coNP

- NP is the class of problems that have succinct certificates (see Proposition 28 on p. 182).
- coNP is the class of problems that have succinct disqualifications:
  - A "no" instance of a problem in coNP possesses a short proof of its being a "no" instance, and only "no" instances have such proofs.
- Clearly $\mathrm{P} \subseteq \mathrm{coNP}$.
- Just because $\mathrm{P} \subseteq \mathrm{NP} \cap \mathrm{coNP}$ does not make $\mathrm{P} = \mathrm{NP} \cap \mathrm{coNP}$.
  - Contrast this with R, RE, and coRE.

Some coNP Problems

- VALIDITY asks if a boolean expression is valid.
- VALIDITY is in coNP.
  - If ϕ is not valid, it can be disqualified very succinctly: a truth assignment that does not satisfy it.
- The disqualification for HAMILTONIAN PATH COMPLEMENT is a Hamiltonian path.

coNP Completeness

Proposition 39 *If L is NP-complete, then its complement $\bar{L} = \Sigma^* - L$ is coNP-complete.*

- Let $\bar{L'}$ be any coNP language.
- Hence $L' \in$ NP.
- Let R be the reduction from L' to L.
- So $x \in L'$ if and only if $R(x) \in L$.
- So $x \in \bar{L'}$ if and only if $R(x) \in \bar{L}$.
- R is a reduction from $\bar{L'}$ to $\bar{L}$.

Corollary 40 VALIDITY *is coNP-complete.*

Possible Relations between P, NP, coNP

- $\mathrm{P} = \mathrm{NP} = \mathrm{coNP}$.
- $\mathrm{NP} = \mathrm{coNP}$ but $\mathrm{P} \neq \mathrm{NP}$.
- $\mathrm{NP} \neq \mathrm{coNP}$ and $\mathrm{P} \neq \mathrm{NP}$ (current "consensus").

coNP Completeness and NP Completeness

Proposition 41 *If a coNP-complete problem is in NP, then* $NP = coNP$.

- Let $L \in$ NP decided by NTM M be coNP-complete.
- For any $L' \in$ coNP, there is a reduction R from L' to L.
- $L' \in$ NP as it is decided by NTM $M(R(x))$.
- The other direction NP $\subseteq$ coNP is symmetric.

The Primality Problem

- An integer p is **prime** if $p > 1$ and all positive numbers other than 1 and p itself cannot divide it.
- PRIMES asks if an integer N is a prime number.
- The method of dividing N by $2, 3, \ldots, \sqrt{N}$ is *not* polynomial-time.
  - The length of N is only $\log N$, but $\sqrt{N} = 2^{0.5 \log N}$.
- No polynomial-time algorithms for PRIMES exist.
- Finding such an algorithm is one of the most important problems in mathematics and computer science.
- Efficient "probabilistic" algorithms for PRIMES exist.

coNP Again

- $\text{NP} \cap \text{coNP}$ is the class of problems that have succinct certificates and succinct disqualifications.
  - Each "yes" instance has a succinct certificate.
  - Each "no" instance has a succinct disqualification.
  - No instance has both.
- $\text{P} \subseteq \text{NP} \cap \text{coNP}$.
- We will see that PRIMES $\in (\text{NP} \cap \text{coNP})$.
- But PRIMES is not known to be in P.
- If PRIMES $\notin \text{P}$, immediately $\text{P} \neq \text{NP}$.

Primitive Roots in Finite Fields

Theorem 42 (Lucas and Lehmer, 1927) *A number $p > 1$ is prime if and only if there is a number $1 < r < p$ (called the* **primitive root** *or* **generator***) such that*

1. $r^{p-1} = 1 \bmod p$, *and*
2. $r^{(p-1)/q} \neq 1 \bmod p$ *for all prime divisors q of $p - 1$.*

- The above theorem can be used to test efficiently primes of the form $2^m + 1$.
- We will prove the theorem later.

Pratt's Theorem

Theorem 43 (Pratt, 1975) PRIMES $\in NP \cap coNP$.

