CS681 **Computational Number Theory**

Lecture 17: Primality is in NP $\cap$ coNP

Instructor: Piyush P Kurur *Scribe: Ramprasad Saptharishi*

Overview

We shall get into primality testing for integers in the next few classes. We shall build up the details starting with showing that it is in NP $\cap$ coNP, discuss randomized algorithm, and then finally get into deterministic polynomial time testing.

We shall prove Pratt's result that it is in NP $\cap$ coNP.

1 Pratt's Theorem

The problem is the following: we are given a number N as input and we want to check if this is prime.

Remember that the input is given in binary. It would be trivial if N was specified in unary in which case the input size is N and hence primality testing in $O(N^c)$ is trivially accomplished by checking every number less than N if it's a factor or not.

The input is provided in binary and therefore we are looking for an algorithm that runs in time polynomial in the input size, which is $\log N$.

Recall the definition of the classes NP and coNP.

Definition 1. NP *is the class of languages L such there exists a polynomial time verification scheme $A(x, y)$ such that $x \in L$ if and only there exists a witness y such that $|y| < |x|^c$ for some constant c and $A(x, y) = 1$.*

coNP *is the class of languages L such that $\bar{L} \in$ NP.*

To get a more intuitive picture, NP is the class of problems that have very short proofs or witnesses. Though it might not be clear if $x \in L$, given a witness y, it is easy to check that (x, y) is a proper solution. For example, sudoku. It might be hard to find a solution but once someone gives us a solution, it is easy to check if the solution is correct.

One could also think of this as guessing a witness y and verifying it using A.

Here is an obvious observation:

Observation 1. *Primality testing is in* coNP.

This equivalent to saying that checking if a number N is composite is in NP which is immediate since the witness is the factor d of the number. Hence, our verification scheme $A(N, d)$ is just checking if d divides N.

Pratt showed that primality testing is infact in NP.

Theorem 2. *Primality testing is in* NP.

Proof. Note that the group $(\mathbb{Z}/N\mathbb{Z})^\star$ is of order $N - 1$ if and only if N is prime. And more over, it is a cyclic group of order $N - 1$ if and only if N is a prime. Thus, we shall find a witness or a certificate that the group is cyclic.

How do we show that a group is cyclic? We guess a generator a. If we are able to show that $a^n \neq 1$ for any $n < N - 1$, we are done. Note that $a^{N-1} = 1$ anyway. Therefore, we just need to check that $a^{(N-1)/p_i} \neq 1$ for every prime divisor p_i of $N - 1$.

Therefore, we not only guess the generator a, we guess the factors $p_1, p_2, \cdots, p_k$ of $N - 1$. But how do we know that the guessed p_is are indeed primes? We guess its witnesses too; induction! Aren't we going in circles? Actually no since the numbers p_is are quite small and it still won't blow up the size of the final certificate.

Let us try and see how large the witness/certificate can get. How large can the prime factors of $N - 1$ be? Since N is prime , $N - 1$ is certainly composite (unless N was 2, a worthless case which can be handled right at the beginning). The largest factor of $N - 1$ can be of size atmost $\sqrt{N}$. How many factors can there exist? Atmost $\log(N - 1)$ of them. Thus if $N - 1 = p_1 p_2 \cdots p_k$ then our witness would be $(a, p_1, p_2, \cdots, p_n)$ and the certificates of each of the p_is. The input is of size $\log N$ and let $S(l)$ be the size of the witness for input of length l. Then:

$$
\begin{aligned}
S(\log N) &= \log^2 N + S(\log p_1) + S(\log p_2) + \cdots + S(\log p_k) \\
&= \log^2 N + (\log p_1)^c + (\log p_2)^c + \cdots + (\log p_k)^c \\
&\leq \log^2 N + (\log p_1 + \log p_2 + \cdots + \log p_k)^c \\
&= \log^2 N + (\log(N - 1))^c \\
&= O((\log N)^c)
\end{aligned}
$$

And since the witness is just polynomially bounded in the size of the input, we can guess the entire certificate and verify. Thus primality testing is in NP. □

And since primality is in NP and coNP, it is in NP $\cap$ coNP.