Randomness in Computing

Lecture 24

CS 537

Last time

- Drunkard's walk
- Markov chains
- Randomized algorithm for 2SAT

Today

- Randomized algorithm for 3SAT
- Gambler's ruin
- Classification of Markov chains
- Stationary distributions

Application: Algorithm for 3SAT

Recall: A 3CNF formula is an AND of clauses

- Each clause is an OR of literals.
- Each literal is a Boolean variable or its negation.
- E.g. $(x_1 \vee \overline{x_2} \vee x_3) \wedge (x_2 \vee \overline{x_3} \vee x_4) \wedge (x_1 \vee x_3 \vee \overline{x_4})$

3SAT Problem (search version): Given a 3CNF formula, find a satisfying assignment if it is satisfiable.

Application: Algorithm for 3SAT

- First try: the same algorithm as for 2SAT

Input: a 3CNF formula ϕ on n variables

parameter

1. Start with an arbitrary truth assignment, e.g., all 0's.
2. Repeat R times, terminating if ϕ is satisfied:
   a) Choose an arbitrary clause C that is not satisfied.
   b) Pick a uniformly random literal in C and flip its assignment.
3. If a satisfying assignment is found, return it.
4. Otherwise, return ``unsatisfiable''.

- We want to analyze the number of steps (iterations) necessary.

Analysis: What should we change?

- Let S = a satisfying assignment of ϕ.
- A_i = an assignment to ϕ after i steps
- X_i = number of variables that have the same value in A_i and S

When $X_i = n$, the algorithm terminates with a satisfying assignment.

- If $X_i = 0$ then $X_{i+1} = 1$

$$\Pr[X_{i+1} = 1 \mid X_i = 0] = 1$$

- If $X_i \in [1, n-1]$ then A_i disagrees with S on 1 to 3 literals of C

$$\Pr[X_{i+1} = j+1 \mid X_i = j] \geq 1/3$$
$$\Pr[X_{i+1} = j-1 \mid X_i = j] \leq 2/3$$

$X_0, X_1, X_2, \ldots$ is not necessarily a Markov chain

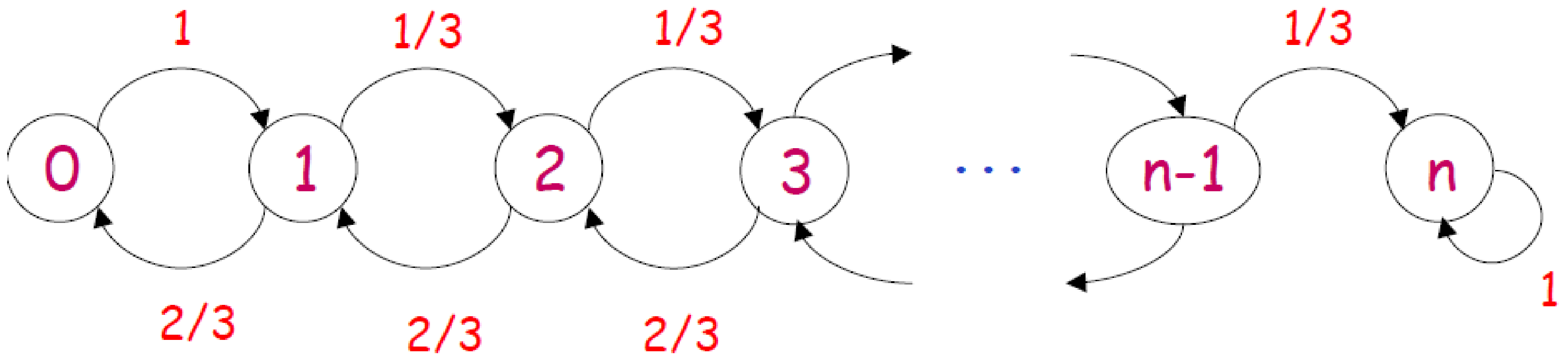
Analysis: What should we change?

