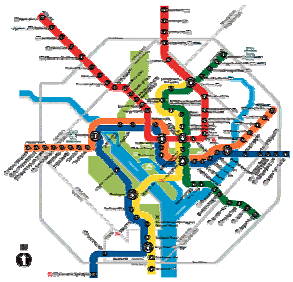
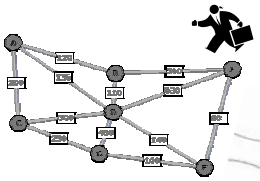
MA/CS-109: Graph Random Walks

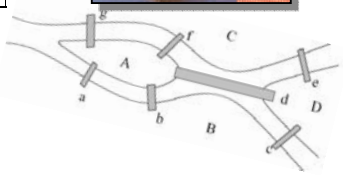
Azer Bestavros

Wandering around in a graph

- We saw that given a map (graph), one can determine paths that satisfy a specific objective
  - Minimize cost for going from A to B (Shortest Path)
  - Visit every node once (Traveling Salesman)
  - Cross every edge once returning to starting point (Eulerian circuit)





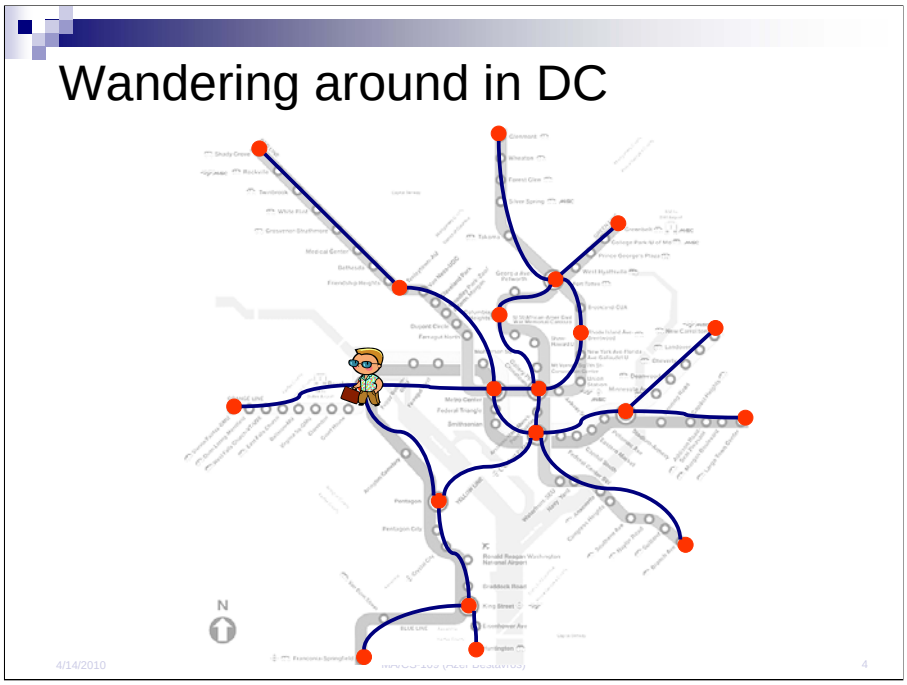


Recall our use of graphs as abstractions of real-world settings (specifically maps) in order to answer specific questions for somebody potentially using that map – for example, finding the shortest path between two points or finding a path that satisfies specific properties (e.g., visiting all nodes while crossing each edge exactly once).

Wandering around in a graph

- We saw that given a map (graph), one can determine paths that satisfy a specific objective
  - Minimize cost for going from A to B (shortest path)
  - Cross every edge once returning to starting point (Eulerian circuit)
- What if we have no objective?
  - A tourist roaming around the city
  - A web surfer aimlessly clicking from a web page to another

Now consider a situation where the “walker” (using the map) has no specific objective. He/she is just wandering around from node to node.



To illustrate this, consider a tourist moving around the subway system in DC where the tourists simply selects trips between nodes randomly – i.e., upon arrival to a station, the tourist selects the next leg randomly (from among all possible outgoing trips from that station. The above animation illustrates an example of what the tourist would be doing.

From the animation one can see that the tourist ends up going in and out some stations much more frequently than others. One can explain this intuitively by noting that the stations in the center of town are much better connected that the chances of wandering in them is higher (if one moves between stations at random).

So, here is a question we may wand to answer: What is the relative frequency with which a “random walker” ends up in one station versus another?

As we have done in this class, we want to answer this question not qualitatively using words like “more” or “much more”, but using some metric that we can be precise about.

Random walks on a graph

1. Start in some random node
2. At random, select one of the outgoing edges of the current node
3. Walk over that edge to a new node
4. Go to 2

- What properties would emerge as a result of doing the above for a *very very very* long time?

