

* Why Approximation Algorithms?

Many problems in Computer Science are 'HARD' to solve EXACTLY

HARD - No Polytime Alg.

* Polynomial Time Algorithm

- No. of instructions executed by the algorithm can be bounded by a polynomial in size of i/p.

- Decision Version of a Problem

Possible Answer: 'Yes' or 'No'

- Complexity Classes

P - All decision problems with polynomial time algorithm

NP - (Non Deterministic Polynomial Time)

Answer - Yes
 \Rightarrow 'Short', Easily Verifiable Proof.

Answer - No
 \Rightarrow No 'Short' Proof convincing.

NP-Complete

Polynomial Time Reduction

INSTANCE of A \rightarrow \boxed{P} \rightarrow Instance of B

Yes of B \Leftrightarrow Yes of A

$A \leq_p B$

B is NP-complete if

i) B is in NP

ii) For Every A in NP

$A \leq_p B$

IMPORTANCE?

We want FAST, CHEAP \exists RELIABLE solution.

But can choose only 2 of 3

- i) Find Optimal Sol
- ii) In Poly Time
- iii) For All Instances

APPROXIMATION ALGORITHM:

An α -approximation algorithm for an optimization problem is a poly-time algorithm that for all instances of the problem, produces a solution within a factor α of the OPTIMAL solⁿ.

For MAXIMIZATION PROBLEM
 α ?

For MINIMIZATION PROBLEM
 α ?