Chapter 10. ABUNDANCE of WITNESSES TECHNIQUE

Abundance of witnesses (AOW) techniques is for two reasons “a real jewel” between methods of the design of randomized algorithms:

- **An application of this techniques usually requires to use deep results from mathematics.**
- **This techniques is deeply related to the question of whether or not, for a given problem, randomized mode of computation can be more efficient than the deterministic mode.**

A witness for a problem P and an input x is such an additional information y that helps to solve the problem P for an input x in much more efficient way than one can do that without y.

In this chapter we will discuss and illustrate this important technique for the design of randomized algorithms.

It is also worth to notice that the fingerprinting technique can be also seen as a simplified case of the abundance of witnesses techniques for the cases that witnesses are easy to find.

BASIC IDEAS – ABUNDANCE of WITNESS TECHNIQUE

- In order to apply the AOW technique, one needs to have for every input instance a set of candidates for witnesses in which there is an abundance (say more than half) of witnesses. If it is the case, one gets a witness with sufficient probability if one chooses randomly an element from such a set of candidates for witnesses.
- An important question, namely, whether for a given algorithmic problem P randomized algorithms can be more efficient than deterministic, is closely related to the question whether witnesses for P are randomly distributed or deterministically ordered.
- If witnesses are ordered, and one can discover this order for a given input instance, then one can find witnesses in a deterministic way and in such a case deterministic algorithms for a given problem can be efficient.
- Randomized algorithms are more efficient only in the case witnesses are randomly distributed in the set of potential witnesses and there is a lot of them.

BASIC IDEAS II

It may happen that there is an order in the set of witnesses, but this order is very hard to find.

An excellent example is the primality testing problem.

Inefficient methods to test primality have been known for years. Two main randomized tests are shown in next slides.

Around 1977 the first randomized algorithms were designed and all of them used abundance of witnesses technique.

In 2002 a definition of witnesses for primality testing was discovered with the property that if p is composite, then there must always be a witness of p's compositeness among the smallest candidates for a witness.

This led to the discovery, by three scientists, M. Agrawal, N. Kayal and N. Saxena, from Kanpur, of the deterministic primality testing algorithm for integers n with the complexity $\mathcal{O}((\lg n)^{12})$. This has been later improved, by Lenstra, to an $\mathcal{O}((\lg n)^6)$ algorithm.

RABIN-MILLER PRIMALITY TESTING

A simple randomized Rabin-Miller's *Monte Carlo* algorithm for prime recognition is based on the following result from the number theory.

Lemma Let $n \in \mathbf{N}$. Denote, for $1 \leq x < n$, by $C(x)$ the condition:

> Either $x^{n-1} \not\equiv 1 (\mathrm{mod}\ n)$ or there is an $m \in N$, $m = \frac{(n-1)}{2^i}$ for some i, such that $gcd(n, x^m - 1) \neq 1$

Fact: If $C(x)$ holds for some $1 \leq x < n$, then n is not prime. If n is not prime, then $C(x)$ holds for at least half of x between 1 and n.

In other words, most of the numbers between 1 and n are witnesses for composability of n.

Rabin-Miller algorithm

- Choose randomly integers $x_1, \ldots, x_m$ such that $1 \leq x_j < n$;
- For each x_j determine whether $C(x_j)$ holds;
- **if** $C(x_j)$ holds for some x_j;
  then n is not prime
  else n is prime, with probability of error 2^{-m}

LEGENDRE and LEGENDRE-JACOBI SYMBOLS

In order to present second randomized algorithm for primality testing we need to introduce concepts of Legendre and Legendre-Jacobi symbols that play a very importat role in the numer theory.

Notation QR_n is the set of quadratic residua, that is of integers x such that $x = y^2 \mod n$ for some y; Complement of QR_n is denoted QNR_n.

Let us now define:

$$(x|m) = \begin{cases} 1 & \text{if } x \in \mathrm{QR}_m \text{ and } m \text{ is prime} \\ -1 & \text{if } x \in \mathrm{QNR}_m \text{ and } m \text{ is prime} \\ \prod_{i=1}^{n}(x|p_i) & \text{if } m = \prod_{i=1}^{n} p_i,\ p_i \text{ are primes, } \gcd(x,m) = 1 \end{cases}$$

$(x|m)$ is called the Legendre symbol if m is prime and the Legendre-Jacobi symbol otherwise.

