CS–E4500 Advanced Course in Algorithms

Week 03 – Tutorial

In a logical formula, a literal is either a Boolean variable or the negation of a Boolean variable. We use $\overline{x}$ to denote the negation of the variable x. A satisfiability (SAT) problem, or a SAT formula, is a logical expression that is the conjunction (AND) of a set of clauses, where each clause is the junction (OR) of literals. For example, the following expression is an instance of SAT:

$$(x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_3}) \wedge (x_1 \vee x_2 \vee x_4) \wedge (x_4 \vee \overline{x_3}) \wedge (x_4 \vee \overline{x_1}) \; .$$

A solution to an instance of a SAT formula is an assignment of the variables to the values True and False so that all the clauses are satisfied. That is, there is at least one true literal in each clause. For example, assigning x_1 to True, x_2 to False, x_3 to False, and x_4 to True satisfies the preceding SAT formula. In general, determining if a SAT formula has a solution is NP-hard.

A related goal, given a SAT formula, is satisfying as many of the clauses as possible. In what follows, let us assume that no clause contains both a variable and its complement, since in this case the clause is always satisfied.

1. Given a set of m clauses, let k_i be the number of literals in the ith clause for $i = 1, \ldots, m$. Let $k = \min_{i=1}^{m} k_i$. Then there is a truth assignment that satisfies at least $m(1 - 2^{-k})$ clauses.

2. Consider the following two-player game. The game begins with k tokens placed at the number 0 on the integer number line spanning $[0, n]$. Each round, one player, called the chooser, selects two disjoint and nonempty sets of tokens A and B. (The sets A and B need not cover all the remaining tokens; they only need to be disjoint.) The second player, called the remover, takes all the tokens from one of the sets off the board. The tokens from the other set all move up one space on the number line from their current position. The chooser wins if any token ever reaches n. The remover wins if the chooser finishes with one token that has not reached n.

   (a) Give a winning strategy for the chooser when $k \geq 2^n$.

   (b) Use the probabilistic method to show that there must exist a winning strategy for the remover when $k < 2^n$.