

Chapter 19

Random Processes

Random Walks are used to model situations in which an object moves in a sequence of steps in randomly chosen directions. For example in Physics, three-dimensional random walks are used to model Brownian motion and gas diffusion. In this chapter we'll examine two examples of random walks. First, we'll model gambling as a simple 1-dimensional random walk—a walk along a straight line. Then we'll explain how the Google search engine used random walks through the graph of world-wide web links to determine the relative importance of websites.

19.1 Gamblers' Ruin

Suppose a gambler starts with an initial stake of n dollars and makes a sequence of \$1 bets. If he wins an individual bet, he gets his money back plus another \$1. If he loses, he loses the \$1.

We can model this scenario as a random walk between integer points on the real line. The position on the line at any time corresponds to the gambler's cash-on-hand or *capital*. Walking one step to the right (left) corresponds to winning (losing) a \$1 bet and thereby increasing (decreasing) his capital by \$1. The gambler plays until either he is bankrupt or increases his capital to a target amount of T dollars. If he reaches his target, then he is called an overall *winner*, and his *profit*, m , will be $T - n$ dollars. If his capital reaches zero dollars before reaching his target, then we say that he is "ruined" or *goes broke*. We'll assume that the gambler has the same probability, p , of winning each individual \$1 bet and that the bets are mutually independent. We'd like to find the probability that the gambler wins.

The gambler's situation as he proceeds with his \$1 bets is illustrated in Figure 19.1. The random walk has boundaries at 0 and T . If the random walk ever reaches either of these boundary values, then it terminates.

In a *fair game*, the gambler is equally likely to win or lose each bet, that is $p = 1/2$. The corresponding random walk is called *unbiased*. The gambler is more likely to win if $p > 1/2$ and less likely to win if $p < 1/2$; these random walks are called

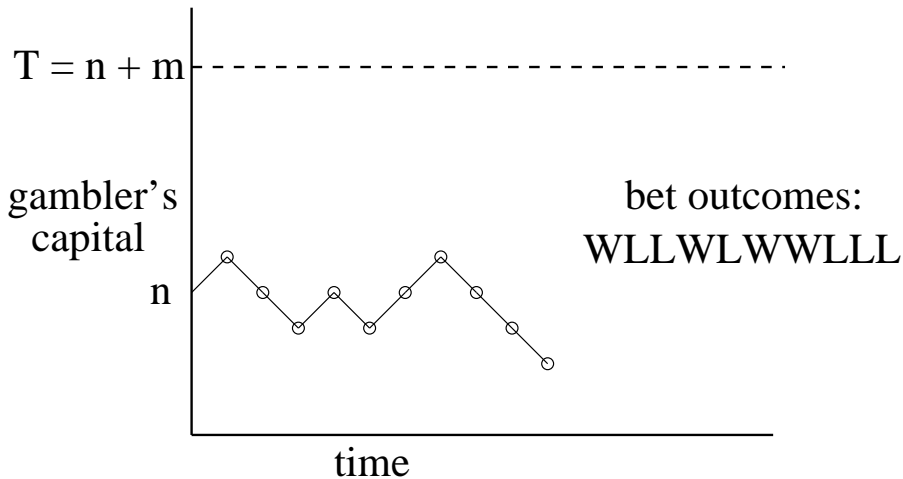


Figure 19.1: This is a graph of the gambler's capital versus time for one possible sequence of bet outcomes. At each time step, the graph goes up with probability p and down with probability $1 - p$. The gambler continues betting until the graph reaches either 0 or T .

biased. We want to determine the probability that the walk terminates at boundary T , namely, the probability that the gambler is a winner. We'll do this by showing that the probability satisfies a simple linear recurrence and solving the recurrence, but before we derive the probability, let's just look at what it turns out to be.

Let's begin by supposing the coin is fair, the gambler starts with 100 dollars, and he wants to double his money. That is, he plays until he goes broke or reaches a target of 200 dollars. Since he starts equidistant from his target and bankruptcy, it's clear by symmetry that his probability of winning in this case is $1/2$.