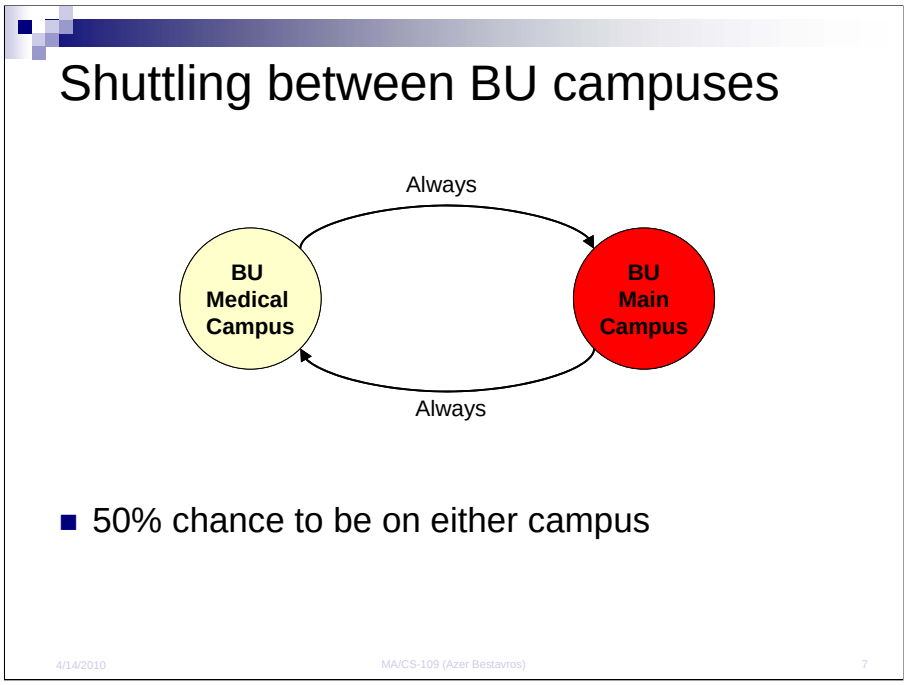
First, let's be specific about the model at hand.

A random walker starts at some random node in the graph, selects an outgoing edge from that node, traverses that edge and then repeats this process for a very very long time.

A specific question

- What are the relative frequencies (or probabilities) of visiting the various nodes in the graph.
  - What is the likelihood of finding the wandering tourist at a given subway station?
  - What is the likelihood that a wandering tourist will stumble upon the Museum of Science?
  - What is the likelihood that a web surfer will end up visiting Google? How about the BU web site?

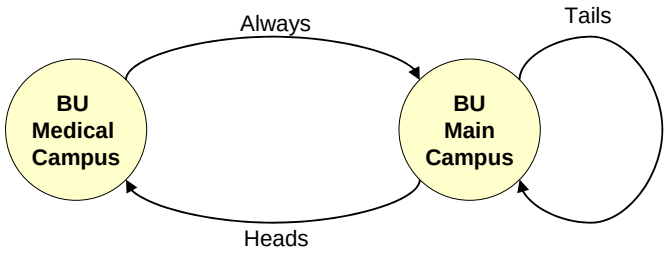
And here is the specific question we want answered: ***What are the relative frequencies of visiting the various nodes in the graph?*** (or stations in the DC Metro, or milestones in Boston, or pages on the web, …)



Let's start with a very simple scenario. Consider a BU student who shuttles between the BU Charles River Campus and the BU Medical Campus (using the BU shuttle). For simplicity, consider consecutive time periods (e.g., one hour) and assume that the students alternates between the two campuses every hour.

Now, if we ask the question "what is the relative frequency of the student being on the main campus versus the medical campus", the answer will be clearly 50% on each. This is obvious; we don't need math to figure this out!

Let's make it more interesting



- Imagine <u>many</u> random walkers (a population)
- Question: At some random point in time, what is the percentage of the population we should expect on each campus?

How about a slightly more complicated scenario.

Instead of one walker (student), consider many walkers (e.g., 1000).

Each one of the student does the following:

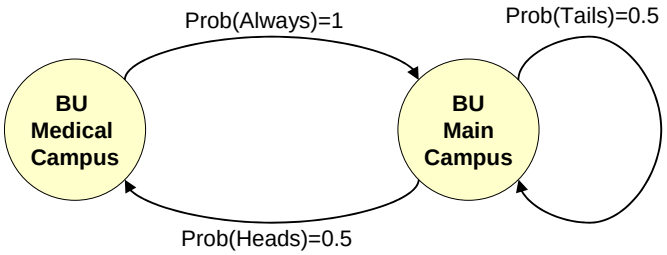
If in some hour, the student is on the main campus, he/she will flip a coin and if the result is heads, the student will hop on the shuttle and go to the medical campus for the next hour, otherwise (if the coin toss is tails) the student will just stay put. From the medical campus, the student always hops on the shuttle and goes back to the main campus.

Now, let's pose the question: At some random point in time in the far future, what is the percentage of the population (of 1,000) we should expect on each campus?

What would be our answer now?

Let's try to figure this out.

A simple example (continued)

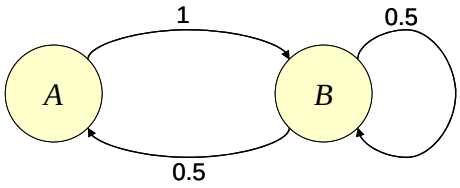


- Denote by $f_t(v)$ the population of vertex v in iteration t.
- Let us write down some relationships…

Let's denote by $f_t(v)$ the number of walkers at time t at node v.

A simple example (continued)

$$f_1(A) = 0.5\, f_0(B)$$
$$f_2(A) = 0.5\, f_1(B)$$
$$f_3(A) = 0.5\, f_2(B)$$
$$\ldots$$
$$f_t(A) = 0.5\, f_{t-1}(B)$$
$$\ldots$$
$$f_M(A) = 0.5\, f_{M-1}(B)$$



4/14/2010 MA/CS-109 (Azer Bestavros) 10

To simply things, let's denote the medical campus with the symbol A and the main campus with symbol B.

