Local Search

Analysis of Papadimitriou's Algorithm

Algorithms: Design and Analysis, Part II

Papadimitriou's Algorithm

n = number of variables

Repeat $\log_2 n$ times:
- choose random initial assignment
- repeat $2n^2$ times:
  - if current assignment satisfies all clauses, halt + report this
  - else, pick arbitrary unsatisfied clause and flip the value of one of its variables [choose between the two uniformly at random]

Report "unsatisfiable".

Obvious good points

(1) runs in polynomial time
(2) always correct on unsatisfiable instances

Satisfiable Instances

Theorem: for a satisfiable 2-SAT instance with n variables, Papadimitriou's algorithm produces a satisfying assignment with probability $\geq 1 - \frac{1}{n}$.

Proof: First focus on a single iteration of the outer for loop.

Fix an arbitrary satisfying assignment a^*.

Let a_t = algorithm's assignment after inner iteration t ($t = 0, 1, 2, \ldots, 2n^2$) [a random variable]

Let X_t = number of variables on which a_t and a^* agree. $X_t \in \{0, 1, 2, \ldots, n\}$

Note: if $X_t = n$, algorithm halts with satisfying assignment a^*.

Proof of Theorem (con'd)

Key point: Suppose a_t not a satisfying assignment and algorithm picks unsatisfied clause with variables x_i, x_j.

Note: since a^* is satisfying, it makes a different assignment than x_i or x_j (or both).

Consequence of algorithm's random variable flip:

① if a^* and a_t differ on <u>both</u> x_i & x_j, then $X_{t+1} = X_t + 1$ (100% probability)

② if a^* and a_t differ on exactly one of x_i, x_j then
$$X_{t+1} = \begin{cases} X_t + 1 & \text{(50\% probability)} \\ X_t - 1 & \text{(50\% probability)} \end{cases}$$

Quiz: Connection to Random Walks

Question: the random variables $X_0, X_1, X_2, \ldots, X_{2n^2}$ behave just like a random walk of the nonnegative integers except that:

(A) sometimes move right with 100% probability (instead of 50%)

(B) might have $X_0 > 0$ instead of $X_0 = 0$

(C) might stop early, before $X_t = n$

(D) all of the above

Completing the Proof

Consequence: Probability that a single iteration of the outer for loop finds a satisfying assignment is

$$\geq \Pr[T_n \leq 2n^2] \geq 1/2$$

from last video

Thus: $\Pr[\text{algorithm fails}] \leq \Pr[\text{all } \log_2 n \text{ independent trials fail}]$

$$\leq \left(\tfrac{1}{2}\right)^{\log_2 n}$$

$$= 1/n.$$

QED!