NP = VP

Presented by Ruth bat Yisrael

DTM and NTM

In essence, a Turing machine is imagined to read and write symbols one at a time on an endless tape by strictly following a set of rules. It determines what action it should perform next according to its internal state and what symbol it currently sees.

A Deterministic (Vanilla) Turing machine has no conflicting rules.

A Non-Deterministic Turing machine has (finitely many) conflicting rules.

Interpretation of NTM

- Multi-Universe Interpretation
- Lucky-Dog Interpretation

Important Notes of NTM

- Time complexity?
- What does it mean that an NTM accepts an input.

Definition of NP & VP

NP: the set of decision problems solvable in polynomial time by an NTM.

VP: the set of decision problems verifiable in polynomial time by a DTM.

NP in VP

- For L in NP, there exists a polynomial NTM M such that M non-deterministically solves L.
- We construct a verifier c as the steps that L takes to solve L.
- Use a DTM M' to simulate M with respect to c.

VP in NP

- For L in VP and A is the verifier of L(what is A). Construct an NTM as follows:
  - Input: x
  - Initialization: c as an empty string
  - Rules: at each step, append 0 or 1 to the end of c, or output A(x, c)

Correct Proof of VP in NP

- Input: x
- Non-deterministically generates a proof c
- Simulate A(x, c), accept x if A(x, c) and reject x otherwise

Bonus

- How to non-deterministically generate a string of (possibly) unfixed length with an NTM