- Define a Markov Chain $Y_0, Y_1, Y_2, \ldots$

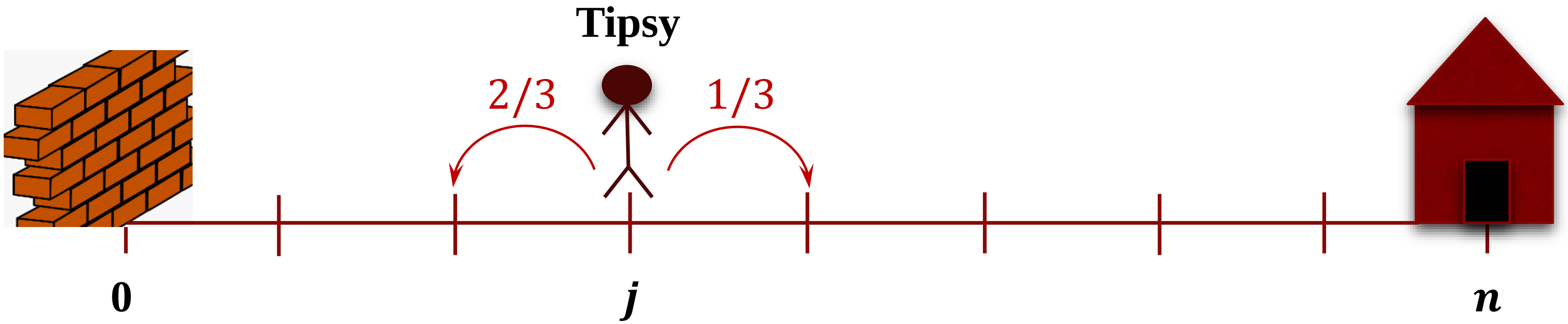
$$Y_0 = X_0$$
$$\Pr[Y_{i+1} = 1 \mid Y_i = 0] = 1$$
$$\Pr[Y_{i+1} = j + 1 \mid Y_i = j] = 1/3$$
$$\Pr[Y_{i+1} = j - 1 \mid Y_i = j] = 2/3$$

- ``Pessimistic version'' of stochastic process $X_0, X_1, X_2, \ldots$

The expected time to reach n is larger for $Y_0, Y_1, Y_2, \ldots$ than for $X_0, X_1, X_2, \ldots$



Expected time to reach n



s_j = expected number of steps to reach position n, starting at postion j

$$s_0 = s_1 + 1$$

$$s_n = 0$$

for $j \in [1, n-1]$: $\quad s_j = 1 + \frac{2s_{j-1}}{3} + \frac{s_{j+1}}{3}$

$$\boxed{\boldsymbol{s_j = 2^{n+2} - 2^{j+2} - 3(n-j)}}$$

$\Theta(2^n)$ steps on average

Not good

Ideas

- The longer the algorithm runs, the more likely it is to move towards 0.

  Idea: Restart after a fixed number of steps.

- How do we get better at the starting assignment?

  Idea: Choose one at random.

What's the distribution of the number of variables that match S?

With nonnegligible probability we significantly exceed $n/2$ matches

Modified algorithm for 3SAT

parameter

Input: a 3CNF formula ϕ on n variables

1. Repeat R times, terminating if ϕ is satisfied:
   a) Start with a **uniformly random** truth assignment.
   b) **Repeat $3n$ times:**
      i. Choose an arbitrary clause C that is not satisfied.
      ii. Pick a uniformly random literal in C and flip its assignment.
2. If a satisfying assignment is found, return it.
3. Otherwise, return ``unsatisfiable''.

Analysis

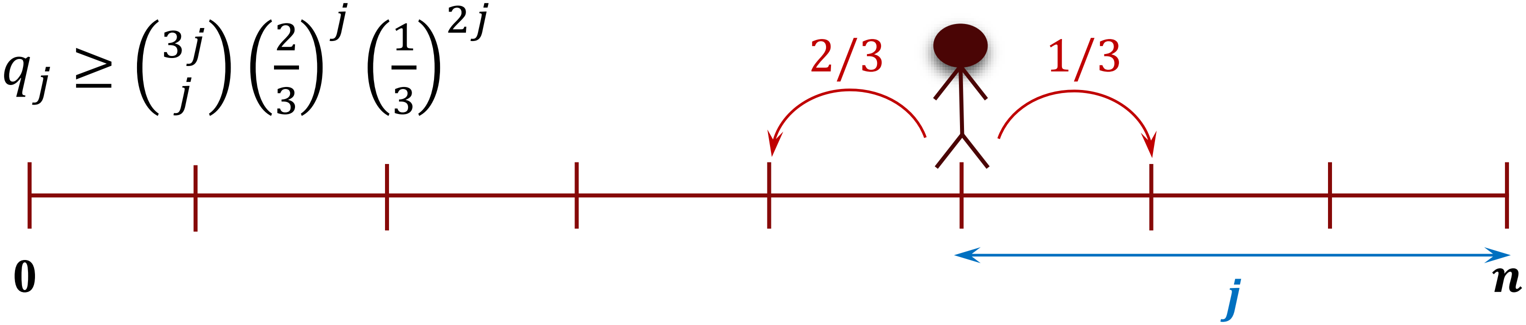
Want to understand: the probability of reaching the satisfying assignment S in $3n$ steps starting from a random assignment.