- PRIMES is in coNP because a succinct disqualification is a divisor.
- Suppose p is a prime.
- p's certificate includes the r in Theorem 42 (p. 251).
- Use recursive doubling to check if $r^{p-1} = 1 \bmod p$ in time polynomial in the length of the input, $\log_2 p$.
- We also need all *prime* divisors of $p-1$: $q_1, q_2, \ldots, q_k$.
- Checking $r^{(p-1)/q_i} \neq 1 \bmod p$ is also easy.

The Proof (continued)

- Checking $q_1, q_2, \ldots, q_k$ are all the prime divisors of $p-1$ is easy.
- We still need certificates for the primality of the q_i's.
- The complete certificate is recursive and tree-like:

$$C(p) = (r; q_1, C(q_1), q_2, C(q_2), \ldots, q_k, C(q_k)).$$

- $C(p)$ can also be checked in polynomial time.
- We finally prove that $C(p)$ is succinct.

The Succinctness of the Certificate

Lemma 44 *The length of* $C(p)$ *is at most quadratic at* $4\log_2^2 p$.

- This claim holds when $p = 2$ or $p = 3$.
- In general, $p - 1$ has $k < \log_2 p$ prime divisors $q_1 = 2, q_2, \ldots, q_k$.
- $C(p)$ requires: 2 parentheses, $2k < 2\log p$ separators, r (at most $\log p$ bits long), $q_1 = 2$ and its certificate 1 (at most 5 bits), the q_i's (at most $2\log p$ bits), and the $C(q_i)$s.

The Proof (continued)

- $C(p)$ is succinct because

$$\begin{aligned}
|C(p)| &\leq 4\log_2 p + 5 + 4\sum_{i=2}^{k} \log_2^2 q_i \\
&\leq 4\log_2 p + 5 + 4(\sum_{i=2}^{k} \log_2 q_i)^2 \\
&\leq 4\log_2 p + 5 + 4\log_2^2 \frac{p-1}{2} \\
&< 4\log_2 p + 5 + 4(\log_2 p - 1)^2 \\
&= 4\log_2^2 p + 9 - 4\log p \leq 4\log^2 p
\end{aligned}$$

for $p \geq 5$.

Basic Modular Arithmetics[a]

- Let $m, n \in \mathbb{Z}^+$.
- $m|n$ means m divides n and m is n's **divisor**.
- We call the numbers $0, 1, \ldots, n-1$ the **residue** modulo n.
- The **greatest common divisor** of m and n is denoted $\gcd(m, n)$.
- The r in Theorem 42 (p. 251) is a primitive root of p.
- We now prove the existence of primitive roots and then Theorem 42.

[a] Carl Friedrich Gauss (1777–1855).

Euler's[a] Totient or Phi Function

- Let

$$\Phi(n) = \{m : 1 \leq m < n, \gcd(m, n) = 1\}$$

  be the set of all positive integers less than n that are prime to n.

  – $\Phi(12) = \{1, 5, 7, 11\}$.

- Define **Euler's function** of n to be $\phi(n) = |\Phi(n)|$.

- $\phi(p) = p - 1$ for prime p, and $\phi(1) = 1$ by convention.

- Euler's function is not expected to be easy to compute without knowing n's factorization.

[a] Leonhard Euler (1707–1783).

Two Properties of Euler's Function

Lemma 45 $\phi(n) = n \prod_{p|n} (1 - \frac{1}{p})$.

- Use the inclusion-exclusion principle (see my *Discrete Mathematics* lecture notes).

Corollary 46 $\phi(mn) = \phi(m)\phi(n)$ *if* $\gcd(m, n) = 1$.

- A number k is relatively prime to mn if and only if $\gcd(k, m) = 1$ and $\gcd(k, n) = 1$.

A Key Lemma

Lemma 47 $\sum_{m|n} \phi(m) = n$.

- Let $\prod_{i=1}^{\ell} p_i^{k_i}$ be the prime factorization of n and consider

$$\prod_{i=1}^{\ell} [\, \phi(1) + \phi(p_i) + \cdots + \phi(p_i^{k_i}) \,]. \tag{2}$$

- Eq. (2) equals n because $\phi(p_i^k) = p_i^k - p_i^{k-1}$ by Lemma 45.
- Expand Eq (2) to yield $\sum_{k_1' \le k_1, \ldots, k_\ell' \le k_\ell} \prod_{i=1}^{\ell} \phi(p_i^{k_i'})$.

