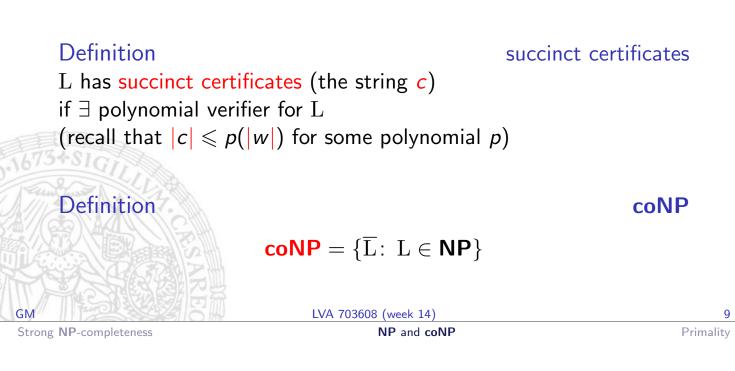


NP and coNP

- Definition polynomial verifier
 A verifier of a language L is an algorithm P such that:
 - $L = \{w \mid \text{there exists a string } c \text{ so that } P \text{ accepts } \langle w, c \rangle \}$
 - A polynomial verifier is one that runs in time polynomial in |w|



Observation

- suppose $L \in \mathbf{coNP}$, and a string x such that $x \not\in L$
- then $x \in \overline{L}$, which implies the existence of a succinct certificate c
- hence the "no"-instance x has a succinct disqualification

Example

VALIDITY = { φ : φ is a valid CNF-formula}

1 if φ is not valid, then the disqualification is an assignment T, such that $T(\varphi) =$ **false**

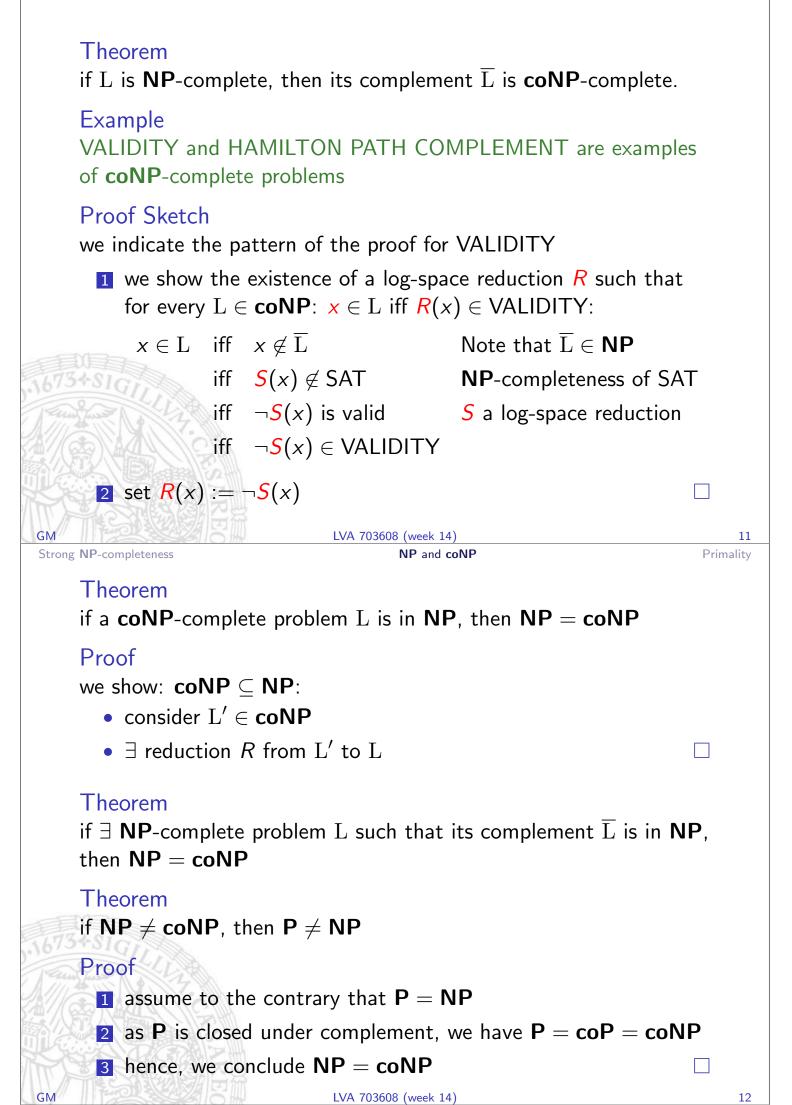
2 the disqualification is succinct, i.e. it is at most polynomial in the length of the formula