Assume that the population at time 0 on the main campus is $f_0(B)$. Since at time 1, everybody who was in A ends up leaving A and half of those who were in B at time 0 end up in A at time 1, we can write the relationship:

$$f_1(A) = 0.5\, f_0(B)$$

Now with a similar logic, we can see that this will be the case for all time periods 1, 2, 3, … all the way to a very very large number – let's call it "M".

Doing so, gives us the relationships shown on the slide, which relate the population on campus A in terms of the population on campus B in the previous time interval.

Adding up all the terms on both sides of the equations, we get…

A simple example (continued)

$$f_1(A)+f_2(A)+f_3(A)+\ldots f_M(A)=$$
$$0.5\left[f_0(B)+f_1(B)+f_2(B)+\ldots+f_{M-1}(B)\right]$$

$$f_1(A)+f_2(A)+f_3(A)+\ldots f_M(A)=$$
$$0.5\left[f_0(B)+f_1(B)+f_2(B)+\ldots+f_{M-1}(B)+f_M(B)-f_M(B)\right]$$

$$f_1(A)+f_2(A)+f_3(A)+\ldots f_M(A)=$$
$$0.5\left[f_1(B)+f_2(B)+\ldots+f_{M-1}(B)+f_M(B)\right]+0.5\left[f_0(B)-f_M(B)\right]$$

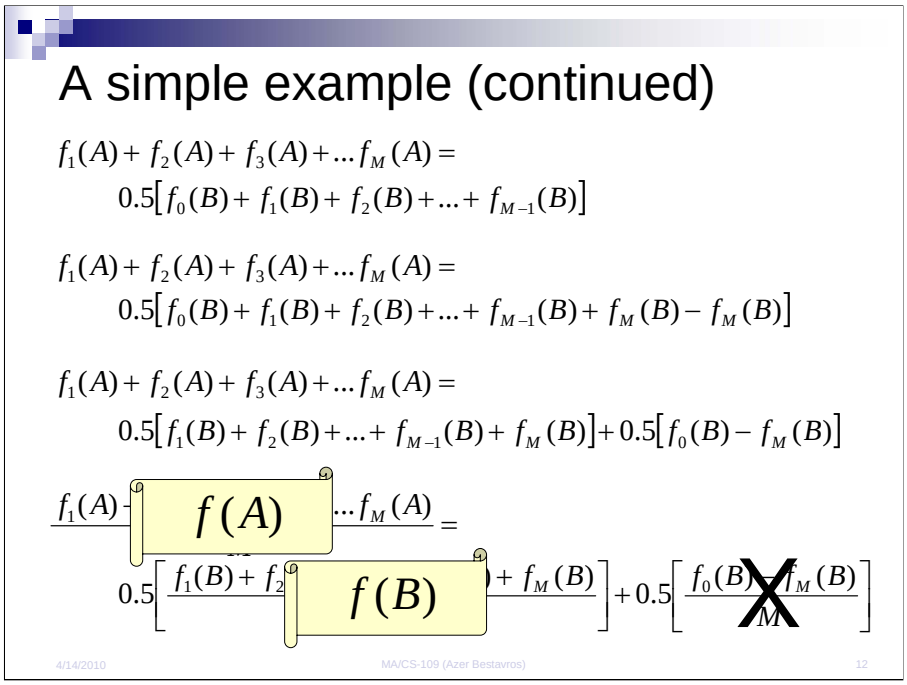
$$\frac{f_1(A)+f_2(A)+f_3(A)+\ldots f_M(A)}{M}=$$
$$0.5\left[\frac{f_1(B)+f_2(B)+\ldots+f_{M-1}(B)+f_M(B)}{M}\right]+0.5\left[\frac{f_0(B)-f_M(B)}{M}\right]$$

4/14/2010 MA/CS-109 (Azer Bestavros) 11

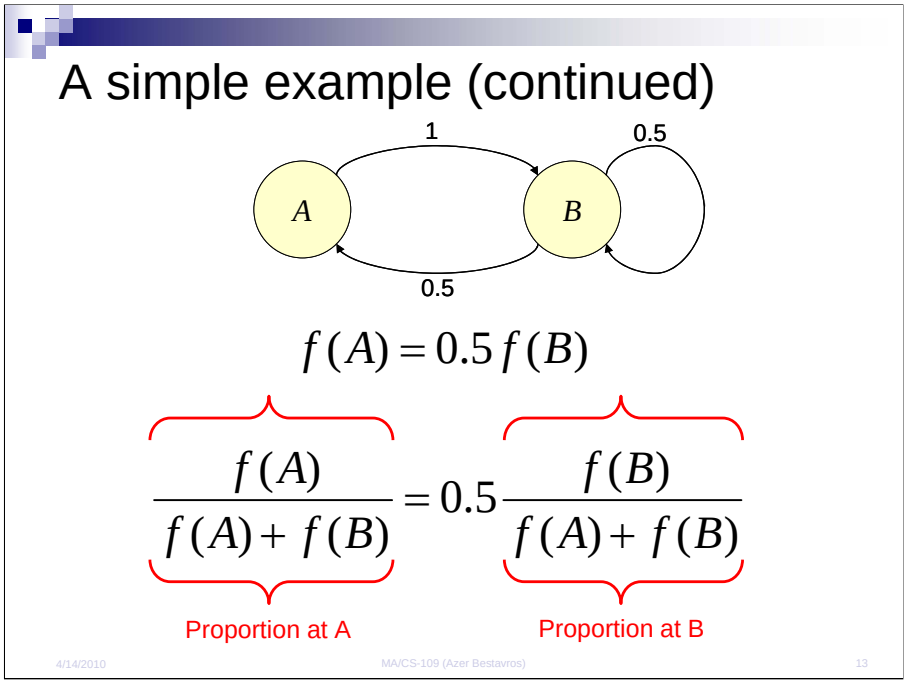
$$f_1(A)+f_2(A)+f_3(A)+\ldots f_M(A)=0.5\left[f_0(B)+f_1(B)+f_2(B)+\ldots+f_{M-1}(B)\right]$$