We'll show below that starting with n dollars and aiming for a target of $T \geq n$ dollars, the probability the gambler reaches his target before going broke is n/T . For example, suppose he want to win the same \$100, but instead starts out with \$500. Now his chances are pretty good: the probability of his making the 100 dollars is $5/6$. And if he started with one million dollars still aiming to win \$100 dollars he almost certain to win: the probability is $1M/(1M + 100) > .9999$.

So in the fair game, the larger the initial stake relative to the target, the higher the probability the gambler will win, which makes some intuitive sense. But note that although the gambler now wins nearly all the time, the game is still fair. When he wins, he only wins \$100; when he loses, he loses big: \$1M. So the gambler's average win is actually zero dollars.

Now suppose instead that the gambler chooses to play roulette in an American casino, always betting \$1 on red. A roulette wheel has 18 black numbers, 18 red numbers, and 2 green numbers, designed so that each number is equally likely to appear. So this game is slightly biased against the gambler: the probability of winning a single bet is $p = 18/38 \approx 0.47$. It's the two green numbers that

slightly bias the bets and give the casino an edge. Still, the bets are almost fair, and you might expect that starting with \$500, the gambler has a reasonable chance of winning \$100 —the 5/6 probability of winning in the unbiased game surely gets reduced, but perhaps not too drastically.

Not so! The gambler's odds of winning \$100 making one dollar bets against the "slightly" unfair roulette wheel are less than 1 in 37,000. If that seems surprising, listen to this: *no matter how much money* the gambler has to start —\$5000, \$50,000, $\$5 \cdot 10^{12}$ —his odds are still less than 1 in 37,000 of winning a mere 100 dollars!

Moral: Don't play!

The theory of random walks is filled with such fascinating and counter-intuitive conclusions.

19.1.1 A Recurrence for the Probability of Winning

The probability the gambler wins is a function of his initial capital, n , his target, $T \geq n$, and the probability, p , that he wins an individual one dollar bet. Let's let p and T be fixed, and let w_n be the gambler's probability of winning when his initial capital is n dollars. For example, w_0 is the probability that the gambler will win given that he starts off broke and w_T is the probability he will win if he starts off with his target amount, so clearly

$$w_0 = 0, \quad (19.1)$$

$$w_T = 1. \quad (19.2)$$

Otherwise, the gambler starts with n dollars, where $0 < n < T$. Consider the outcome of his first bet. The gambler wins the first bet with probability p . In this case, he is left with $n + 1$ dollars and becomes a winner with probability w_{n+1} . On the other hand, he loses the first bet with probability $q := 1 - p$. Now he is left with $n - 1$ dollars and becomes a winner with probability w_{n-1} . By the Total Probability Rule, he wins with probability $w_n = pw_{n+1} + qw_{n-1}$. Solving for w_{n+1} we have

$$w_{n+1} = \frac{w_n}{p} - rw_{n-1} \quad (19.3)$$

where

$$r ::= \frac{q}{p}.$$

This recurrence holds only for $n + 1 \leq T$, but there's no harm in using (19.3) to define w_{n+1} for all $n + 1 > 1$. Now, letting

$$W(x) ::= w_0 + w_1x + w_2x^2 + \cdots$$

be the generating function for the w_n , we derive from (19.3) and (19.1) using our generating function methods that

$$xW(x) = \frac{w_1x}{(1-x)(1-rx)}, \quad (19.4)$$

so if $p \neq q$, then using partial fractions we can calculate that

$$W(x) = \frac{w_1}{r-1} \left(\frac{1}{1-rx} - \frac{1}{1-x} \right),$$

which implies

$$w_n = w_1 \frac{r^n - 1}{r - 1}. \quad (19.5)$$

Now we can use (19.5) to solve for w_1 by letting $n = T$ to get

$$w_1 = \frac{r - 1}{r^T - 1}.$$

Plugging this value of w_1 into (19.5), we finally arrive at the solution:

$$w_n = \frac{r^n - 1}{r^T - 1}. \quad (19.6)$$

The expression (19.6) for the probability that the Gambler wins in the biased game is a little hard to interpret. There is a simpler upper bound which is nearly tight when the gambler's starting capital is large and the game is biased *against* the gambler. Then both the numerator and denominator in the quotient in (19.6) are positive, and the quotient is less than one. This implies that

$$w_n < \frac{r^n}{r^T} = r^{T-n},$$

which proves:

Corollary 19.1.1. *In the Gambler's Ruin game with probability $p < 1/2$ of winning each individual bet, with initial capital, n , and target, T ,*

$$\Pr \{ \text{the gambler is a winner} \} < \left(\frac{p}{q} \right)^{T-n} \quad (19.7)$$

The amount $T - n$ is called the Gambler's *intended profit*. So the gambler gains his intended profit before going broke with probability at most p/q raised to the intended-profit power. Notice that this upper bound does not depend on the gambler's starting capital, but only on his intended profit. This has the amazing consequence we announced above: *no matter how much money he starts with*, if he makes \$1 bets on red in roulette aiming to win \$100, the probability that he wins is less than

$$\left(\frac{18/38}{20/38} \right)^{100} = \left(\frac{9}{10} \right)^{100} < \frac{1}{37,648}.$$

The bound (19.7) is exponential in the intended profit. So, for example, doubling his intended profit will square his probability of winning. In particular, the

probability that the gambler's stake goes up 200 dollars before he goes broke playing roulette is at most

$$(9/10)^{200} = ((9/10)^{100})^2 = \left(\frac{1}{37,648}\right)^2,$$

which is about 1 in 70 billion.

The solution (19.6) only applies to biased walks, but the method above works just as well in getting a formula for the unbiased case (except that the partial fractions involve a repeated root). But it's simpler settle the fair case simply by taking the limit as r approaches 1 of (19.6). By L'Hopital's Rule this limit is n/T , as we claimed above.

19.1.2 Intuition

Why is the gambler so unlikely to make money when the game is slightly biased against him? Intuitively, there are two forces at work. First, the gambler's capital has random upward and downward *swings* due to runs of good and bad luck. Second, the gambler's capital will have a steady, downward *drift*, because the negative bias means an average loss of a few cents on each \$1 bet. The situation is shown in Figure 19.2.

Our intuition is that if the gambler starts with, say, a billion dollars, then he is sure to play for a very long time, so at some point there should be a lucky, upward swing that puts him \$100 ahead. The problem is that his capital is steadily drifting downward. If the gambler does not have a lucky, upward swing early on, then he is doomed. After his capital drifts downward a few hundred dollars, he needs a huge upward swing to save himself. And such a huge swing is extremely improbable. As a rule of thumb, *drift dominates swings* in the long term.

19.1.3 Problems

Homework Problems

Problem 19.1.

A drunken sailor wanders along main street, which conveniently consists of the points along the x axis with integral coordinates. In each step, the sailor moves one unit left or right along the x axis. A particular *path* taken by the sailor can be described by a sequence of "left" and "right" steps. For example, $\langle \text{left}, \text{left}, \text{right} \rangle$ describes the walk that goes left twice then goes right.

We model this scenario with a random walk graph whose vertices are the integers and with edges going in each direction between consecutive integers. All edges are labelled $1/2$.

The sailor begins his random walk at the origin. This is described by an initial distribution which labels the origin with probability 1 and all other vertices with probability 0. After one step, the sailor is equally likely to be at location 1 or -1 ,

