RWTH Aachen University **WS 2010/11**
Lehrgebiet Theoretische Informatik Tutorial 9
Kneis—Langer—Sikdar 14.12.2010

Tutorial Exact Algorithms

The problem k-XSAT is defined as:

Input:	A formula F in CNF, an integer k.
Question:	Is there an assignment to the variables of F, such that a) each clause is exactly satisfied, and b) exactly k variables are set to *true*

The problem k-Monotone-XSAT is the restriction of k-XSAT to *monotone* formulas, where each variable occurs positively only.

Exercise T17

Prove IS $\leq_p$ k-Monotone XSAT and k-XSAT $\leq_p$ k-Monotone-XSAT.

Solution

Let $d = \Delta(G)$. For a graph $G = (V, E)$ and $k \in \mathbf{N}$ we construct an instance F for k-Monotone XSAT as follows:

We have variables x_v for each node $v \in V$ and additionally variables y_v^i for $1 \leq i \leq d$ for each node $v \in V$.

Let $e = u, v$ be the $i - th$ incident edge to v and the $j - th$ incident edge to u. Then F contains the clause $\{x_v, x_u, y_v^i, y_i^j\}$. Moreover, for each node v, F contains the clauses $\{x_v, y_v^t\}$ for $\deg(v) < t \leq d$.

We claim that F has a exact satisfying assignment with at most $k + (|V| - k)d$ literals set to *true* iff G has an independent set of size k.

Let I be an independent set of size k in G. Then setting all x_v with $v \in I$ to *true* exactly satisfies $d|I|$ clauses, because no clause contains x_v and x_u with $u, v \in I$ and all clauses containing some x_u with $u \in I$ are satisfied. For the remaining clauses, we pick a y_v^i in this clause and set it to *true*. Since there are in total $|V|d$ clauses, we set $k + (|V| - k)d$ variables to *true* and obtain a satisfying assignment.

Now let X denote the variables set to true in an exact satisfying assignment A and let $X = k + (|V| - k)d$. Note that each x_v with $v \in V$ occurs in exactly d clauses and each y occurs in exactly one clause. As there are $|V|d$ clauses, X contains at least k variables x_v with $v \in V$. Let $I = \{v \in V \mid x_v \in X\}$. Since for each edge $e = \{u, v\} \in E$ there is a clause in F containing both x_v and x_u, and since A exactly satisfies F, I is an independent set. Hence, G contains an independent set of size k.

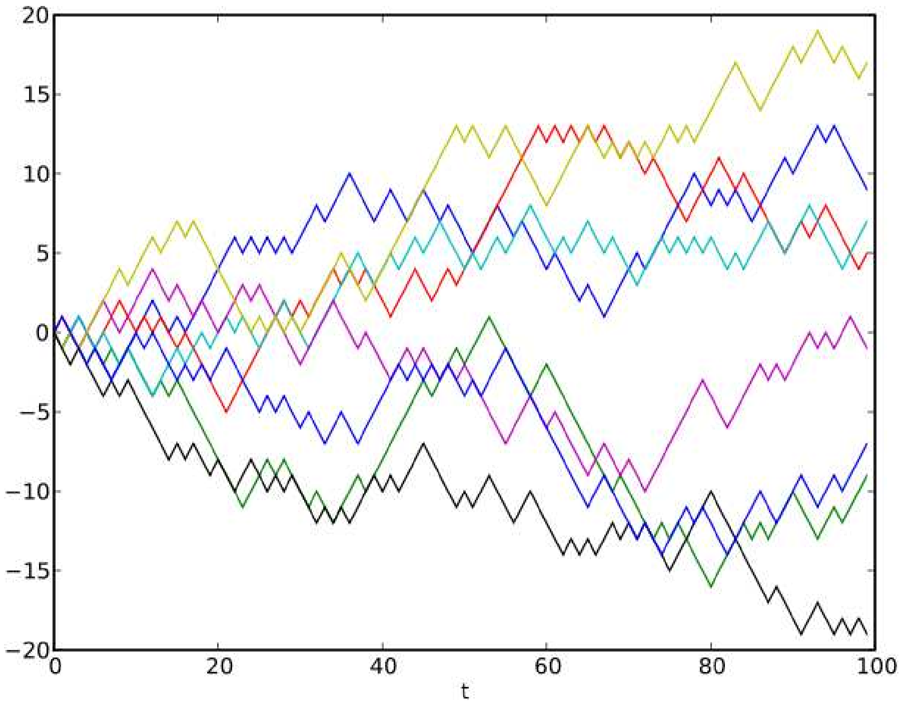
To proof k-XSAT $\leq_p$ k-Monotone-XSAT, we simply replace each variable x by two new variables x_t and x_f. Adding the clause $\{x_f, x_t\}$ guarantees that exactly one of these variables is set to *true*. Now, we simply replace $\overline{x}$ with x_f and x with x_t. Finally, we add the clauses $\{x_f, x_c\}$. Let F' be the new formula and F the old one.

