Example: A Random Walk

Consider a particle moving in an one-dimensional line.

At each point in time, the particle will move either 1 step to the right with probability p or 1 step to the left with probability $1 - p$.

Markov chain with transition probabilities:

$P_{i,i+1} = p$; $P_{i,i-1} = 1 - p$; $P_{i,i} = 0$; $i = 0, \pm 1, \ldots$.

For $j > i$,

$P^n_{i,j} = \binom{n}{(n-i+j)/2} p^{(n-i+j)/2} (1-p)^{(n+i-j)/2}$.

where $\binom{n}{x} = 0$ when x is not a nonnegative integer $\leq n$.

Randomized 2-SAT Algorithm

Given a formula with up to two variables per clause, find a Boolean assignment that satisfies all clauses.

Algorithm:

1. Start with an arbitrary assignment.
2. **Repeat** till all clauses are satisfied:
   (a) Pick an unsatisfied clause.
   (b) If the clause has one variable change the value of that variable.
   (c) If the clause has two variables choose one uniformly at random and change its value.

What the is the expected run-time of this algorithm?

W.l.o.g. assume that all clauses have two variables.

Assume that the formula has a satisfying assignment. Pick one such assignment S.

Let X_i be the number of variables in the current assignment A_i that match with S after iteration i of the algorithm.

Let n be the number of variables.

$$Pr(X_i = 1 \mid X_{i-1} = 0) = 1$$

For $1 \le t \le n - 1$,

$$Prob(X_i = t + 1 \mid X_{i-1} = t) \ge 1/2$$

$$Prob(X_i = t - 1 \mid X_{i-1} = t) \le 1/2$$

Consider the Markov chain:

$$Y_0 = X_0$$
$$Pr(Y_i = 1 \mid Y_{i-1} = 0) = 1$$

for $1 \leq t \leq n - 1$,

$$Prob(Y_i = t + 1 \mid Y_{i-1} = t) = 1/2$$

$$Prob(Y_i = t - 1 \mid Y_{i-1} = t) = 1/2$$

The expected number of steps to reach n starting from any point is larger for the Markov chain Y than for the process X.

Let D_t be the expected number of steps to termination when we have t incorrect variable assignments.

$D_0 = 0.$

$D_n = 1 + D_{n-1}.$

$D_t = 1 + \frac{1}{2}D_{t+1} + \frac{1}{2}D_{t-1}$

We "guess" $D_t = t(2n - t)$ and $D_0 = 0.$

$$D_t = 1 + \frac{1}{2}(t+1)(2n-t-1) + \frac{1}{2}(t-1)(2n-t+1) =$$

$$1+\frac{1}{2}(2nt+2n-t^2-t-t-1+2nt-2n-t^2+t+t-1) =$$

$$1 + 2nt - t^2 - 1 = t(2n - t).$$

$$D_n = 1 + D_{n-1} = 1 + (n-1)(2n - n + 1) = n^2.$$

Theorem 1. *Assuming that the 2-SAT formula has a satisfying assignment the expected run-time to find a satisfying assignment is* $O(n^2)$*.*

Theorem 2. *There is a one-sided error randomized algorithm for the 2-SAT problem that terminates in* $O(n^2 \log n)$ *time with high probability and returns an assignment when the formula is satisfiable, and always returns "UNSATISFIABLE" when no assignment exists.*

Proof. The probability that the algorithm does not find an assignment when exists in $2n^2$ steps is bounded by $\frac{1}{2}$. $\square$

A Randomized algorithm for 3-SAT

Consider a randomized algorithm for 3-SAT similar to the one for 2-SAT.

As before, let S be a satisfying assignment.

Let X_i be the number of variables in the current assignment A_i that match with S after iteration i of the algorithm.

$$Pr(X_i = 1 \mid X_{i-1} = 0) = 1$$

For $1 \leq t \leq n-1$,

$$Prob(X_i = t+1 \mid X_{i-1} = t) \geq 1/3$$

$$Prob(X_i = t-1 \mid X_{i-1} = t) \leq 2/3$$

$$Y_0 = X_0$$

$$Pr(Y_i = 1 \mid Y_{i-1} = 0) = 1$$

for $1 \leq t \leq n-1$,

$$Prob(Y_i = t+1 \mid Y_{i-1} = t) = 1/2$$

$$Prob(Y_i = t-1 \mid Y_{i-1} = t) = 1/2$$

Let D_t be the expected number of steps to termination when we have t incorrect variable assignments.

$D_0 = 0.$

$D_n = 1 + D_{n-1}.$

$D_t = 1 + \frac{2}{3}D_{t+1} + \frac{1}{3}D_{t-1}$

We can show that the expected number of steps to reach n is $\Theta(2^n)$.

A Faster Randomized algorithm for 3-SAT

Algorithm:

Step 1. Repeat till all clauses are satisfied:

1. Start with an truth assignment chosen uniformly at random.

2. **Repeat** the following up to $3n$ times, terminating if a satisfying assignment is found:
   (a) Pick an arbitrary unsatisfied clause.
   (b) Choose one variable uniformly at random and change its value.

Step 2. If a valid truth assignment has been found, return it.
Step 3. Otherwise, return "unsatisfiable".

Theorem 3. *Assuming that the 3-SAT formula has a satisfying assignment the expected run-time to find a satisfying assignment is* $O(n^{3/2}(4/3)^n)$.

Proof:

Assume that the formula has a satisfying assignment S.

We determine the expected number of times we should repeat Step 1 before we find a satisfying assignment.

Let q be the probability that the algorithm reaches a satisfying assignment in $3n$ steps starting with a truth assignment chosen uniformly at random.

Let q_j be a lower bound on the probability that the algorithm reaches S (or some other satisfying assignment) when it starts with a truth assignment that includes exactly j variables that do not agree with S. Then,

$$q_j \geq \binom{3j}{j}(\frac{2}{3})^j(\frac{1}{3})^{2j}$$

.