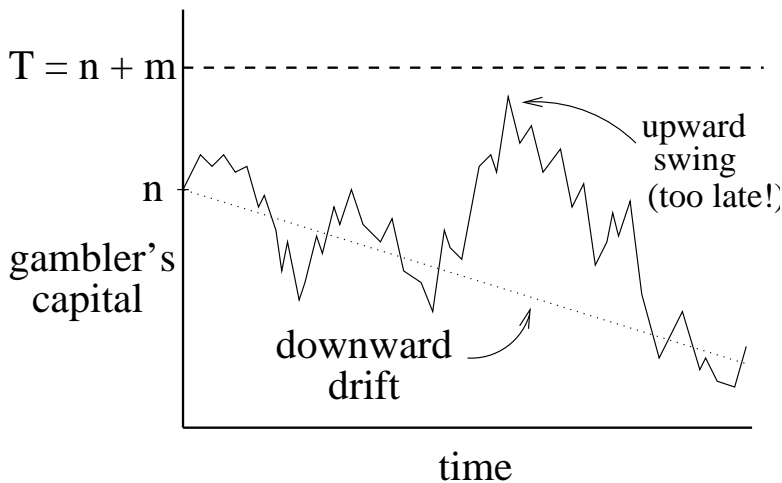


Figure 19.2: In an unfair game, the gambler's capital swings randomly up and down, but steadily drifts downward. If the gambler does not have a winning swing early on, then his capital drifts downward, and later upward swings are insufficient to make him a winner.

so the distribution after one step gives label $1/2$ to the vertices 1 and -1 and labels all other vertices with probability 0 .

(a) Give the distributions after the 2nd, 3rd, and 4th step by filling in the table of probabilities below, where omitted entries are 0 . For each row, write all the nonzero entries so they have the same denominator.

	location								
	-4	-3	-2	-1	0	1	2	3	4
initially					1				
after 1 step				$1/2$	0	$1/2$			
after 2 steps			?	?	?	?	?		
after 3 steps		?	?	?	?	?	?	?	
after 4 steps	?	?	?	?	?	?	?	?	?

(b)

1. What is the final location of a t -step path that moves right exactly i times?
2. How many different paths are there that end at that location?
3. What is the probability that the sailor ends at this location?

(c) Let L be the random variable giving the sailor's location after t steps, and let $B ::= (L+t)/2$. Use the answer to part (b) to show that B has an unbiased binomial density function.

(d) Again let L be the random variable giving the sailor's location after t steps, where t is even. Show that

$$\Pr \left\{ |L| < \frac{\sqrt{t}}{2} \right\} < \frac{1}{2}.$$

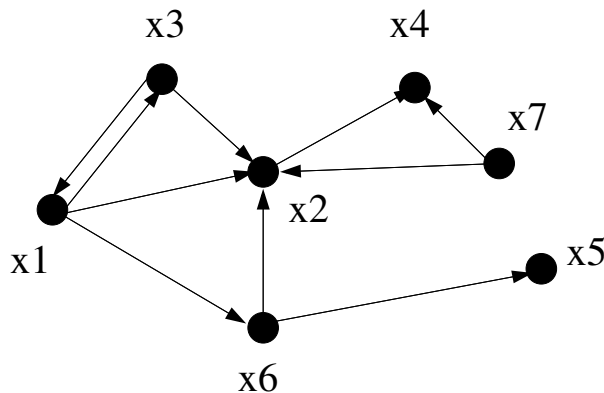
So there is a better than even chance that the sailor ends up at least $\sqrt{t}/2$ steps from where he started.

Hint: Work in terms of B . Then you can use an estimate that bounds the binomial distribution. Alternatively, observe that the origin is the most likely final location and then use the asymptotic estimate

$$\Pr \{L = 0\} = \Pr \{B = t/2\} \sim \sqrt{\frac{2}{\pi t}}.$$

19.2 Random Walks on Graphs

The hyperlink structure of the World Wide Web can be described as a digraph. The vertices are the web pages with a directed edge from vertex x to vertex y if x has a link to y . For example, in the following graph the vertices x_1, \dots, x_n correspond to web pages and $x_i \rightarrow x_j$ is a directed edge when page x_i contains a hyperlink to page x_j .



The web graph is an enormous graph with many billions and probably even trillions of vertices. At first glance, this graph wouldn't seem to be very interesting. But in 1995, two students at Stanford, Larry Page and indexBrin, Sergey Sergey Brin realized that the structure of this graph could be very useful in building a search engine. Traditional document searching programs had been around for a long time and they worked in a fairly straightforward way. Basically, you would enter some search terms and the searching program would return all documents containing those terms. A relevance score might also be returned for each document based on the frequency or position that the search terms appeared in

the document. For example, if the search term appeared in the title or appeared 100 times in a document, that document would get a higher score. So if an author wanted a document to get a higher score for certain keywords, he would put the keywords in the title and make it appear in lots of places. You can even see this today with some bogus web sites.