Some Properties of Euler's Function (continued)

- By Corollary 46 (p. 258),

$$\prod_{i=1}^{\ell} \phi(p_i^{k_i'}) = \phi(\prod_{i=1}^{\ell} p_i^{k_i'}).$$

- Each $\prod_{i=1}^{\ell} p_i^{k_i'}$ is a unique divisor of $n = \prod_{i=1}^{\ell} p_i^{k_i}$.
- Eq. (2) becomes

$$\sum_{m|n} \phi(m).$$

The Chinese Remainder Theorem

- Let $n = n_1 n_2 \cdots n_k$, where n_i are pairwise relatively prime.
- For any integers $a_1, a_2, \ldots, a_k$, the set of simultaneous equations

$$\begin{aligned} x &= a_1 \bmod n_1 \\ x &= a_2 \bmod n_2 \\ &\vdots \\ x &= a_k \bmod n_k \end{aligned}$$

has a unique solution modulo n for the unknown x.

Fermat's[a] "Little" Theorem

Lemma 48 *For all* $0 < a < p$, $a^{p-1} = 1 \bmod p$.

- Consider $a\Phi(p) = \{am \bmod p : m \in \Phi(p)\}$.
- $a\Phi(p) = \Phi(p)$.
  - Suppose $am = am' \bmod p$ for $m > m'$, where $m, m' \in \Phi(p)$.
  - That means $a(m - m') = 0 \bmod p$, and p divides a or $m - m'$, which is impossible.
- Hence $(p-1)! = a^{p-1}(p-1)! \bmod p$.
- Finally, $(a^{p-1} - 1) = 0 \bmod p$ because $p \not| (p-1)!$.

[a] Pierre de Fermat (1601–1665).

The Fermat-Euler Theorem

Corollary 49 *For all* $a \in \Phi(n)$, $a^{\phi(n)} = 1 \bmod n$.

- As $12 = 2^2 \times 3$,

$$\phi(12) = 2^1 \times (2-1) + 3^0 \times (3-1) = 4.$$

- In fact, $\Phi(12) = \{1, 5, 7, 11\}$.
- For example,

$$5^4 = 625 = 1 \bmod 12.$$

Exponents

- The **exponent** of $m \in \Phi(p)$ is the least $k \in \mathbb{Z}^+$ such that

$$m^k = 1 \bmod p.$$

- Every residue $s \in \Phi(p)$ has an exponent.
  - $1, s, s^2, s^3, \ldots$ eventually repeats itself, say $s^i = s^j \bmod p$, which means $s^{j-i} = 1 \bmod p$.
- If the exponent of m is k and $m^\ell = 1 \bmod p$, then $k|\ell$.
  - Otherwise, $\ell = qk + a$ for $0 < a < k$, and $m^\ell = m^{qk+a} = m^a = 1 \bmod p$, a contradiction.

Lemma 50 *Any nonzero polynomial of degree k has at most k distinct roots modulo p.*

Exponents and Primitive Roots

- From Fermat's "little" theorem, all exponents divide $p-1$.
- A primitive root of p is thus a number of exponent $p-1$.
- Let $R(k)$ denote the total number of residues in $\Phi(p)$ that have exponent k.
- We already knew that $R(k) = 0$ for $k \not| p-1$.
- Any $a \in \Phi(p)$ of exponent k satisfies $x^k = 1 \bmod p$.
- Hence there are at most k residues of exponent k, i.e., $R(k) \leq k$.

The Number of Residues of Exponent k: $R(k)$

- Let s be a residue of exponent k.
- $1, s, s^2, \ldots, s^{k-1}$ are all distinct modulo p.
  - Otherwise, $s^i = s^j \bmod p$ with $i < j$ and s is of exponent $j - i < k$.
- As all these k distinct numbers satisfy $x^k = 1 \bmod p$, they are all the solutions of $x^k = 1 \bmod p$.
- But not all of them have exponent k.
- Suppose $\ell < k$ and $\ell \notin \Phi(k)$ with $\gcd(\ell, k) = d$.
- Then $(s^\ell)^{k/d} = 1 \bmod p$ and s^ℓ has exponent of $\leq k/d$.
- Hence $R(k) \leq \phi(k)$.

