CS 218 Design and Analysis of Algorithms

Nutan Limaye

Indian Institute of Technology, Bombay
nutan@cse.iitb.ac.in

Module 3: NP hardness and reductions

Polynomial time reductions and NP-hardness

Understanding hard problems.

We will explore the space of hard problems.

In order to eventually arrive at a mathematical theory about them.

Main technique in this exploration: understand the relative strength of hard problems.

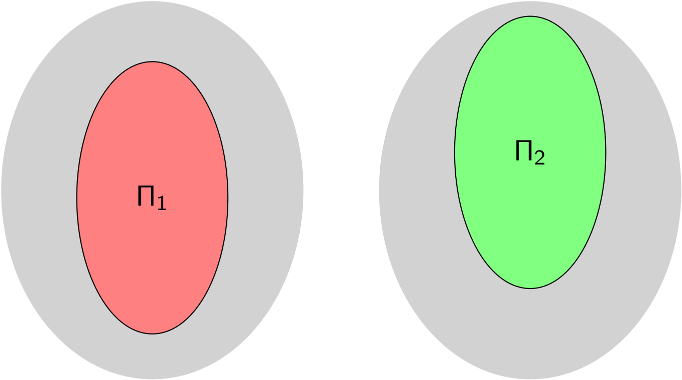
We would like to say things like "Problem Π is at least as hard as problem Π'"

We will formalise this notion using reductions.

Polynomial time reductions and NP-hardness

Definition

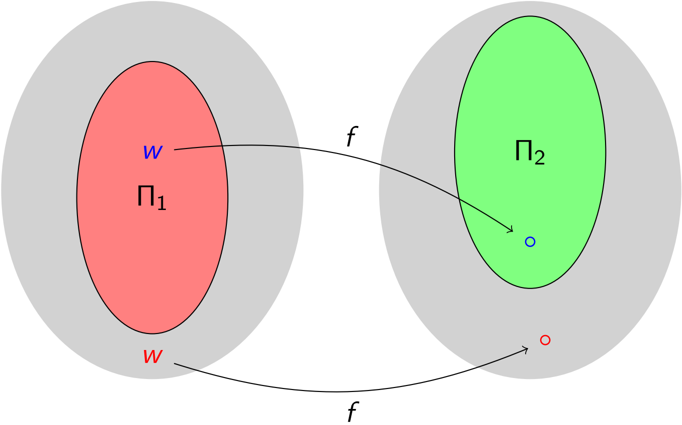
A problem Π_1 is said to be polynomial time reducible to another problem Π_2, denoted as $\Pi_1 \leq_m \Pi_2$, if there exists a polynomial time computable function f such that for all inputs w, $w \in \Pi_1 \Leftrightarrow f(w) \in \Pi_2$.



Polynomial time reductions and NP-hardness

Definition

A problem Π_1 is said to be polynomial time reducible to another problem Π_2, denoted as $\Pi_1 \leq_m \Pi_2$, if there exists a polynomial time computable function f such that for all inputs w, $w \in \Pi_1 \Leftrightarrow f(w) \in \Pi_2$.



Polynomial time reductions and NP-hardness

Definition

A problem Π is said to be NP-hard if for every problem $\Pi' \in$ NP, there is a polynomial time reduction such that $\Pi' \leq_m \Pi$.

Definition

A problem Π is said to be NP-complete if the following two conditions hold:

- Π is in NP.
- Π is NP-hard.

Theorem ([Cook-Levin, 1970])

SAT is NP-complete.