- Let q be the probability that Markov chain Y reaches state n in $3n$ steps starting from a state that corresponds to a random assignment.
- Let $\boldsymbol{q_j}$ be the probability that Markov chain Y reaches state n in $3n$ steps starting from the state $\boldsymbol{n-j}$.

$$q = \sum_{j=0}^{n} \Pr[\text{starting in state } n-j] \cdot q_j$$

- One way for Y to reach state n from state $n-j$ is
  to move left j times and right $2j$ times in the first $3j$ moves.

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



Analysis: Bounding q_j

So far: $q_j \geq \binom{3j}{j} \left(\frac{2}{3}\right)^j \left(\frac{1}{3}\right)^{2j}$

- By Stirling's formula, $m! = \Theta\left(\sqrt{m} \cdot \left(\frac{m}{e}\right)^m\right)$
- When $j > 0$,

Analysis: Bounding q

So far: q is the probability that Markov chain Y reaches state n in $3n$ steps starting from a state that corresponds to a random assignment.

$$q = \sum_{j=0}^{n} \Pr[\text{starting in state } n-j] \cdot q_j\,; \qquad q_j = \Omega\left(\frac{1}{\sqrt{j}} \cdot \frac{1}{2^j}\right) \text{ when j} > 0$$

Analysis: final touches

- When ϕ is satisfiable, one run finds a satisfying assignment with probability at least $q = \Omega\left(\frac{1}{\sqrt{n}} \cdot \left(\frac{3}{4}\right)^n\right)$
- The number of runs until finding a satisfying assignment is a geometric random variable with expectation at most
$$\frac{1}{q} = O\left(\sqrt{n} \cdot \left(\frac{4}{3}\right)^n\right)$$
- Each run uses $3n$ steps, so the expected number of steps is
$$O\left(n\sqrt{n} \cdot \left(\frac{4}{3}\right)^n\right)$$
- As for 2SAT, we set R to $2a$ times the expected number of steps to get a Monte Carlo algorithm that fails w. p. at most 2^{-a}.

The Gambler's Ruin

Player 1 | **Player 2**

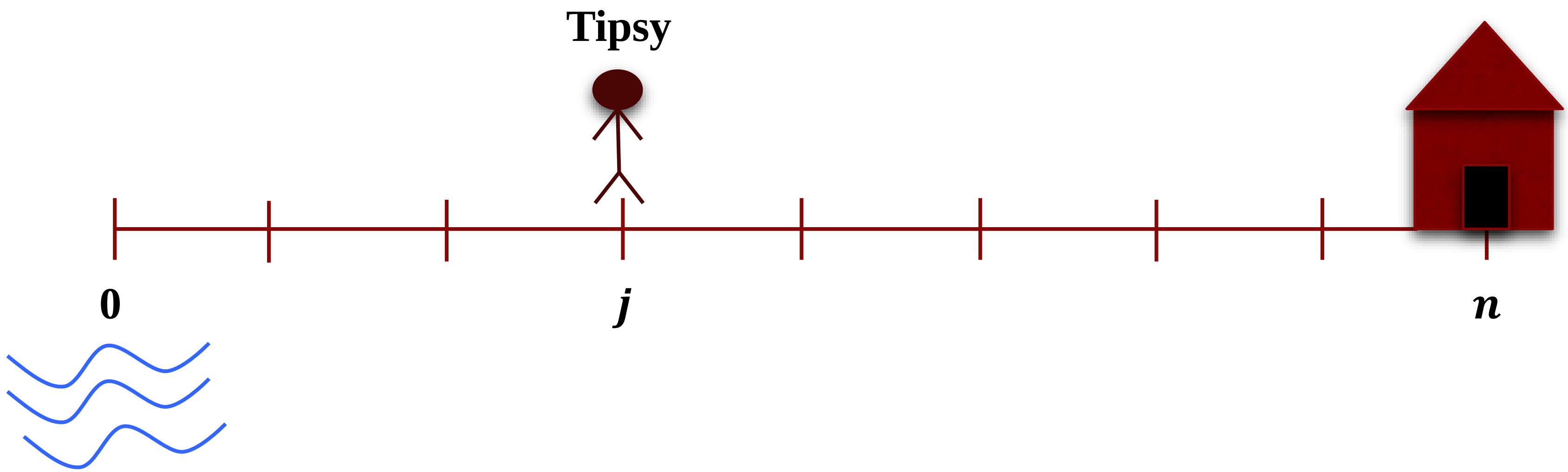
with probability ½
Player 1 loses 1 dollar

