Analysis of Algorithms - CS 323
Lecture #13 - May 11, 2016

Notes by: Natan Sachakov

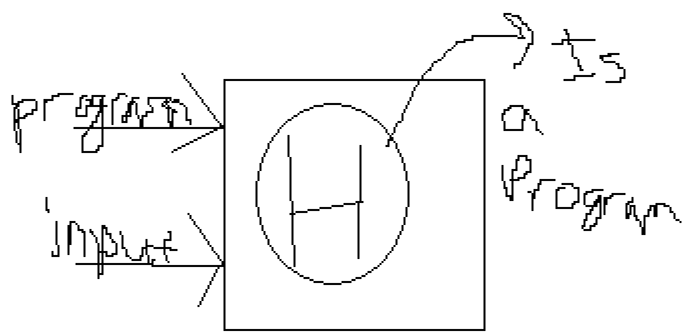
Limits of Computation / NP-Completeness

- Unsolvable problems (undecidable)
- Intractible problems (infeasible)

Undecidable problem examples

<u>Diophantine Equation Problem</u> - Given a polynomial expression with a given degree and coefficient. Find the expression P(x1,x2,...,xn)=0.

<u>Halting Problem</u> - Is there a general program, that when given a program as an input and an input to the program, will say if the input program for the given input will hault or not.



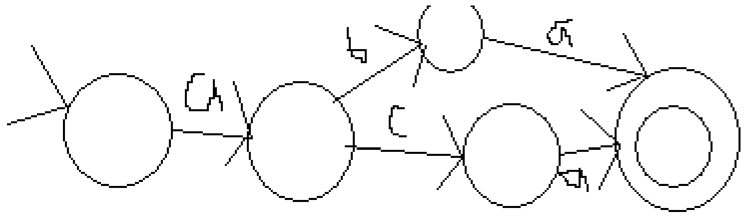
Intractible

An <u>Intractible</u> problem takes too long to solve to be useful. These problems take beyond polynomial time to solve.

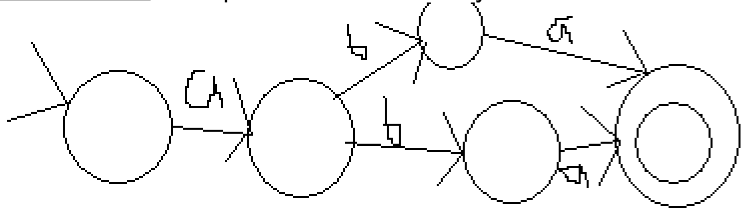
polynomials are closed under addition, multiplication and composition. This means that if you take a polynomial and add, multiply or compose it with another that the output will also be a polynomial.

Classes of Intractible problems

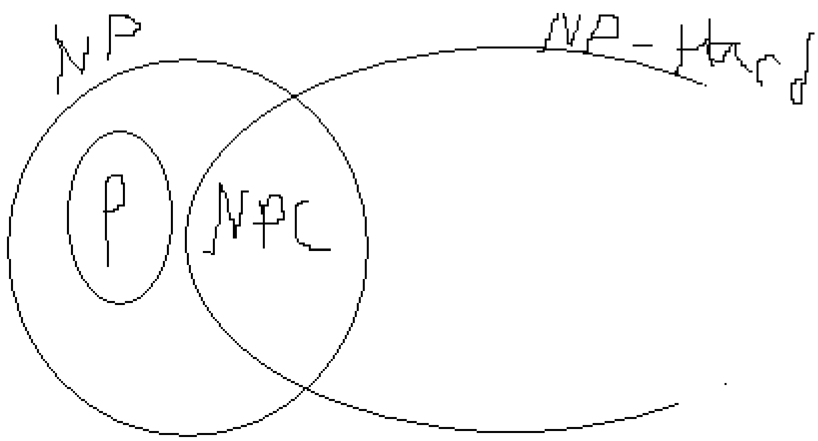
<u>Determinism</u> - Predictable. Only one choice.



<u>Non-Determinism</u> - Not predictable. Has many choices.



<u>P</u> - Class of problems that can be solved in deterministic polynomial time.
<u>NP</u> - Class of problems that can be solved in non-deterministic polynomial time.
<u>NPC</u> (NP-Complete)
<u>NP-Hard</u> - AT least as hard as anything in NPC. This class of problems includes undecidable problems.



Non-deterministic problems can be changed to deterministic problems(NP to P) but the scale changes. Twice as many inputs still makes the problem exponential as a deterministic problem.

Intractible problem examples

<u>Traveling SalesPerson(TSP)</u> -Visit every city in the cheapest way.

Optimization problem of TSP - What is the best route to take. Output is a value.
Decision problem of TSP - Does there exist a path that meets a constraint. Yes or no output.

<u>Hamiltonian Circuit Problem</u>(Exponential time complexity)

Visit eery vertex exactly once and return to the source vertex.

Reduction- Think of Hamiltonian problem as an instance of the TSP. If you can solve TSP in polynomial time then you can solve the Hamiltonian Circuit Problem in polynomial time. Assign all existing edge weights to 1. Is the total cost of the edge weights < n?

Polynomial Time Reduction - Should only take polynomial time to convert one problem into another problem.

Since it takes polynomial time to convert The Hamiltonian Circuit Problem to the TSP, and it takes polynomial time to solve the TSP. Therefore it takes polynomial time to solve the Hamiltonian Circuit Problem.

List of NPC problems

SAT
General 0-1 Knapsack
Hamiltonian Circuit
TSP
Bin-Packing (BP)
Graph Coloring
Subset-Sum

An NP solution can be verified in polynomial time. (The decision version of the problem only).

Example of NP-Hard problem that is more complex than an NPC problem is the True Quantified Boolean Formula. If and NP - Hard problem can not be reduced to an NPC problem than it is greater than an NP problem.

Approximation Algorithms(Heuristics)

These algorithms are not perfect but practical.

Optimality - how optimal is the solution?
Completeness - Does it give all solutions?
Accuracy - How wrong is the algorithm?
Margin of error:
c- heuristic solution
c*-optimal solution

absolute error bound - $|c-c^*| < e(n)$ (error grows with n)
ratio error bound - $c/c^* < e(n)$
relative error bound - $|c-c^*|/c^* < e(n)$

backtracking in an algorithm reduces search space. It optimizes code.