Density of Primes

- Qn: • How many primes in $1, \dots, x$? $\pi(x)$.
  - ▷ $\pi(x) \to \infty$.
  - • What's $\pi(x)/x$? $\approx 1/\log x$.
    [Gauss conjectured]

Theorem (Density estimate; Erdös 1930s):
$(x/\log_2 x) - 2 < \pi(x) < (5x/\log_2 x)$, $\forall x \geq 5$.

- This needs three simple counting lemmas:

Lemma 1 (lower bd.): $\binom{2n}{n} \geq 4^n/(2n+1)$.

Lemma 2: If prime-power $p^\nu \mid \binom{2n}{n}$, then $p^\nu \leq 2n$.

Lemma 3 (upper bd.): $\prod_{\text{prime } p \leq n} p < 4^n$.

Pf (Lemma 1): $\sum_{i=0}^{2n} \binom{2n}{i} = 4^n \Rightarrow \binom{2n}{n} \geq \frac{4^n}{2n+1}$. (maxima) □

Pf (Lemma 2): $\binom{2n}{n} = \frac{(n+1)\cdots(2n)}{1\cdot 2\cdots n}$. Say, $p^i \mid x$ for $x \in [n]$, then $\exists y \in [n+1 \ldots 2n]$, $p^i \mid y$.

$\Rightarrow$ If $p^v \mid \binom{2n}{n}$, then $p^v > n$ & $p^v \mid y$ for some $y \in [n+1 \dots 2n]$.

$\Rightarrow p^v < 2n$.

□

Pf (Lemma 3): · Base case of induction: $n=5$.

· Let's consider $n =: 2m+1$:

$$\prod_{\text{prime } p \le 2m+1} p \le \left(\prod_{1 \le p \le m+1} p\right) \cdot \left(\prod_{m+1 < p \le 2m+1} p\right)$$

$$< 4^{m+1} \cdot \binom{2m+1}{m} \quad [\because \forall p \in [m+1 \dots 2m+1],\ p \mid \binom{2m+1}{m}]$$

$$< 4^{m+1} \cdot \frac{2^{2m+1}}{2} \quad \left[\binom{2m+1}{m} = \binom{2m+1}{m+1}\right]$$

$$= 4^{2m+1}.$$

$\Rightarrow$ induction holds !

$[n =: 2m]$: $\prod_{\text{prime } p \le 2m} p = \prod_{\text{prime } p \le 2m-1} p \cdot \prod_{\text{prime } p \le 2m-1} p$ $\overset{\text{[ind. hyp.]}}{<} 4^{2m-1} \cdot 4^{2m-1} < 4^{2m}$

$\Rightarrow$ Lemma 3 holds. $\square$

Pf (Density Thm): • Let's show the upper bound on $\pi(x)$. We'll use Lemma 3.

• #primes in $(\sqrt{x}, x]$ is $\pi(x) - \pi(\sqrt{x}) =: N$.

$\Rightarrow (\sqrt{x})^N < \prod_{p \le x} p < 4^x$

$\Rightarrow (\pi(x) - \pi(\sqrt{x})) \cdot \log \sqrt{x} < 2x$

$$\Rightarrow \quad \pi(x) < \frac{4x}{\log_2 x} + \pi(\sqrt{x})$$

$$< \frac{4x}{\log_2 x} + \sqrt{x} < \frac{5x}{\log_2 x}$$

[for $x \geq 2$]

- Let's show the lower bound on $\pi(x)$:
  We'll mainly use Lemma 1.

$$\frac{4^n}{2n+1} < \binom{2n}{n} =: \prod_{p \mid \binom{2n}{n}} p^{\nu_p} \leq \prod_{p \mid \binom{2n}{n}} (2n) \leq (2n)^{\pi(2n)}$$

$\Rightarrow$ Take $\log_2(\cdot)$ both sides:

$$2n - \log(2n+1) \leq \pi(2n) \cdot \log 2n$$

$$\Rightarrow \quad \pi(2n) \geq \frac{2n}{\log 2n} - \log_{2n}(2n+1)$$

- For arbitrary x, pick an $n \in \mathbb{N}$ s.t. $2n < x \leq 2(n+1)$.

$$\Rightarrow \quad \pi(x) \geq \pi(2n) \geq \frac{2n}{\log 2n} - \log_{2n}(2n+1)$$

$$\geq \frac{x-2}{\log_2 x} - \log_x(4x+4)$$

$$> \frac{x}{\log_2 x} - 2 \quad , \text{ for } x \geq 5. \qquad \square$$

Corollary: $\forall x \geq 5$, $\exists$ prime in $[x, 6x]$.

Pf: · Consider $\pi(6x) - \pi(x)$

$$\simeq \frac{6x}{\log x} - \frac{5x}{\log x} > 0. \quad \square$$

Exercise: (Bertrand's postulate; Chebyshev 1848)
$\forall n \geq 1$, $\exists$ prime in $[n, 2n]$.