This approach works fine if you only have a few documents that match a search term. But on the web, there are billions of documents and millions of matches to a typical search.

For example, a few years ago a search on Google for “math for computer science notes” gave 378,000 hits! How does Google decide which 10 or 20 to show first? It wouldn’t be smart to pick a page that gets a high keyword score because it has “math math . . . math” across the front of the document.

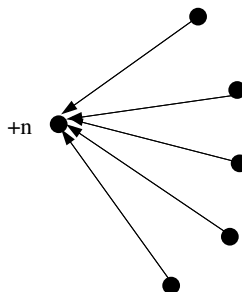
One way to get placed high on the list is to pay Google an advertising fees—and Google gets an enormous revenue stream from these fees. Of course an early listing is worth a fee only if an advertiser’s target audience is attracted to the listing. But an audience does get attracted to Google listings because its ranking method is really good at determining the most relevant web pages. For example, Google demonstrated its accuracy in our case by giving first rank to the Fall 2002 open courseware page for 6.042 : -) . So how did Google know to pick 6.042 to be first out of 378,000?

Well back in 1995, Larry and Sergey got the idea to allow the digraph structure of the web to determine which pages are likely to be the most important.

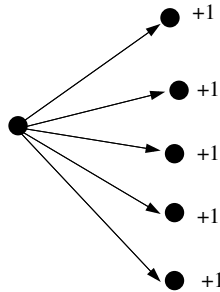
19.2.1 A First Crack at Page Rank

Looking at the web graph, any idea which vertex/page might be the best to rank 1st? Assume that all the pages match the search terms for now. Well, intuitively, we should choose x_2 , since lots of other pages point to it. This leads us to their first idea: try defining the *page rank* of x to be the number of links pointing to x , that is, $\text{indegree}(x)$. The idea is to think of web pages as voting for the most important page—the more votes, the better rank.

Of course, there are some problems with this idea. Suppose you wanted to have your page get a high ranking. One thing you could do is to create lots of dummy pages with links to your page.



There is another problem—a page could become unfairly influential by having lots of links to other pages it wanted to hype.



So this strategy for high ranking would amount to, “vote early, vote often,” which is no good if you want to build a search engine that’s worth paying fees for. So, admittedly, their original idea was not so great. It was better than nothing, but certainly not worth billions of dollars.

19.2.2 Random Walk on the Web Graph

But then Sergey and Larry thought some more and came up with a couple of improvements. Instead of just counting the indegree of a vertex, they considered the probability of being at each page after a long random walk on the web graph. In particular, they decided to model a user’s web experience as following each link on a page with uniform probability. That is, they assigned each edge $x \rightarrow y$ of the web graph with a probability conditioned on being on page x :

$$\Pr \{ \text{follow link } x \rightarrow y \mid \text{at page } x \} ::= \frac{1}{\text{outdegree}(x)}.$$

The user experience is then just a random walk on the web graph.

For example, if the user is at page x , and there are three links from page x , then each link is followed with probability $1/3$.

We can also compute the probability of arriving at a particular page, y , by summing over all edges pointing to y . We thus have

$$\begin{aligned} \Pr \{ \text{go to } y \} &= \sum_{\text{edges } x \rightarrow y} \Pr \{ \text{follow link } x \rightarrow y \mid \text{at page } x \} \cdot \Pr \{ \text{at page } x \} \\ &= \sum_{\text{edges } x \rightarrow y} \frac{\Pr \{ \text{at } x \}}{\text{outdegree}(x)} \end{aligned} \quad (19.8)$$

