Approximation Algorithms

Approximation algorithms are algorithms designed to solve problems that are not solvable in polynomial time for approximate solutions. These problems are known as NP complete problems. These problems are significantly effective to solve real world problems, therefore, it becomes important to solve them using a different approach.

NP complete problems can still be solved in three cases: the input could be so small that the execution time is reduced, some problems can still be classified into problems that can be solved in polynomial time, or use approximation algorithms to find near-optima solutions for the problems. This leads to the concept of performance ratios of an approximation problem.

Performance Ratios

The main idea behind calculating the performance ratio of an approximation algorithm, which is also called as an approximation ratio, is to find how close the approximate solution is to the optimal solution.

The approximate ratio is represented using ρ(n) where n is the input size of the algorithm, C is the near-optimal solution obtained by the algorithm, C* is the optimal solution for the problem. The algorithm has an approximate ratio of ρ(n) if and only if –

$$max\left\{\frac{C}{C*},\frac{C*}{C}\right\}\leq\rho\left(n\right)$$

The algorithm is then called a ρ(n)-approximation algorithm. Approximation Algorithms can be applied on two types of optimization problems: minimization problems and maximization problems. If the optimal solution of the problem is to find the maximum cost, the problem is known as the maximization problem; and if the optimal solution of the problem is to find the minimum cost, then the problem is known as a minimization problem.

For maximization problems, the approximation ratio is calculated by C*/C since 0 ≤ C ≤ C*. For minimization problems, the approximation ratio is calculated by C/C* since 0 ≤ C* ≤ C.

Assuming that the costs of approximation algorithms are all positive, the performance ratio is well defined and will not be less than 1. If the value is 1, that means the approximate algorithm generates the exact optimal solution.

Examples

Few popular examples of the approximation algorithms are −

- Vertex Cover Problem
- Set Cover Problem
- Travelling Salesperson Problem
- The Subset Sum Problem