Size of $R(k)$ (continued)

- Because all $p-1$ residues have an exponent,

$$p-1 = \sum_{k|(p-1)} R(k) \leq \sum_{k|(p-1)} \phi(k) = p-1$$

  by Lemma 46 on p. 258.

- Hence

$$R(k) = \begin{cases} \phi(k) & \text{when } k|p-1 \\ 0 & \text{otherwise} \end{cases}$$

- In particular, $R(p-1) = \phi(p-1) > 0$, and p has at least one primitive root.

- This proves one direction of Theorem 42 (p. 251).

A Few Calculations

- Let $p = 13$.
- From p. 263, we know $\phi(p-1) = 4$.
- Hence $R(4) = 4$.
- And there are 4 primitives roots of p.
- As $\Phi(p-1) = \{1, 5, 7, 11\}$, the primitive roots are g^1, g^5, g^7, g^{11} for any primitive root g.

The Other Direction of Theorem 42 (p. 251)

- Suppose that p is not a prime and we proceed to show that no primitive roots exist.
- Suppose that $r^{p-1} = 1 \bmod p$.
- $r^{\phi(p)} = 1 \bmod p$ by the Fermat-Euler theorem (p. 263).
- Because p is not prime, $\phi(p) < p - 1$.
- Let k be the smallest integer such that $r^k = 1 \bmod p$.
- As $k|p-1$ and $k|\phi(p)$, $k < p - 1$.
- Let q be a prime divisor of $(p-1)/k > 1$.
- Then $r^{(p-1)/q} = 1 \bmod p$, violating the 2nd condition of the primitive root on p. 251.

Randomized Algorithms[a]

- Randomized algorithms are algorithms that flip unbiased coins.
- There are important problems for which there are no known efficient *deterministic* algorithms but for which very efficient algorithms exist.
  - Primality tests, extraction of square roots, etc.
- There are problems where randomization is *necessary*.
  - Secure protocols.

[a] Rabin, 1976, Solovay, Strassen, 1977.

Bipartite Perfect Matching

- We are given a **bipartite graph** $G = (U, V, E)$.
  - $U = \{u_1, u_2, \ldots, u_n\}$.
  - $V = \{v_1, v_2, \ldots, v_n\}$.
  - $E \subseteq U \times V$.
- We are asked if there is a **perfect matching**.
  - A permutation π of $\{1, 2, \ldots, n\}$ such that

$$(u_i, v_{\pi(i)}) \in E$$

    for all $u_i \in U$.

Symbolic Determinants

- Given a bipartite graph G, construct the $n \times n$ matrix A^G whose (i,j)th entry A^G_{ij} is a variable x_{ij} if $(u_i, v_j) \in E$ and zero otherwise.
- The **determinant** of A^G is

$$\det(A^G) = \sum_{\pi} \sigma(\pi) \prod_{i=1}^{n} A^G_{i,\pi(i)},$$

where π ranges over all permutations of n elements and $\sigma(\pi)$ is 1 if π is the product of an even number of transpositions and -1 otherwise.

