Outline

- The complexity class coNP
- The complexity class coNP \cap NP
- The complexity class PSPACE
  - a PSPACE-complete problem
  - PSPACE and 2-player games

coNP

- language L is in coNP iff its complement (co-L) is in NP
- it is believed that NP \neq coNP
- note: P = NP implies NP = coNP
  - proving NP \neq coNP would prove P \neq NP
  - another major open problem...

Quantifier characterization of coNP

- another example
  3-DNF-TAUTOLEGY = \{\phi : \phi is a 3-DNF formula and for all x, \phi(x) = 1\}
  - proof?
- another example:
  EQUIV-CIRCUIT = \{(C_1, C_2) : C_1 and C_2 are Boolean circuits and for all x, C_1(x) = C_2(x)\}
  - proof?

Theorem: language L is in coNP if and only if it is expressible as:
\[ L = \{ x : \forall y, |y| \leq |x|^k, (x, y) \in R \} \]
where R is a language in P.

Disjunctive Normal Form = OR of ANDs

March 1, 2021
Proof interpretation of coNP

- What is a proof?
- Good formalization comes from NP:
  \[ L = \{ x \mid \exists y, \ |y| \leq |x|^{n}, \ (x, y) \in R \} \text{, and } \text{Re} = \text{P} \]
- "proof" "short" proof "proof verifier"
- NP languages have short proofs of membership
- co-NP languages have short proofs of non-membership
- coNP-complete languages are least likely to have short proofs of membership

coNP

- what complexity class do the following languages belong in?
  - COMPOSITES = \{ x : \text{integer } x \text{ is a composite} \}
  - PRIMES = \{ x : \text{integer } x \text{ is a prime number} \}
  - GRAPH-NONISOMORPHISM = \{(G, H) : G \text{ and } H \text{ are graphs that are not isomorphic} \}
  - EXPANSION = \{(G = (V,E), \alpha > 0) : \text{every subset } S \subseteq V \text{ of size at most } |V|/2 \text{ has at least } \alpha |S| \text{ neighbors} \}

NP \cap \text{coNP}

- Might guess \( NP \cap \text{coNP} = P \) by analogy with \( \text{RE} \) (since \( \text{RE} \cap \text{coRE} = \text{DECIDABLE} \))
- Not believed to be true.
- A problem in \( NP \cap \text{coNP} \) not believed to be in P:
  \[ L = \{ (x, k) : \text{integer } x \text{ has a prime factor } p < k \} \]
  (decision version of factoring)

PRIMES in NP

**Theorem:** (Pratt 1975) PRIMES is in NP.
PRIMES = \{ x : \forall 1 < y < x, \ y \text{ does not divide } x \}

- Proof outline:
  - Step 1: give "3" characterization of PRIMES
  - Step 2: this \( \Rightarrow \) short certificate of primality
  - Step 3: certificate checkable in poly time (we will skip, because…)

**Theorem:** (M. Agrawal, N. Kayal, N. Saxena 2002) PRIMES is in P.
Summary

- Picture of the way we believe things are:

\[
\begin{align*}
\text{P} & \quad \text{NP} \cap \text{coNP} \\
\text{NP} & \quad \text{EXP} \\
\text{coNP} & \quad \text{(decision version of ) FACTORING} \\
\text{decidable languages} &
\end{align*}
\]