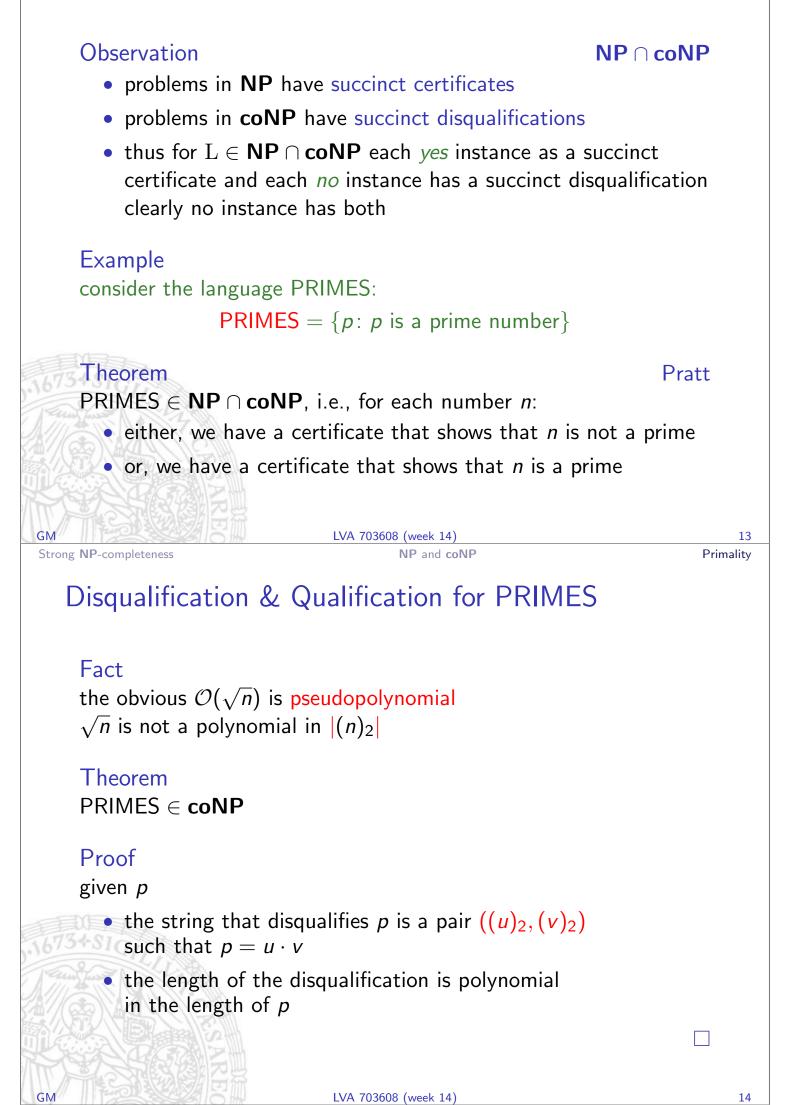
Example

GM

HAMILTON PATH COMPLEMENT =

 $\{G: G \text{ is a directed graph without Hamilton path}\}$





GM

Theorem A number p > 1 is prime iff there is a number $r \in \{2, \ldots, p-1\}$ such that $r^{p-1} = 1 \mod p$, and $r^{\frac{p-1}{q}} \neq 1 \mod p$ for all prime divisors q of p - 1; r is called primitive root Proof Idea employ Fermat's (small) Theorem for all $r \in \{1, \dots, p-1\}$: $r^{p-1} = 1 \mod p$ Theorem PRIMES ∈ **NP** Proof the certificate consists of 1 the primitive root r **2** the prime divisors q_1, \ldots, q_k **3** primality certificates for q_i GM LVA 703608 (week 14) 15 Primality Strong NP-completeness NP and coNP Nondeterministic Algorithm given p (in binary) 1 if p = 2, accept; if p > 2 and p even, reject **2** guess prime factorisation of $p - 1 = q_1^{k_1} \cdots q_m^{k_m}$ verify by multiplication **3** guess $r \in \{2, \ldots, p-1\}$ and verify that $r^{p-1} = 1 \mod p$ 4 verify for each *i*: $r^{\frac{p-1}{q_i}} \neq 1 \mod p$ **5** recursively verify that q_1, \ldots, q_m are prime Observation step 1 is constant; step 2 polynomial in log p; steps 3 & 4 can be

performed in polytime (in $\log n$) by repeatedly squaring; solving the recursion yields a polytime algorithm