Using simple algebraic manipulation (by adding a quantity and subtracting it from the right hand side, rearranging terms, and then dividing both sides by M, we get

$$\frac{f_1(A)+f_2(A)+f_3(A)+\ldots f_M(A)}{M}=0.5\left[\frac{f_1(B)+f_2(B)+\ldots+f_{M-1}(B)+f_M(B)}{M}\right]+0.5\left[\frac{f_0(B)-f_M(B)}{M}\right]$$



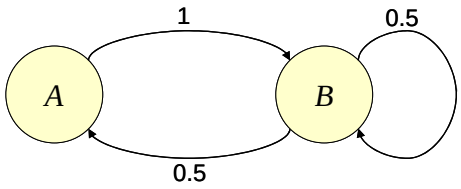
We note that the left-hand-side of the equation is the sum of the population in A in all time intervals (from 1 to M) divided by M, which is precisely the average size of the population in A over time, which we denote by $f(A)$. Similarly we can substitute on the right hand side for the value of $f(B)$. This leaves us with the second term on the right hand side. Since M is very very large (compared to the difference between two numbers that are close to each other -- the population of B at time 0 and at time M), we can simply ignore that term as it tends to zero.



Now, dividing both sides by the total population, which is $f(A)+f(B)$, which gives us the proportion of the population at each node, which we denote bt *Pr(A)* and *Pr(B)* – which is also the probability of a walker being in A versus B, respectively.

Thus, we can write the relationship *Pr(A) = 0.5 Pr(B).*

A simple example (continued)



$$\Pr(A) = 0.5\Pr(B)$$

$$\Pr(A) + \Pr(B) = 1$$

$$\Pr(A) = \frac{1}{3} \quad \Pr(B) = \frac{2}{3}$$

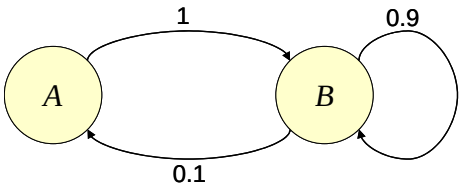
Another relationship we can write is that the sum of the two proportions, i.e., Pr(A) + Pr(B) , must be equal to 1.

So, we have two equations in two unknowns.

By substituting for Pr(A) from the first equation into the second, we get one equation in one unknown, which gives us the value of Pr(B) = 2/3. This means that Pr(A) which is half of Pr(B) is 1/3.

That's it, we figured it out. We would expect one third of the population in A (the medical campus) and two thirds of the population in B (the main campus).

How about this one?



$$\Pr(A) = 0.1\Pr(B)$$

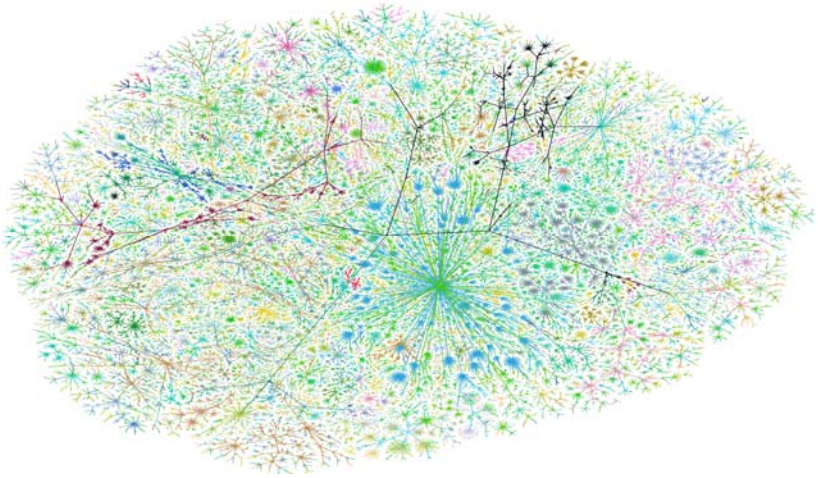
$$\Pr(A) + \Pr(B) = 1$$

$$\Pr(A) = 9\% \quad \Pr(B) = 91\%$$

We can also consider a scenario where the walkers (students) choose to stay on the main campus 9 out of every 10 times and choose to shuttle to the medical campus 1 out of every10 times.

Following the exact same process, we conclude that 0.91 (i.e., 91%) of the population will be on the main campus with the rest (i.e., 9%) on the medical campus.