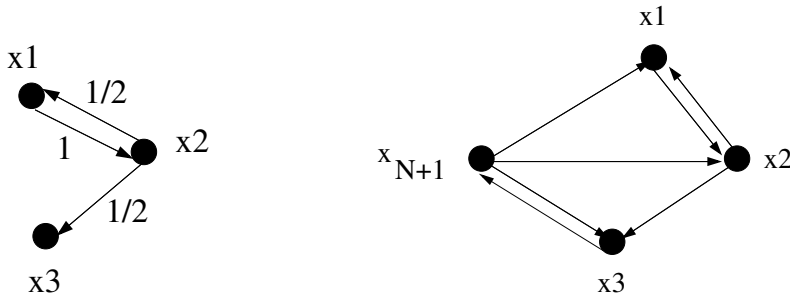
For example, in our web graph, we have

$$\Pr \{ \text{go to } x_4 \} = \frac{\Pr \{ \text{at } x_7 \}}{2} + \frac{\Pr \{ \text{at } x_2 \}}{1}.$$

One can think of this equation as x_7 sending half its probability to x_2 and the other half to x_4 . The page x_2 sends all of its probability to x_4 .

There's one aspect of the web graph described thus far that doesn't mesh with the user experience —some pages have no hyperlinks out. Under the current model, the user cannot escape these pages. In reality, however, the user doesn't fall off the end of the web into a void of nothingness. Instead, he restarts his web journey.

To model this aspect of the web, Sergey and Larry added a supervertex to the web graph and had every page with no hyperlinks point to it. Moreover, the supervertex points to every other vertex in the graph, allowing you to restart the walk from a random place. For example, below left is a graph and below right is the same graph after adding the supervertex x_{N+1} .



The addition of the supervertex also removes the possibility that the value $1/\text{outdegree}(x)$ might involve a division by zero.

19.2.3 Stationary Distribution & Page Rank

The basic idea of page rank is just a stationary distribution over the web graph, so let's define a stationary distribution.

Suppose each vertex is assigned a probability that corresponds, intuitively, to the likelihood that a random walker is at that vertex at a randomly chosen time. We assume that the walk never leaves the vertices in the graph, so we require that

$$\sum_{\text{vertices } x} \Pr \{\text{at } x\} = 1. \quad (19.9)$$

Definition 19.2.1. An assignment of probabilities to vertices in a digraph is a *stationary distribution* if for all vertices x

$$\Pr \{\text{at } x\} = \Pr \{\text{go to } x \text{ at next step}\}$$

Sergey and Larry defined their page ranks to be a stationary distribution. They did this by solving the following system of linear equations: find a nonnegative number, $\text{PR}(x)$, for each vertex, x , such that

$$\text{PR}(x) = \sum_{\text{edges } y \rightarrow x} \frac{\text{PR}(y)}{\text{outdegree}(y)}, \quad (19.10)$$

corresponding to the intuitive equations given in (19.8). These numbers must also satisfy the additional constraint corresponding to (19.9):

$$\sum_{\text{vertices } x} \text{PR}(x) = 1. \quad (19.11)$$

So if there are n vertices, then equations (19.10) and (19.11) provide a system of $n + 1$ linear equations in the n variables, $\text{PR}(x)$. Note that constraint (19.11) is needed because the remaining constraints (19.10) could be satisfied by letting $\text{PR}(x) ::= 0$ for all x , which is useless.

Sergey and Larry were smart fellows, and they set up their page rank algorithm so it would always have a meaningful solution. Their addition of a supervertex ensures there is always a *unique* stationary distribution. Moreover, starting from *any* vertex and taking a sufficiently long random walk on the graph, the probability of being at each page will get closer and closer to the stationary distribution. Note that general digraphs without supervertices may have neither of these properties: there may not be a unique stationary distribution, and even when there is, there may be starting points from which the probabilities of positions during a random walk do not converge to the stationary distribution.

Now just keeping track of the digraph whose vertices are billions of web pages is a daunting task. That's why Google is building power plants. Indeed, Larry and Sergey named their system Google after the number 10^{100} —which called a “googol”—to reflect the fact that the web graph is so enormous.