Now, if x is set to *true* in F, we set x_t to true, but then are forced to set x_f to *false* and x_c to *true*. If x is set to *false*, we only set x_f to *true*. An assignment that sets k variables to *true* thus requires $n + k$ variables set to *true* in the new formula.

Since we know that exactly one of the variables x_f and x_t is set to *true*, an exact satisfying assignment that sets $n+k$ variables in F' to *true* sets k variables x_c to *true* and hence k variables in x_t. Thus, F has an exact satisfying assignment that sets k variables to *true*.

Exercise T18

A drunken sailor is in the pub and needs to go home. Since he's drunk, he cannot control the direction properly, and thus with each step forward he *randomly* stumbles one step to the left or one step to the right. Assuming a starting position of 0, and an offset of $+1/-1$ for each step, what is the expected position of the drunken sailor after n steps? If his ship is at a fixed position k at distance n, what is the probability that the sailor reaches his ship?



Martin Doege (GNU FDL)

Solution

Take independent random variables $X_1, X_2, \ldots$, where each variable is either 1 or -1 with a probability of 1/2 each. Let $S_n := \sum_{k=1}^{n} X_k$. Then $E(S_n) = \sum_{k=1}^{n}(-1/2 + 1/2) = 0$.
Among the 2^n random walks with n steps, a couple lead to the desired final position k. Each walk is equally likely, i.e., has a probability of $1/2^n$, so we just need to count the number of walks ending in position x. In order for S_n to be equal to a number x it is necessary and sufficient that the number of $+1$ steps in the walk exceeds those of -1 by exactly k. Thus, the number of walks which satisfy S_k is precisely the number of ways of choosing $(n+k)/2$ elements from an n element set (for this to be non-zero, it is necessary that $n+k$ be an even number), which is an entry in Pascal's triangle denoted by $\binom{n}{(n+k)/2}$.
Therefore, the probability that $S_n = k$ is equal to $2^{-n}\binom{n}{(n+k)/2}$.

Homework Assignment H17 (10 Points)
Prove that k-Monotone-XSAT $\leq_p$ k-Partition.

Solution
Let F be an input formula for k-Monotone-XSAT.
For each variable $x \in F$, we add the set $s_x = \{c \in F \mid x \in c\}$ of all clauses containing x to $\mathcal{S}$.
We claim that F has a exact satisfying assignment with k variables set to *true* iff $\mathcal{S}$ has a k-partition.
Let X be the set of variables set to *true* in an exact satisfying assignment with $|X| = k$. Since no variables occur negatively in F and since each clause must be satisfied, we know that $\bigcup_{x \in X} s_x$ contains all clauses. Moreover, each clause c is exact satisfied, hence c can only occur in one s_x with $x \in X$. Thus, $\{s_x \mid x_{\in} X\}$ is k-partition of $\mathcal{S}$.
Let S be a k-partition of $\mathcal{S}$. Since each element of $\mathcal{U}$ is covered, each clause $c \in F$ by setting the variables corresponding to the sets of S to *true*. And since each clause occurs in only one $s \in S$, each clause is satisfied exactly.

Homework Assignment H18 (10 Points)
List all possible 3-exchange local search transformations for the Traveling Sales Person problem.
Recall that the 2-exchange operation for TSP was defined as follows: Given a Hamiltonian tour T, take two edges $\{a, b\}$ and $\{c, d\}$ in the tour, where a, b, c, d are distinct nodes, and replace them by the edges $\{a, d\}$ and $\{b, c\}$.

Solution
Let $a, b, \ldots, c, d, \ldots, e, f \ldots, a$ be a tour in G. We only consider solutions that exchange all edges in $\{a, b\}$, $\{c, d\}$, $\{e, f\}$.
Note that all solutions contain the paths $b, \ldots, c$, $d, \ldots, e$, and $f \ldots, a$.
If we use $\{a, c\}$, we have to use the path $c, \ldots, b$. From b, we cannot use the edge to f, as this would yield two disconnected smaller cycles. If we use the edge from b to dm we also have to take $\{e, f\}$, which violates the condition that all edges are replaced. Hence, we only obtain one cycle $a, c, \ldots, b, e, \ldots, d, f \ldots, a$ that uses $\{a, c\}$.
If we use $\{a, d\}$, we then follow the path $d, \ldots, e$ and end up with two solutions for $\{e, b\}$ and $\{e, c\}$ as the next step:

- $\{a, d\}$, $\{e, b\}$, $\{c, f\}$ yields $a, d, \ldots, e, b, \ldots, c, f \ldots, a$
- $\{a, d\}$, $\{e, c\}$, $\{b, f\}$ yields $a, d, \ldots, e, c, \ldots, b, f \ldots, a$

Finally, if we use $\{a, e\}$, we again have to follow the path $d, \ldots, e$, but then have to follow the edge to b as we cannot go back to f early. This yields the path $a, e, \ldots, d, b, \ldots, c, f \ldots, a$.
This are all possible edges starting at a and hence we have enumerated all solutions. There are three additional exchanges that leave on edge unchanged.