How about this one?

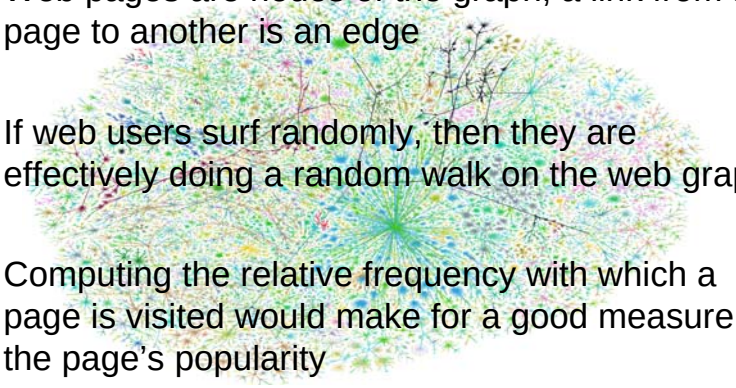


Why is it interesting to randomly walk web graphs?

Well, these were toy examples!
How about a very very large graph – the web graph?

Random walks on web graphs

- Web pages are nodes of the graph; a link from a page to another is an edge
- If web users surf randomly, then they are effectively doing a random walk on the web graph
- Computing the relative frequency with which a page is visited would make for a good measure of the page's popularity

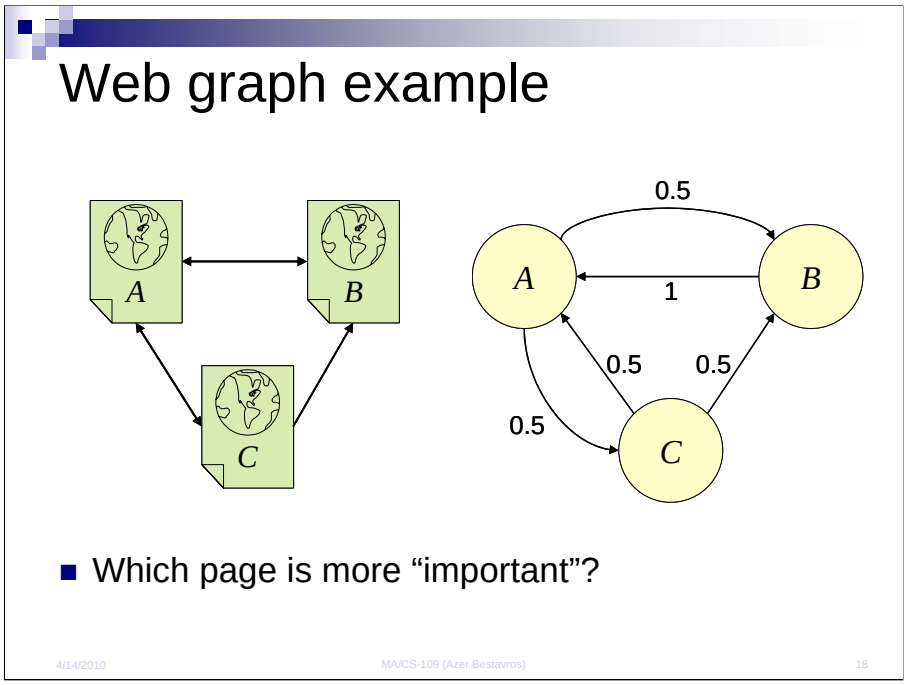


The web graph is made up of nodes (the web pages) and directed edges (the links from one page to another).

If web users surf the web randomly, going from one page to the next by picking one of the links from that page at random, then they are effectively doing a random walk on the web graph!

Now we can ask the question: What percentage of web surfers would end up in one page versus another?

This quantity could be interpreted as the "popularity" of a web page. This is a very valuable thing to know since it could be used by advertisers to decide the worth of putting an ad on one page (e.g., cnn.com) versus another (e.g., msnbc.com).



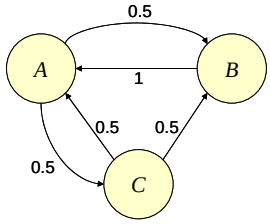
Just to illustrate this, consider a web of simply three pages, with the specific link structure shown (on the left).

From that structure, we can infer the web graph (on the right) where we labeled the edges with the probability of traversing that edge. For example, since out of page A there are two outgoing links, and since random web surfers (graph walkers) would “equally likely” pick one of these, we label them both with a probability of 0.5.

Now we ask the question: What is the relative popularity of pages A, B, and C? Which one of these pages is more popular than the others?