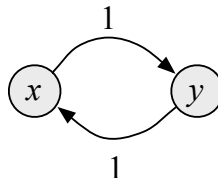
Anyway, now you can see how 6.042 ranked first out of 378,000 matches. Lots of other universities used our notes and presumably have links to the 6.042 open courseware site, and the university sites themselves are legitimate, which ultimately leads to 6.042 getting a high page rank in the web graph.

19.2.4 Problems

Class Problems

Problem 19.2.

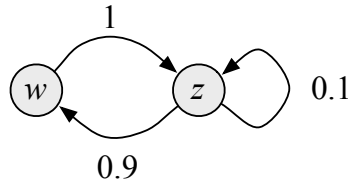
Consider the following random-walk graph:



(a) Find a stationary distribution.

(b) If you start at node x and take a (long) random walk, does the distribution over nodes ever get close to the stationary distribution? Explain.

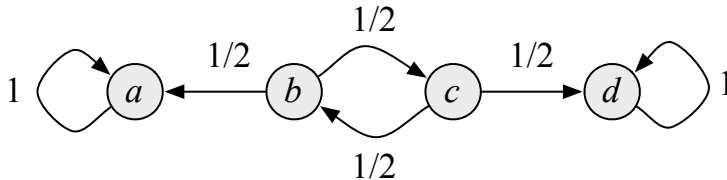
Consider the following random-walk graph:



(c) Find a stationary distribution.

(d) If you start at node w and take a (long) random walk, does the distribution over nodes ever get close to the stationary distribution? We don't want you to prove anything here, just write out a few steps and see what's happening.

Consider the following random-walk graph:



(e) Describe the stationary distributions for this graph.

(f) If you start at node b and take a long random walk, the probability you are at node d will be close to what fraction? Explain.

Homework Problems

Problem 19.3.

A digraph is *strongly connected* iff there is a directed path between every pair of distinct vertices. In this problem we consider a finite random walk graph that is strongly connected.

(a) Let d_1 and d_2 be distinct distributions for the graph, and define the *maximum dilation*, γ , of d_1 over d_2 to be

$$\gamma ::= \max_{x \in V} \frac{d_1(x)}{d_2(x)}.$$

Call a vertex, x , *dilated* if $d_1(x)/d_2(x) = \gamma$. Show that there is an edge, $y \rightarrow z$, from an undilated vertex y to a dilated vertex, z . *Hint*: Choose any dilated vertex, x , and

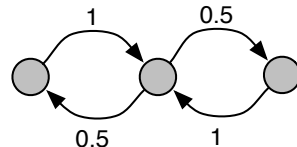
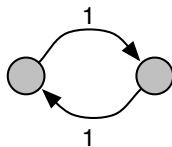
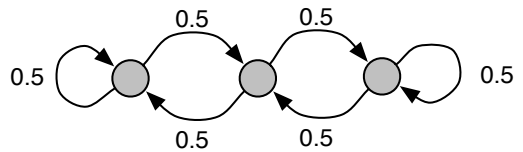
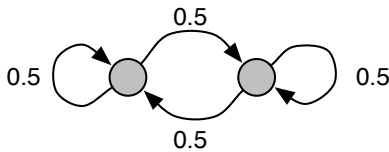
consider the set, D , of dilated vertices connected to x by a directed path (going to x) that only uses dilated vertices. Explain why $D \neq V$, and then use the fact that the graph is strongly connected.

(b) Prove that the graph has *at most one* stationary distribution. (There always *is* a stationary distribution, but we're not asking you prove this.) *Hint:* Let d_1 be a stationary distribution and d_2 be a different distribution. Let z be the vertex from part (a). Show that starting from d_2 , the probability of z changes at the next step. That is, $\hat{d}_2(z) \neq d_2(z)$.

Exam Problems

Problem 19.4.

For which of the following graphs is the uniform distribution over nodes a stationary distribution? The edges are labeled with transition probabilities. Explain your reasoning.



MIT OpenCourseWare
<http://ocw.mit.edu>

6.042J / 18.062J Mathematics for Computer Science
Spring 2010

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.