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

We used the following techniques for reduction

- Simple Equivalence
- Special Case to General Case
- Encoding with Gadgets

A very powerful technique is to exploit transitivity of reductions

Theorem: If $X \leq_p Y$ and $Y \leq_p Z$, then $X \leq_p Z$

$\triangleright$ **Polynomial time reduction is a transitive relation**

Polynomial time reduction is a transitive relation

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

Theorem: If $X \leq_p Y$ and $Y \leq_p Z$, then $X \leq_p Z$

Proof: Let $\mathcal{A}_Z$ be an algorithm for Z

- Given any instance I_X of X we will solve X on I_X using $\mathcal{A}_Z^+$
  - There is an algorithm $\mathcal{A}_Y$ for Y using $\mathcal{A}_Z^+$ (maybe many others too)
  - There is an algorithm $\mathcal{A}_X$ for X using $\mathcal{A}_Y^+$
- $\mathcal{B}_{\mathcal{X}}$: the new algorithm for X uses everything as of $\mathcal{A}_X$ but it uses the specific algorithm $\mathcal{A}_Y$ that is built upon $\mathcal{A}_Z$

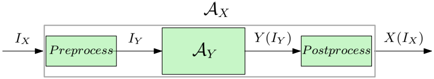
$\triangleright$ We essentially compose the two reductions into one

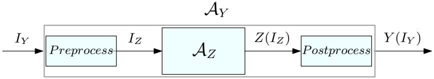
Polynomial time reduction is a transitive relation

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

Theorem: If $X \leq_p Y$ and $Y \leq_p Z$, then $X \leq_p Z$



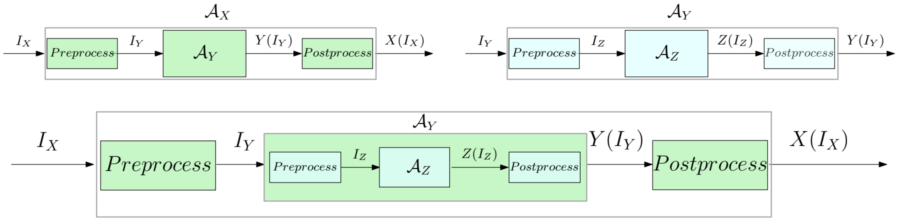


Polynomial time reduction is a transitive relation

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

Theorem: If $X \leq_p Y$ and $Y \leq_p Z$, then $X \leq_p Z$



Polynomial time reduction is a transitive relation

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

Theorem: If $X \leq_p Y$ and $Y \leq_p Z$, then $X \leq_p Z$

Transitivity is an extremely useful property of reduction

- $\text{SAT}(f) \leq_p \text{3-SAT}(f')$ and $\text{3-SAT}(f) \leq_p \text{IND-SET}(G, k)$
  - From these we conclude that $\text{SAT}(f) \leq_p \text{IND-SET}(G, k)$
- $\text{SAT}(f) \leq_p \text{3-SAT}(f') \leq_p \text{IND-SET}(G, k) \leq_p \text{VERTEX-COVER}(G, t) \leq_p \text{SET-COVER}(U, \mathcal{S}, l)$
  - From these we conclude that $\text{SAT}(f) \leq_p \text{SET-COVER}(U, \mathcal{S}, l)$
- And many others