Lecture 13

Gidon Rosalki

2024-12-11

We will look at two types of problem: Maximisation and minimasation.

Definition 0.1. *Let there be S the solution space of some given algorithm problem, and $q : S \to \mathbb{R}^+$, a function that maps a solution to a value.*

*A **maximisation** problem searches $s \in S : q(s) = \max_{s \in S} \{q(s)\}$ A **minimisation** problem searches $s \in S : q(s) = \min_{s \in S} \{q(s)\}$*

Definition 0.2. *For $c > 1$ we will say that an algorithm that approaches c is a maximisation problem if for every input, the algorithm returns a legal solution s such that*

$$q(s) \geq \frac{q(s')}{c}$$

For $c > 1$ we will say that an algorithm that approaches c is a minimisation problem if for every input, the algorithm returns a legal solution s such that

$$q(s) \leq \frac{q(s')}{c}$$

Example 1 (Minimum set cover)**.** ***Input:*** *A natural number n, r subsets $A_1, \dots, A_r$ of $[n]$ such that*

$$\bigcup_{i \in [r]} A_i = [n]$$

Output: *Find a subset $S \subseteq [r]$ of minimal indices such that*

$$\bigcup_{i \in S} A_i = [n]$$

This problem is NP hard. So let's find an approximation algorithm.

Solution - Greedy.

1. **Initialisation:** $G = \emptyset \wedge X = [n]$
2. **Iteration:** Let $i^* \in [r]$ such that $|X \cap A_{i^*}| = \max_{i \in [r]} \{|A_i \cap X|\}$
3. We will add the sets to the collection $G = G \cup \{i^*\}$, we will remove the elements that have been included in the overall set $X = X \setminus A_{i^*}$
4. We will stop when all the elements have been picked: $X = \emptyset$

□