Web graph example

$$\Pr(A) + \Pr(B) + \Pr(C) = 1$$
$$\Pr(A) = \Pr(B) + 0.5\Pr(C)$$
$$\Pr(C) = 0.5\Pr(A)$$



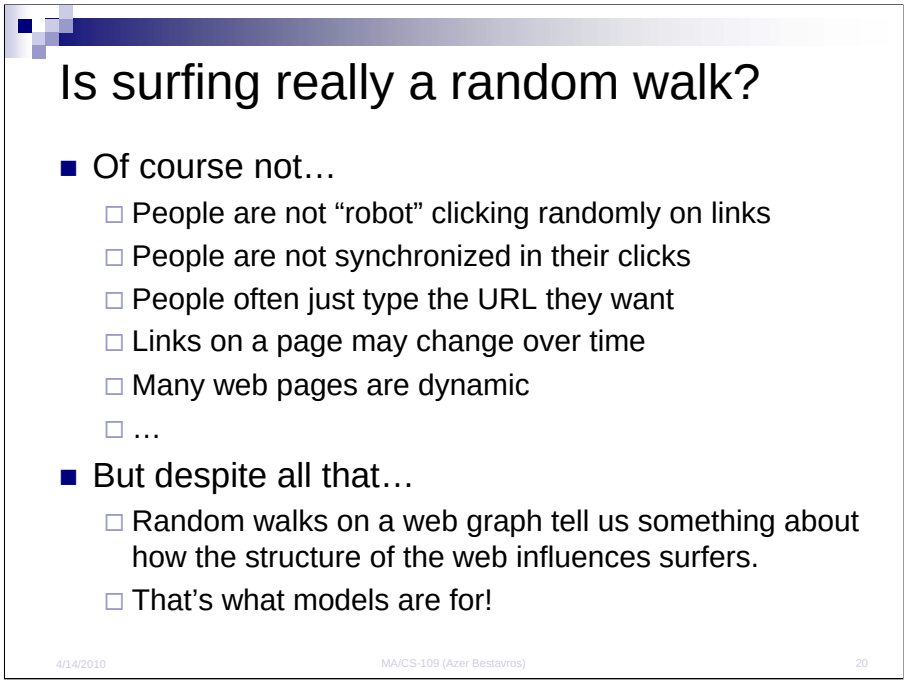
$$\Pr(A) = 0.444$$
$$\Pr(B) = 0.333$$
$$\Pr(C) = 0.222$$

By following the same process as before, we can write a set of equations. Since we have three unknowns – Pr(A), Pr(B), and Pr(C) – we must have a set of 3 equations.

The first equation is always the easiest one to write – that the sum of the relative popularity is 1. Now, by looking at one of the states (say A) we can say that at any time interval, the population in A will be composed of everybody who was in B + half those in C in the previous time interval. This gives us a second equation. Doing the same for another state (say C), we get the third equation.

With 3 equations in 3 unknown, we can find the relative popularity of A, B, and C – namely, Pr(A) = 0.444, Pr(B)=0.333, and Pr(C)=0.222.

So, page A is the most popular – it is the one in which we would expect that most random web surfers will be.



But, is a “random walk” the best way to model surfing behavior?

Clearly, people are not “robot” clicking randomly on links; they do not go from one page to the other every time interval (e.g., every minute); they are not synchronized in their actions; they may visit a web page by typing its URL in the browser, or by searching for it and then clicking on the search results. Moreover, the web structure itself is not a static graph since links change as web pages change (when you add a link to your Facebook page, you are effectively changing the structure of the web).

These are all valid points, but despite all that, the random walk model is a very good abstraction that allows us to come up with precise answers.

Notice that trying to capture all the nuances of how people surf the web, will result in a very complicated object that would be rather impossible to “solve” (or to comprehend in the first place).

This is precisely why we use abstractions and models – it is to simplify the real world so that we may be able to answer questions!

Random walks on web graphs

- But, the web graph has ~ 100 billion pages! We are not about to write 100 billion equations and solve them?!
- We can simulate the random walk and measure the frequency of visits to the various pages.
- This process is at the heart of Google's *"PageRank"* algorithm (and ~ $400 stock value).

So far, we figured out how to convert a real-world object (maps or web) to a graph and then using random walks solve analytically for the "popularity" of various nodes in the graph (stations or web pages).

But the graphs corresponding to real-word objects are HUGE. The web has hundreds of billions of pages! We are not about to write 100 billion equations and expect to solve them algebraically!!!

The way we deal with this is to "simulate" the random walkers!

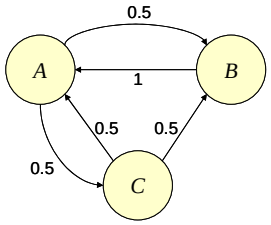
BTW, this technique is at the heart of the Google search technology – a technology that propelled one of the most successful companies in hostory!

Random walk simulation

1. Initialize:
   Assume that some arbitrary population at each node

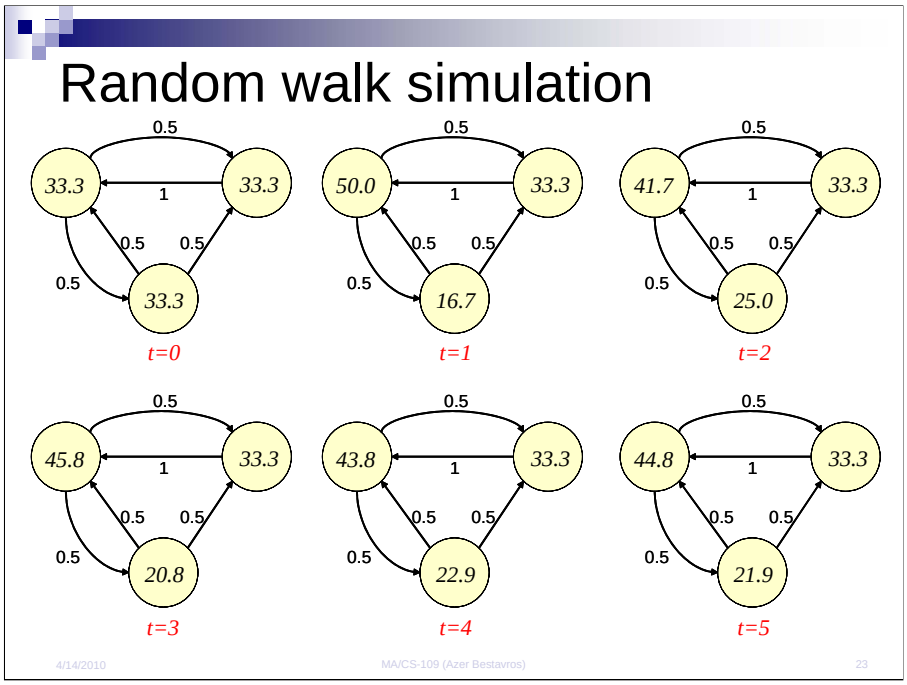
2. Compute new population:
   Use edge probabilities to calculate the population at the next time interval