Determinants and Bipartite Perfect Matching

In

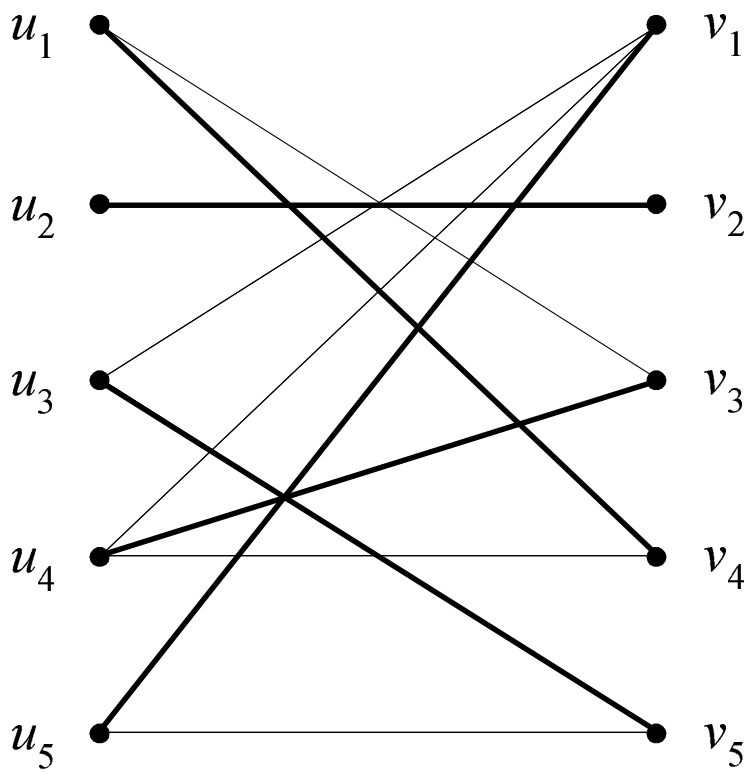
$$\sum_{\pi} \sigma(\pi) \prod_{i=1}^{n} A^{G}_{i,\pi(i)},$$

note the following:

- Each summand corresponds to a prefect matching π.
- As all variables appear only *once*, all of these summands are different monomials and will not cancel.

Proposition 51 *G has a perfect matching if and only if* $\det(A^G)$ *is not identically zero.*

A Bipartite Graph



A Perfect Matching

- The matrix is

$$A^G = \begin{bmatrix} 0 & 0 & x_{13} & \boxed{x_{14}} & 0 \\ 0 & \boxed{x_{22}} & 0 & 0 & 0 \\ x_{31} & 0 & 0 & 0 & \boxed{x_{35}} \\ x_{41} & 0 & \boxed{x_{43}} & x_{44} & 0 \\ \boxed{x_{51}} & 0 & 0 & 0 & x_{55} \end{bmatrix}.$$

- $\det(A^G)$ contains term $x_{14}x_{22}x_{35}x_{43}x_{51}$, which denotes a perfect matching.

How To Test If a Polynomial Is Identically Zero?

- $\det(A^G)$ is a polynomial in n^2 variables.
- There are exponentially many terms in $\det(A^G)$.
- Expanding the determinant polynomial is not feasible.
  - Too many terms.
- Observation: If $\det(A^G)$ is identically zero, then it remains zero if we substitute *arbitrary* integers for the variables $x_{11}, \ldots, x_{nn}$.
- But is the likelihood of obtaining a zero when $\det(A^G)$ is not *identically zero*?

Number of Roots of a Polynomials

Lemma 52 (Schwartz, 1980) *Let* $p(x_1, x_2, \ldots, x_m) \not\equiv 0$ *be a polynomial in* m *variables each of degree at most* d*. Let* $M \in \mathbb{Z}^+$*. Then the number of* m*-tuples* $(x_1, x_2, \ldots, x_m) \in \{0, 1, \ldots, M-1\}^m$ *such that* $p(x_1, x_2, \ldots, x_m) = 0$ *is*

$$\leq mdM^{m-1}.$$

- By induction on m.

Density Attack

- The density of roots in the domain is at most

$$\frac{mdM^{m-1}}{M^m} = \frac{md}{M}.$$

- This suggests a sampling algorithm.

A Randomized Bipartite Perfect Matching Algorithm

1: Choose n^2 integers $i_{11}, \ldots, i_{nn}$ from $\{0, 1, \ldots, 2n^2 - 1\}$ randomly;
1: Calculate $\det(A^G(i_{11}, \ldots, i_{nn}))$ by Gaussian elimination;
2: **if** $\det(A^G(i_{11}, \ldots, i_{nn})) \neq 0$ **then**
3: **return** "G has a perfect matching";
4: **else**
5: **return** "G probably has no perfect matchings";
6: **end if**

Analysis

- If G has no perfect matchings, the algorithm will always be correct.
- Suppose that G has a perfect matching.
  - The algorithm will answer incorrectly with probability at most $n^2 d/(2n^2) = 0.5$ because $d = 1$.
  - By repeating the algorithm *independently* k times, we can bring down the error probability to 2^{-k}.
    - The algorithm outputs "G has no perfect matchings" if and only if all of the k runs say so.