Rules to compute $(x|m)$

1. Euler's criterion: $x|p \equiv x^{\frac{p-1}{2}} \pmod{p}$ if $p > 2$ is prime, $x \in \mathbf{Z}_p^*$
2. If $x \equiv y \pmod{m}$, then $(x|m) = (y|m)$.
3. $(x|m) \cdot (y|m) = (xy|m)$.
4. $(-1|m) = (-1)^{\frac{m-1}{2}}$, if m is odd.
5. $(2|m) = (-1)^{\frac{m^2-1}{8}}$, if m is odd
6. Law of quadratic reciprocity: If $\gcd(m, n) = 1, m, n$ are odd, then $(n|m)(m|n) = (-1)^{\frac{(m-1)(n-1)}{4}}$.

Example:

$$\begin{aligned}
(28|97) &= (2|97)(2|97)(7|97) = (7|97) \\
&= (97|7)(-1)^{\frac{(97-1)(7-1)}{4}} = (6|7) \\
&= (2|7)(3|7) = (-1)^6(3|7) = (7|3)(-1)^3 = -(1|3) = -1
\end{aligned}$$

At least 72 different proofs are known for the law of quadratic reciprocity.

SOME TECHNICAL RESULTS

Theorem (Lagrange) If $(H, \circ)$ is a subgroup of a group $(G, \circ)$, then $|H|$ divides $|G|$.

Definition Euler quotient function is defined as follows

$$\Phi(n) = |\{m; 1 \leq m < n, m \text{ does not divide } n\}.$$

Theorem (Euler's Totient Theorem)
$n^{\Phi(m)} \equiv 1 \pmod{m}$ if $n < m, \gcd(m, n) = 1$
Corollary $n^{-1} \equiv n^{\Phi(m)-1} \pmod{m}$ if $n < m, \gcd(m, n) = 1$

Theorem (Fermat's Little Theorem)
For any a, it holds $a^p \equiv a \pmod{p}$ if p is prime.

Proof. Theorem is true for $a = 1$. Assume it is true for some a. By induction

$$(a + 1)^p \equiv a^p + 1 \equiv a + 1 \mod p.$$

SOLOVAY-STRASSEN's PRIME RECOGNITION ALGORITHM

It follows from the Lagrange theorem that if the following fast Monte Carlo algorithm — based on the fact that computation of Legendre-Jacobi symbols can be done fast — reports that a given number n is composite, then this is 100%, true and if it reports that it is a prime, then the error is at most $\frac{1}{2}$.

begin choose randomly an integer $a \in \{1, \ldots, n\}$
  if $\gcd(a, n) \neq 1$ **then return** "composite"
    else if $(a|n) \not\equiv a^{\frac{n-1}{2}} (\mathrm{mod} n)$
      then return "composite";
  return "prime"
end

Indeed, if n is composite, then all integers $a \in \mathbf{Z}_n^\star$ such that $(a|n) \equiv a^{\frac{n-1}{2}} (\mathrm{mod} n)$ form a proper subgroup of the group $\mathbf{Z}_n^*$. This implies that most of the elements $a \in \mathbf{Z}_n^*$ are such that $(a|n) \not\equiv a^{\frac{n-1}{2}} (\mathrm{mod}\ n)$ and therefore they can "witness" coppositeness of n, if n is composite.

COMBINATION of TWO RANDOMIZED PRIMALITY TESTING ALGORITMOV

1. Soloway and Strassen (and also Rabin, independently) developed an algorithm such that for given integer n and another number a
   - If n is prime, then for any choice of a the output of the algorithm is prime
   - If n is composite, then for most of the choices of a the output of the algorithm is composite.
2. Adleman and Huang dveloped an algorithm such that for given integer n and another number a"
   - If n is prime, then for most of the choices of a, the output is prime
   - If n is composite, then for any choice of a, the output is composite.

If we combine both of these algorithms, will we always get a correct answer? Can we run above algorithms in parallel till either the first one outputs "composite" or second outputs "prime"

No, the trouble is that there is a small chance that none of the possibility ever happen

CHARACTERIZATION of WITNESSES

A good definition of a witness for proving a FACT (for example, that "n is composite") should satisfy the following conditions:

1. A witness of the FACT should offer a possibility to verify the FACT easily.
2. One should be able easily verify for a to-be witness whether it is indeed a witness.
3. It should be possible to specify a set of candidates for witnesses with abundance of witnesses.

Example Fermat's Little Theorem says: For every prime p and every $a \in \{1, 2, \ldots, p-1\}$, $a^{p-1} \bmod p = 1$. We could therefore try to define

A number $a \in \{1, 2, \ldots, n-1\}$ is a witness of the fact "$n \notin$ PRIME" iff

$$a^{n-1} \bmod n \neq 1.$$

However, this does not work because some composite integers have very few such witnesses. There are even integers, so called Carmichael numbers, that have no witnesses of such a type. Examples of Carmichael numbers: 561, 1105, 1729.