3. Check for change:
   If population changed then go to 2



So, what do we mean by “simulating random walkers”?

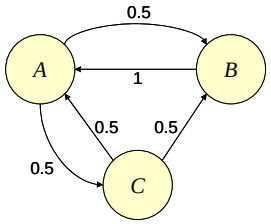
It’s simple, just start with some arbitrary population at each node. Now, each time step, calculate the new population at each node, and keep doing this until things stabilize (or until we get tired). This would give us the relative popularity of the various nodes!



Here is an example. We started with 100 people, whom we divided equally among the three nodes (web pages) at time t=0. Now we do the simulation for a number of steps and we notice how the population seems to “stabilize” with A attracting around 44 people, B attracting 33, and C attracting 22.

Random walk simulation

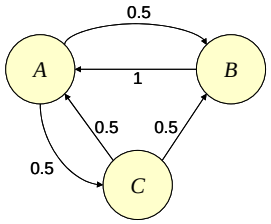
Iteration	Pr(A)	Pr(B)	Pr(C)
0	33.33	33.33	33.33
1	50.00	33.33	16.67
2	41.67	33.33	25.00
3	45.83	33.33	20.83
4	43.75	33.33	22.92
5	44.79	33.33	21.88
6	44.27	33.33	22.40
7	44.53	33.33	22.14
8	44.40	33.33	22.27
9	44.47	33.33	22.20
10	44.43	33.33	22.23
11	44.45	33.33	22.22
12	44.44	33.33	22.22
13	44.45	33.33	22.22
14	44.44	33.33	22.22
15	44.44	33.33	22.22



Indeed, the spreadsheet shown is exactly the resulting evolution of the population and as we can see, by the 11th time step, the population stabilizes and we have our answer.

Random walk simulation

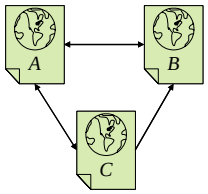
Iteration	Pr(A)	Pr(B)	Pr(C)
0	0.33	0.33	0.33
1	0.50	0.33	0.17
2	0.42	0.33	0.25
3	0.46	0.33	0.21
4	0.44	0.33	0.23
5	0.45	0.33	0.22
6	0.44	0.33	0.22
7	0.45	0.33	0.22
8	0.44	0.33	0.22
9	0.44	0.33	0.22
10	0.44	0.33	0.22
11	0.44	0.33	0.22
12	0.44	0.33	0.22
13	0.44	0.33	0.22
14	0.44	0.33	0.22
15	0.44	0.33	0.22



Rather than reporting the number of people we expect to see at a node, it is customary to report the fraction of the population at each node. This is nothing but the population at the node divided by the total population – a number between 0 and 1, which we can think of as the probability that a single person will end up in a node. Indeed, we would get the same answer if we start with any initial population!

(Larry) PageRank

PageRank relies on the uniquely democratic nature of the web by using its vast link structure as an indicator of an individual page's value. In essence, Google interprets a link from page A to page B as a vote, by page A, for page B. But, Google looks at more than the sheer volume of votes, or links a page receives; it also analyzes the page that casts the vote. Votes cast by pages that are themselves "important" weigh more heavily and help to make other pages "important".



Source: Google

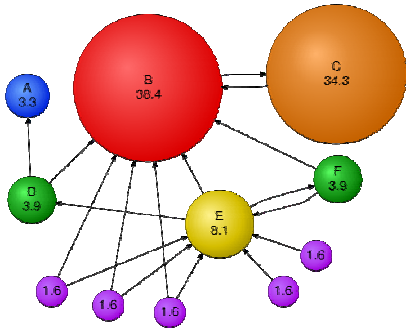
Using random walks to estimate the popularity of various web pages (based on the structure of the web – i.e., how the various pages are linked to one another) was the value proposition for Google search. By sorting the various pages returned from a keyword search based on their relative popularity, users found the search results to be much more useful – you really don't want to scroll down to the 10th page of search results to find what you are looking for.

Indeed, Google makes an interesting observation: The structure of the web reflects "votes" from one page to another. If a page points to another, then this constitutes a vote by the referring page for the referred-to page. If you post a link on your Facebook page, you are effectively saying that "this link is interesting".

