Problem Set 9

Jaan Tollander de Balsch

March 22, 2020

H9.1

Let ϕ be CNF formula with m clauses. We can formulate a greedy algorithm for deterministic 2-approximation for MAX SAT..

Create sorted formula ϕ'.

1) Count the amount of each literal in ϕ.
2) Sort literals in each clause from most frequent to the least frequent, that is, in decreasing order.
3) Lexicographically sort clauses in decreasing order.

Greedily assign truth values to the variables:

For all clauses in ϕ' or until all variables have been assigned a truth value

1) If clause is *satisfied* or *unsatisfied* move to the next clause.
2) If clause is *undetermined*, find the first literal with unassigned variable. Then assign a truth value for the variable such that the clause satisfied.

The sorting is required, otherwise given formula

$$l_1 \wedge l_2$$

we can build a worst case by adding clauses

$$\neg l_1 \wedge \neg l_2 \wedge (\neg l_1 \vee \neg l_2).$$

The greedy assignment would be $l_1 = l_2 = T$, then $\neg l_1 = \neg l_2 = (\neg l_1 \vee \neg l_2) = F$, thus the formula would not be 2-approximation.

Proof: Since the literals and clauses are sorted in decreasing order, we can only add two of the three clauses, otherwise the clauses would remain sorted. Therefore, the solution is 2-approximation in the worst case.

NOTE: (Full proof would require generalizing the example.)

H9.2

MIN SET COVER

Instance: A finite set U and a family $S = \{S_1, ..., S_m\}$ of subset of U.
Feasible solution: A subfamily $T \subseteq S$ such that their union is U.
Objective: Minimize $c(T) = |T|$.

Greedy approximation algorithm for MIN SET COVER.

Input: Set samily S over U.

Output: Set cover T.

1) Start from $T = \emptyset$
2) Find the set $S' \in S$ that covers most uncovered elements in U.
3) Add S' to T.
4) Repeat until all elements of U are covered.

For every integer $n = 2^m$ where $m \in \mathbb{N}$, there is an instance of MIN SET COVER such that

1) There are n elements in the base set.
2) The optimal cover uses two sets.
3) The greedy approximation algorithm picks $\log_2 n$ sets.

First we will define two functions

$$\text{even}(i) = \begin{cases} 1, & i \text{ is even} \\ 0, & \text{otherwise} \end{cases}, \quad \text{odd}(i) = \begin{cases} 1, & i \text{ is odd} \\ 0, & \text{otherwise} \end{cases}.$$

We define the size of base set as $|U| = n$. Then, lets partition U into two equal-sized, disjoint subsets S' and S''. Let $T = \{S', S''\}$ be the optimal cover with two sets.

Now, we can construct the instance $S = \{S_1, ..., S_m\}$ in which greedy algorithm picks all $m = \log_2 n$ sets as follows.

We begin with $K'_0 = S', K''_0 = S''$ and $T'_0 = \emptyset, T''_0 = \emptyset$. For all $i = 1, ..., m-2$ we have

1) $K'_i = K'_{i-1} \setminus T'_{i-1}, \quad T'_i \subseteq K'_i, \quad |T'_i| = |S'|/2^i + \text{even}(i)$
2) $K''_i = K''_{i-1} \setminus T''_{i-1}, \quad T''_i \subseteq K''_i, \quad |T''_i| = |S''|/2^i + \text{odd}(i)$
3) $S_i = T'_i \cup T''_i$

Finally, we define $S_{m-1} = S'$ and $S_m = S''$.

The idea of the construction is that the set $S_i = T_i' \cup T_i''$ for $i = 1, ..., m-2$ contains alternatingly half and half plus one of the elements that sets S' or S'' would cover if chosen.

Thus S_i always contains one more element that either set S' or S'' could cover, therefore greedy algorithm will always choose it.

H9.3

(a)

(b)

(c)