with probability ½
Player 2 loses 1 dollar

Limit: ℓ_1 dollars | **Limit: ℓ_2 dollars**

- State at time t: number of dollars won by Player 1 (could be negative)
- Find the probability that Player 1 wins ℓ_2 dollars before losing ℓ_1 dollars and the expected time to finish the game.

Recall: Drunkard's walk



- $\Pr[\text{Tipsy goes home } \mid \text{he started at position } j] = \frac{j}{n}$
- Expected number of steps to finish the walk, starting at postion j, is $j(n-j)$

Poll questions

The probability Player 1 wins ℓ_2 before losing ℓ_1 dollars is

A. $\frac{\ell_1}{\ell_2}$

B. $\frac{\ell_1}{\ell_1+\ell_2}$

C. $\frac{\ell_2}{\ell_1+\ell_2}$

D. $\frac{1}{2}$

- The expected time to finish the game is

A. $\ell_1(\ell_2 - \ell_1)$

B. $\ell_1\ell_2$

C. $(\ell_1 + \ell_2)(\ell_2 - \ell_1)$

D. ℓ_2^2

Classification of Markov chains

- We want to study Markov chains that ``mix'' well.
- We will define Markov chains that avoid some problematic behaviors: irreducible and aperiodic.
- A finite Markov chain is irreducible if its graph representation consists of one strongly connected component.

Periodicity

- Example: a Markov chain whose states are integers and it moves to each neighboring state with probability ½.

  If the chain starts at 0, when can it be in an even-numbered state?

- A state is periodic if there exists an integer $\Delta > 1$ such that $\Pr[X_{t+s} = j \mid X_t = j] = 0$ unless s is divisible by Δ ; otherwise, it is aperiodic.
- A Markov chain is aperiodic if *all* its states are aperiodic.

CS 537 Stationary Distributions

Recall: $\bar{p}(t+1) = \bar{p}(t)\boldsymbol{P}$, where $\bar{p}(t)$ is the distribution of the state of the chain at time t and $\boldsymbol{P}$ is its transition probability matrix.

- A stationary distribution of a Markov chain is a probability distribution $\bar{\pi}$ such that $\bar{\pi} = \bar{\pi}\boldsymbol{P}$.

(Describes steady state behavior of a Markov chain.)

Example: Define Markov chain by the following random walk on the nodes of an n-cycle. At each step, stay at the same node w.p. ½;
go left w.p. ¼ and right w.p. ¼.

Fundamental theorem

- A stationary distribution of a Markov chain is a probability distribution $\bar{\pi}$ such that $\bar{\pi} = \bar{\pi}\boldsymbol{P}$.

(Describes steady state behavior of a Markov chain.)

Fundamental Theorem of Markov Chains (selected items)

Every finite, irreducible and aperiodic Markov chain satisfies the following:

1. There is a unique stationary distribution $\bar{\pi} = (\pi_0, \pi_1, \dots, \pi_n)$, where $\pi_i > 0$ for all $i \in \{0,1,\dots,n\}$.
2. For all $i \in \{0,1,\dots,n\}$, the hitting time $h_{ii} = 1/\pi_i$.

- The hitting time from u to v, denoted $h_{u,v}$, is the expected time to reach state v from state u.