Polynomial Time Reduction

- Polynomial Time Reduction Definition
- Reduction by Equivalence
- Reduction from Special Cases to General Case
- Reduction by Encoding with Gadgets
- Transitivity of Reductions
- Decision, Search and Optimization Problem
- Self-Reducibility

Imdad ullah Khan

Polynomial time reduction is a way to compare hardness of problems

Problem A is polynomial time reducible to Problem B, $A \leq_p B$

If any instance of problem A can be solved using a polynomial amount of computation plus a polynomial number of calls to a solution of problem B

If any algorithm for problem B can be used [called (once or more) with *'clever'* legal inputs] to solve any instance of problem A

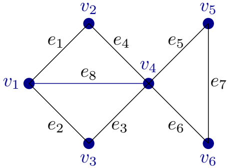
Subroutine for B takes an instance y of B and returns the solution $B(y)$

Algorithm for A transforms an instance x of A to an instance y of B. Then transforms $B(y)$ to $A(x)$

Reduction from special case to general case

$$\text{VERTEX-COVER}(G, k) \quad \leq_p \quad \text{SET-COVER}(U, \mathcal{S}, k')$$

- Let $\mathcal{A}$ be an algorithm solving SET-COVER$(U, \mathcal{S}, k')$
- Let $[G, k]$ be an instance of the VERTEX-COVER problem
- Make $U = E$, $\qquad k' = k$
- $\mathcal{S} = \{S_1, \dots, S_n\}$, where $S_i = \{e \in E \mid e \text{ is incident on } v_i\}$



$U = \{e_1, e_2, e_2, e_4, e_5, e_6, e_7, e_8\}$

$S_1 = \{e_1, e_2, e_8\}$
$S_3 = \{e_2, e_3\}$
$S_2 = \{e_1, e_4\}$
$S_4 = \{e_3, e_4, e_5, e_6, e_8\}$
$S_5 = \{e_5, e_7\}$
$S_6 = \{e_6, e_7\}$

Theorem: $[U, \mathcal{S}]$ has a set cover of size k iff G has a vertex cover of size k

- If $\mathcal{A}(U, \mathcal{S}, k') = $ **Yes**, then output **Yes**, else output **No**

Reduction from special case to general case

We get the following reduction very similarly

$$\text{INDEPENDENT-SET}(G,k) \quad \leq_p \quad \text{SET-PACKING}(U,\mathcal{S},k')$$

The following reductions are even more straight forward. They follow from respective definitions of the problems

$$\text{3-SAT}(f) \quad \leq_p \quad \text{SAT}(f')$$

$$\text{SUBSET-SUM}(U,w,C) \quad \leq_p \quad \text{KNAPSACK}(U,w,v,C')$$

Please complete their details. Explicitly and formally writing them will help understand the important notion of reduction