych

Testy pierwszości, Fermata

- Tw. Fermata: jeśli n jest pierwsza (i $NWD(a, n) = 1$), to $a^{n-1} = 1 \mod n$
  – a więc na pewno nie jest pierwsza w.p.p.
  – przedtem można/warto obliczyć $NWD(a, n)$
  –„świadek pierwszości": liczba a t.ż. $a^{n-1} = 1 \mod n$
  – wówczas n jest„pseudopierwsza przy podstawie a"
- albo wszystkie $a < n$ są świadkami albo mniej niż połowa
  – dw: jeśli a nie jest świadkiem ale b jest, to $a \cdot b$ nie jest, czyli świadków b nie może być więcej niż połowa
  – np. $10^{32} = 1 \mod 33$, ale $2^{32} = 4 \mod 33$
  – liczby Carmichaela: nie są pierwsze, ale spełniona jest teza tw. Fermata – jest ich bardzo niewiele, np. 561
  – a więc: jeśli wiele losowych liczb a jest świadkiem pierwszości n, to szansa, że liczba n nie jest pierwsza jest niewielka (np. $\leqslant 2^{-40}$)

Testy pierwszości, Millera—Rabina

- Jeśli $x^2 = y^2 \mod n$ ale $x \neq \pm y$ (różne pierwiastki), to znajdziemy rozkład n
  – dw. $n|(x-y)\cdot(x+y)$,
  z założenia n nie dzieli żadnego czynnika, więc jest złożone
  i $NWD(x-y, n)$ daje rozkład

Testy pierwszości, Millera—Rabina, c.d.

- obliczamy a^{n-1} dla pewnego a, $n-1 = m\cdot 2^k$, $k > 0$
- zamiast szybkiego potęgowania odkładamy kwadraty na koniec
  definiujemy ciąg $b_0 = a^m \mod n$, $b_{i+1} = b_i^2 \mod n$
  – jeśli a jest świadkiem pierwszości (Fermat), to $\exists i.\, b_i = 1 \mod n$,
  jeśli $i = 0$ albo $b_{i-1} = -1 \mod n$, to nic nie zyskujemy
  – ale jeśli $b_{i-1} \neq -1 \mod n$, to widać nietrywialny pierwiastek
  z jedynki i na pewno n jest złożona, nawet znany jest rozkład
  – np. $n = 561$, $n-1 = 35\cdot 2^4$, testujemy dla $a = 2$
  $b_0 = 2^{35} = 263 \mod 561$
  $b_1 = 263^2 = 166 \mod 561$
  $b_2 = 166^2 = 67 \mod 561$
  $b_3 = 67^2 = 1 \mod 561$
  $NWD(67-1, 561) = 33$, $561 = 33\cdot 17$

Szukanie liczby pierwszej

- Dana liczba n, sprawdzamy, czy 2, 3, 5, 7 i inne małe liczby są świadkami pierwszości w sensie Rabina–Millera
  – tw.: jeśli n jest złożona, to mniej niż $\frac{1}{4}$ liczb jest świadkiem pierwszości w sensie Rabina–Millera
  – np. 2 jest rzadko świadkiem pierwszości liczby złożonej
  – dla większości liczb złożonych świadków pierwszości w sensie Rabina–Millera jest dużo mniej niż $\frac{1}{4}$
  – po kilkunastu/kilkudziesięciu nieudanych próbach znalezienia nieświadka uznajemy liczbę za pierwszą
- Złożoność: dla liczby n-bitowej wynosi n^3
  – n iteracji algorytmu (na pewno n^2 jest wystarczające)
  – C testów na świadka (dla liczby złożonej kontrprzykład znajdzie się po kilku testach, można postawić granicę np. $C = 20$ testów)
  – n potęgowań dla testowania jednego świadka

Liczby silnie pierwsze

- Liczba p jest silnie pierwsza jeśli
  – p jest pierwsza
  – $(p-1)/2$ też jest pierwsza
- Np. $p = 2^k + 1$ może być liczbą pierwszą
  – ale $p-1$ jest potęgą 2
  – tak daleko odbiega od silnej pierwszości, jak to możliwe
- Szukanie liczb silnie pierwszych
  – szukamy liczby pierwszej q (jedna na $\ln(q)$ liczb)
  – sprawdzamy czy $2\cdot q + 1$ jest pierwsza (też jedna szansa na $\ln(q)$)
  – złożoność: jedna potęga wyżej
- Słabsza wersja:
  – $(p-1)/2$ ma "duży" dzielnik pierwszy
  – szukamy liczby pierwszej q np. 160-bitowej
  – następnie testujemy na pierwszość liczby $2\cdot\alpha\cdot q + 1$ dla liczb α mających $n-160$ bitów i znajdujemy liczbę n bitową