The class NP:

A language is in NP if and only if it is decided by some non-deterministic polynomial time Turing machine.

A verifier for a language L is an algorithm V, where

L = { w | V accepts <w, c> for some string c (certificate) }

A polynomial time verifier, is a verifier that runs in polynomial time in the length of w.

NP - class of languages that have polynomial time verifiers. ("Easy to check")

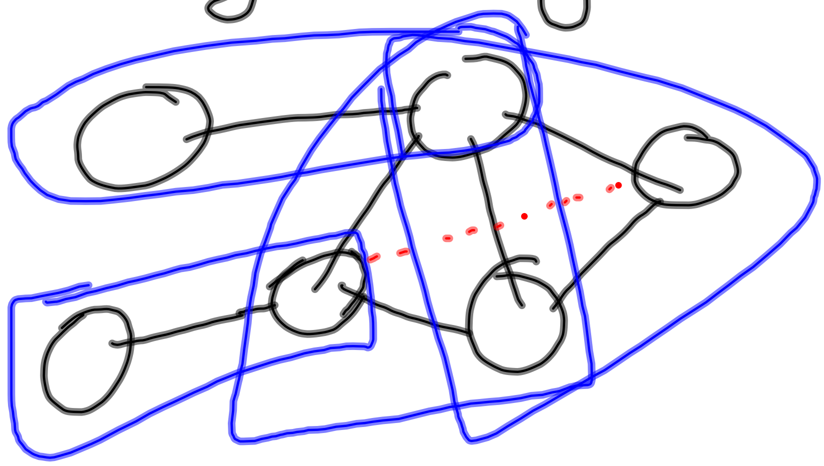
HAMPATH verifier V

$$V \langle \underbrace{\langle G, s, k \rangle}_{\text{checking}}, P \rangle$$

possible path

Is P a Ham. path in G that starts at s and goes to k?

CLIQUE - in an undir. graph is a subgraph, where every two nodes are connected by an edge.



CLIQUE = $\{\langle G, k\rangle \mid$ G is an undirected graph w/ a clique of size k $\}$
(k-clique)

CLIQUE is in NP

2 methods of proof

I. give a polynomial time verifier.

II. give a non-det. polynomial time decider.

I. Construct a verifier V w/ a clique c as the certificate.

V = on input $\langle \langle G, k\rangle, c\rangle$

$O(k)$ 1. Test if c is a set of k-nodes

$O(k^2)$ 2. Test if G contains all edges connecting nodes in c.

3. If both tests pass, accept otherwise, reject

II. construct a Non det. TM that runs in polynomial time that decides CLIQUE

T = on input <G, k> where
G is an undirected Graph and
k is a number.

1. non-deterministically select a subset c of k nodes in G.

2. Test whether G contains all edges connecting nodes in c.

3. if yes accept, if no reject.

SUBSET_SUM = $\{ \langle S, t \rangle \mid$

$S = \{ x_1, \dots x_k \}$ and for

some $\{ y_1, \dots, y_\ell \} \subseteq S$

we have $\sum_{i=1}^{\ell} y_i = t \}$

---

power set $\mathcal{P}(S)$ or 2^S

I. Construct a verifier

V on input $\langle \langle S, t \rangle, c \rangle$

1. Test if $c \subseteq S$.
2. Test if c sums to t.
3. if both pass, accept otherwise, reject.