ANOTHER ATTEMPT

The following theorem holds

Theorem Let $p > 2$ be an odd integer. Then

$$p \text{ is a prime } \quad \leftrightarrow (a^{\frac{p-1}{2}} \bmod p) \in \{1, p-1\} \text{ for all } a \in \mathbf{Z}_p - \{0\}$$

We could terefore try to define a set of witnesses for $n \geq 3$ as follows:

A number $a \in \{1, 2, \ldots, n-1\}$
is a witness of the fact $n \notin$ PRIME iff
$a^{(n-1)/2} \bmod n \notin \{1, n-1\}$

The following theorem shows that the above definition assures the abundance of witness for at least every second odd integer greater than 2.

Theorem For every Blum integer n, that is such and n that that $n \bmod 4 = 3$, it holds:
(a) if n is a prime, then

$$a^{(n-1)/2} \bmod n \in \{1, n-1\}$$

for all $a \in \{1, \ldots, n-1\}$, and

(b) if n is composite, then

$$a^{(n-1)/2} \bmod \; n \notin \{1, n-1\}$$

for at least half of the elements from $\{1, 2, \ldots, n-1\}$.

SIMPLIFIED SOLOVAY STRASSEN'S ALGORITHM

Algorithm works only for Blum integers:

1. Choose randomly $a_i \in \{2, \ldots, n-1\}$, for $i = 1, \ldots, k$;
2. Compute all $A_i = a_i^{(n-1)/2} \bmod n$
3. If one $A_i \notin \{-1, 1\}$, then n is surely composite; otherwise probability that n is not prime is smaller than $\frac{1}{2^k}$

PASCAL TRIANGLES

It is a triangle having in $n+1$-th raw, $n \geq 1$, $n+1$ integers, $\binom{n}{i}$, $0 \leq i \leq n$.

$$\begin{array}{ccccccccccccccccc}
&&&&&&&&1&&&&&&&&\\
&&&&&&&1&&1&&&&&&&\\
&&&&&&1&&2&&1&&&&&&\\
&&&&&1&&3&&3&&1&&&&&\\
&&&&1&&4&&6&&4&&1&&&&\\
&&&1&&5&&10&&10&&5&&1&&&\\
&&1&&6&&15&&20&&15&&6&&1&&\\
&1&&7&&21&&35&&35&&21&&7&&1&\\
1&&8&&28&&56&&70&&56&&28&&8&&1
\end{array}$$

One can show that when ignoring first and last number of each raw, namely $\binom{n}{0}$ and $\binom{n}{n}$, then all numbers in the n-th raw are divided by n if and only if n is prime.

Aggrawal, Kayal and Saxenna found an efficient way how to make use of the above fact.

STORY of AKS PRIMALITY TEST

- **On 6.8.2002 Manindra Agrawal, Neeraj Kayal and Nitin Saxena, from the Indian Institute of Technology in Kanpur published the paper "PRIMES are in P" for which they received in 2006 Godel prize and also Fulkerson prize.**
- **To test m digit integer with the AKS algorithm requires $\mathcal{O}(m^{12})$ time.**
- **Shortly after that they improved the algorithm to have complexity $\mathcal{O}(m^{10.5})$ and $\mathcal{O}(m^{7.5})$.**
- **In 2005, Carl Pomerance and H.W. Lenstra, Jr, developed a variant of AKS algorithm with complexity $\mathcal{O}(m^6)$.**

BASIC IDEAS of the AKS METHOD

- **The KAS primality test is based on the following theorem:**
- **Theorem An integer $n \geq 2$ is prime if and only if the polynomial congruence relation**

$$(x-a)^n \equiv (x^n - a) (\mathrm{mod}\ n)$$

holds for all integers a.

- **While the above relation constitutes a primality test in itself, verifying it takes exponential time.**

- **To reduce computational complexity, AKS makes use of the related congrueence**

$$(x-a)^n \equiv (x^n - a) (\mathrm{mod}\ n, x^r - 1)$$

which is the same as

$$(x-a)^n - (x^n - a) \equiv 0 (\mathrm{mod}\ n, x^r - 1)$$

for some polynomials f and g. This congruence can be checked in polynomial time.

- **Thhe proof of correctness consists in showing that there exists a suitably small r and suitably small set of integers A such that, if the above congruence holds for all $a \in A$, then n must be a prime.**

EXERCISES

1. Show that number of primes is infinite. (Already Archimedes new a simple proof.)