Open (Legendre's conjecture, 1800s): $\forall n \geq 2$, $[n^2, (n+1)^2]$ has a prime.

Applications in cryptography

— Goal: Transmitting messages over an insecure channel s.t.
0) errors can be corrected (error-correcting)
1) adversaries can't understand. (security)

Defn: • Alice wants to send B message $M \in \{0,1\}^n$

• Alice encodes $M \xmapsto{E} C$
plaintext — ciphertext.

- Encryption algo $\mathcal{E}$ uses (S, M) to compute C. (S: secret)
- Decryption algo. $\mathcal{D}$ uses (S, C) to compute M.

-Eg. Substitution cipher (Caesar's cipher):

- $S :=$ permutation $\pi : [a \ldots z] \to [a \ldots z]$ $\quad \#\geq 26!$
- $\mathcal{E}(S, M) = C := \pi(M)$
- $\mathcal{D}(S, C) := \pi^{-1}(C) = \pi^{-1}\pi(M) = M.$

▷ Using many M's (which are meaningful texts)

the frequency analysis helps break S.

- Eg. Block cipher : Divide the plaintext into blocks, each is a string in $\{0,1\}^b$. So, message in $\{0,1\}^n$ is seen as $\{0,1\}^{b \times n/b}$ & then encoded block-by-block; using some function

$$F : \{0,1\}^s \times \{0,1\}^b \longrightarrow \{0,1\}^b. \qquad (b \gg 1)$$

↑ secret ↑ block ↑ image

▷ $C := F(S, M)$

- Eg. DES, AES, MD-5, ... (algebra based)

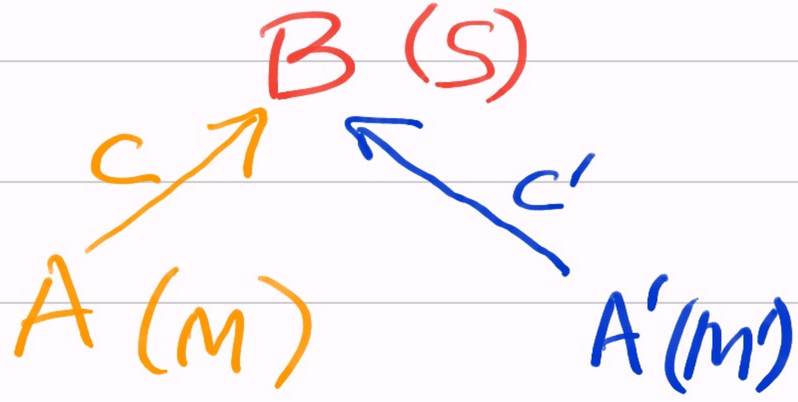
Public-key cryptosystem – RSA

– Most of your "secure" applications are based on RSA.

– Two keys are needed for the bank: Private (S)
Public (P)

– Bank uses $D(S, C)$ $(= M)$.

– Customer uses $E(P, M) =: C$



Requirement: Knowing P & C, one can neither get S nor M.

- In 1977, Rivest, Shamir & Adleman designed the first practical system with such P&S.

- Key-generation: B picks primes $p < q$ ($\approx$400 digits) (guess & verify probabilistically)
  - $n := p \cdot q$ ($\approx$800 digits)
  - Pick an e coprime to $\varphi(n)$. $[1 < e < \varphi(n)]$ $\lceil$ 800 digits
  - Publish $(n, e) =: P$.

Encryption (by customer A): $C \leftarrow M^e \bmod n$
& transmit C to B.

(Algo: Write e in base-2 & use repeated squaring mod n. $\Rightarrow \tilde{} (\log_2 e)\cdot(\log_2 n)$ steps.)

→ Adversary has (C, n, e); yet can't find M by any known fast algo.!

$(p-1)(q-1)$

Decryption (by B): Compute $d := e^{-1} \bmod \varphi(n)$.
(← Ext. Euclid gcd)

· Compute $C^d \bmod n$

$D \equiv M$

Pf: $C^d \equiv M^{ed} \equiv M \cdot M^{k\cdot\varphi(n)} \equiv M \bmod n$.

($\because (M,n)=1$) (Euler's thm)

□

▷ key-gen., Enc., Dec. are efficient.

▷ RSA is secure (if n can't be factored & $C^{1/e} \bmod n$ can't be found & $\varphi(n)$ is hard.)

Exercise: Algo. for $\varphi(n) \Leftrightarrow$ factoring algo.

Exercise: $|p-q|$ very small $\Rightarrow$ factoring algo.

Ex: Algo for $\sqrt{C} \bmod n \Leftrightarrow$ factoring algo.

— Speed-up in Enc.: e is "small".

— " " Dec.: B can use Chinese-Rem. (mod p resp. mod q) to compute $C^d \bmod n$.