TD n°13

16–17 mai 2018

Exercice 1 - LLL algorithmique

Soit $k \geqslant 40$ un entier pair. Le problème k-SAT est NP-complet (il l'est pour tout $k \geqslant 3$ fixé). Si on suppose maintenant que chaque variable apparait dans au plus $T = 2^{k/4}$ clauses, on sait par LLL qu'il y a toujours une solution. Pour autant, on ne sait a priori pas comment en trouver une efficacement.
Proposez un algorithme en deux phases (c'est le premier indice) pour résoudre ces instances particulières en temps polynomial en espérance.

Exercice 2 - Markov chains

For a Markov chain with a state space S of size d and a transition matrix $P_{i,j}$ what is the largest value of N such that $P^N_{i,j} > 0$ but $P^n_{i,j} = 0$ for all $1 \leq n < N$.

Exercice 3 - Triangles non-monochromatiques

A *coloring* of a graph is an assignment of a color to each of its vertices. A graph is k-colorable if there is a coloring of the graph with k colors such that no two adjacent vertices have the same color. Let G be a 3-colorable graph with n vertices.

3.1 Show that there exists a coloring of the graph using 2 colors with the property that no triangle is monochromatic. (Note that two adjacent vertices can be of the same color.)

Consider the following algorithm for coloring the vertices of G with two colors so that no triangle is monochromatic. The algorithm begins with an arbitrary 2-coloring of G. While there are any monochromatic triangles in G, the algorithm chooses one such triangle and changes the color of a randomly chosen vertex of that triangle. In the following you will derive the upper bound on the expected number of such recoloring steps before the algorithm terminates.
By the problem statement, there exists (an unknown) 3-coloring Red, Blue, Yellow of G. Denote R (resp. B, Y) the set of red (resp. blue, yellow) vertices if this coloring. Consider now a 2-coloring c of G (say, red and blue) and let $m(c)$ be the number of vertices from R that are not colored red in c plus the number of vertices from B that are not colored blue (i.e., $m(c) = \#\{v \in R | c(v) = \text{ blue }\} + \#\{v \in B | c(v) = \text{ red }\}$).

3.2 What would $m(c) = n$ or $m(c) = 0$ mean ?

3.3 Using ideas from the 2-SAT algorithm, model the evolution of $m(c)$ in a Markov chain on state $\{0, \ldots, n\}$. What can you tell about states j and $n - j$?

3.4 Let h_j be the expected number of recoloring needed to be performed for a coloring c s.t. $m(c) = j$. Express h_j as a function of h_{j-1} and h_{j+1} for $j = 1, \ldots, n-1$. Determine h_0, h_n.

3.5 Show that $h_j = h_{j+1} + f(j)$ for a certain function f to be determined s.t. $f(0) = -h_1$.

3.6 Show that $h_{\frac{n}{2}} = \mathcal{O}(n^2)$ and conclude on the running time of the algorithm. (One can use the symmetry relation between h_1 and h_{n-1}.)