Google search uses a slightly modified version of random walks by recognizing that the popularity of a web page has two sources – the first is due to random walking (as we have done so far) and the second is due to people going to the web page directly (e.g., by typing a URL directly, instead of clicking on a link from another page). The resulting algorithm is PageRank, which is due to "Larry Page" a co-founder of Google. PageRank attributes a weight to the two sources of web page popualrity and computes a score based on the joint contribution of these two sources. While we will not cover how PageRank does this, it suffices to say that it is very very similar to random walks. So, for all practical purposes, you can simply think of PageRank as the equivalent of evaluating popularity using random walks.

Ashton Kutcher vs CNN

- It is not only how many "followers" you have, but how "important" these followers are!



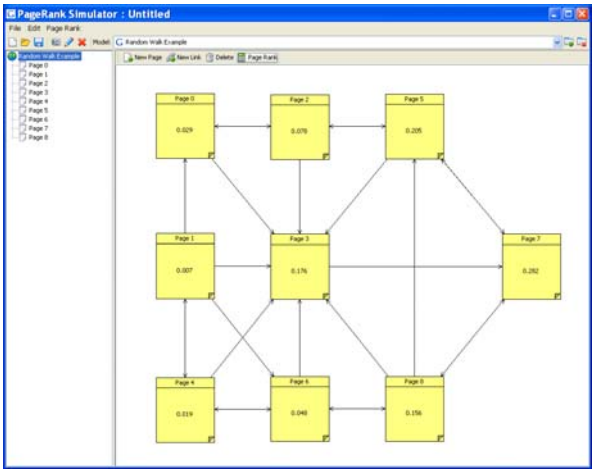
- Important implications for evaluating scholarship!

An interesting point to note is that the popularity of a web page (according to a random walk evaluation or PageRank evaluation) is not necessarily dependent on the number of other pages pointing to it, but in fact it is dependent on the popularity of those pages pointing to it as well. This is why the Twitter "debate" about who is more popular "Ashton Kutcher" or "CNN" based on the number of "followers" of each does not make sense!!!

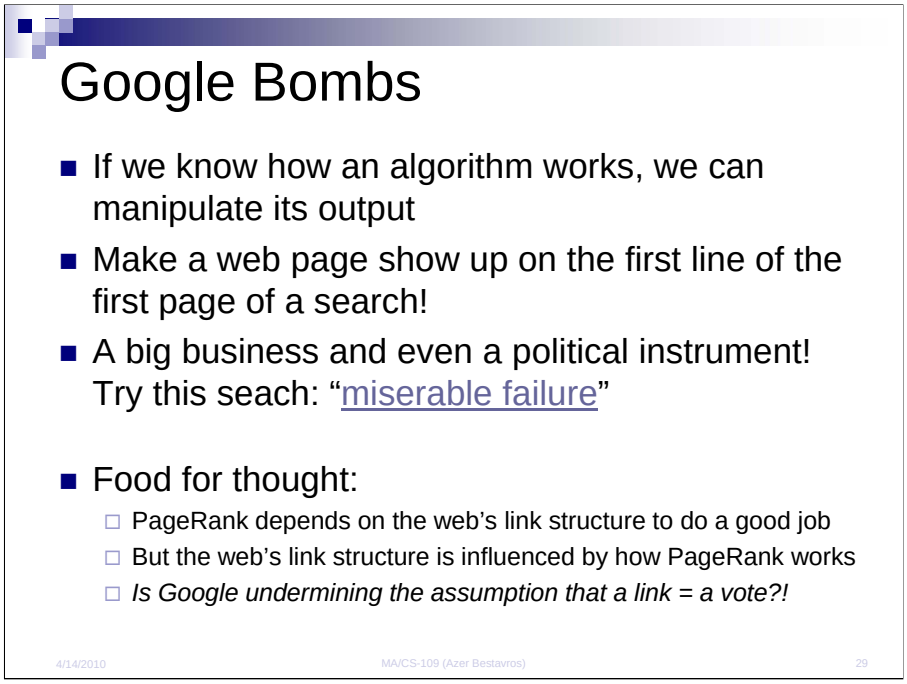
According to random walks (and Google), "not all web pages are created equal and this a vote from an important web page should count for more than a vote from an insignificant web page".

This concept has significant implications for evaluating scholarship – do we simply count the number of citations to a paper, or do we take into account the importance/popularity of the paper making the citation?

PageRank Simulation



Demo of PageRank simulations… Simulator available through Moodle.



One last point to make. The fact that we know how Google sorts web pages means that we can manipulate it! This is precisely what people did in the early days of web search technologies and what has become known as "Google Bombing" or "Google Bombs". People who "make" Google bombs do so by creating a bunch of web non-sense web pages that link to a page that they want to artificially raise its popularity.

The most famous of Google Bombs is the "miserable failure" bomb! Just Bing "miserable failure" to see who comes up very high on the list of returned web pages (hint: it is a US president ☺)

Finally, we make the following (philosophical) observation: PageRank depends on the web's link structure to do a good job. But the web's link structure is influenced by how PageRank works -- e.g., people are likely to point to pages that Google ranks highly (because they use Google to find these pages). This means that Google is introducing bias in the web structure! This raises the following intriguing question: *Is Google undermining the assumption that a link = a vote?!! Is Google undermining its own business proposition?! Is a software program (PageRank) influencing the evolution of our society, our public opinion, etc.? This is scary in many ways – it suggests that algorithms are acting like "DNA"; they affect how we evolve as a society!*