CS-621 Theory Gems October 17, 2012

Lecture 9

Lecturer: Aleksander Mądry *Scribes: Dorina Thanou, Xiaowen Dong*

1 Introduction

Over the next couple of lectures, our focus will be on graphs. Graphs are one of the most basic objects in computer science. They have plethora of applications – essentially, almost any CS problem can be phrased in graph language (even though, of course, it is not always the best thing to do).

Formally, we view a graph $G = (V, E)$ as consisting of a set V of *vertices* with a set E of *edges* that connect pairs of vertices, i.e., we have $E \subseteq V \times V$. For an edge $e = (v, u)$, we call v and u the *endpoints* of the edge e. Sometimes, we consider graphs that are *directed*, i.e., we assume that each edge $e = (v, u)$ is directed from its *tail* v to its *head* v. (Note that, as a result, in directed graphs the edge (u, v) is different than the edge (v, u).) To emphasize this difference, we usually call the edges of a directed graph *arcs*. In Figure 1, we see an example of an undirected and directed graph. We will usually denote the size $|V|$ of a vertex set of a graph by n and the size $|E|$ of its edge set by m.

Finally, from time to time, we will be considering *weighted* graphs $G = (V, E, w)$, in which we additionally have a *weight vector* w that assigns an non-negative weight w_e to each edge e.

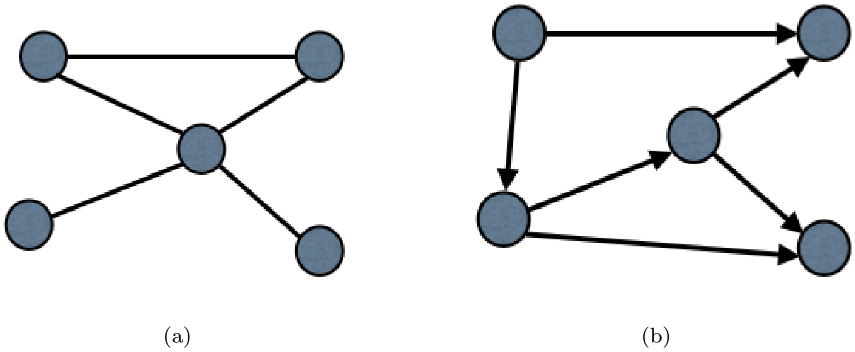


Figure 1: (a) Undirected graph (b) Directed graph

1.1 How To Explore a Graph?

Given a graph $G = (V, E)$, one of the most basic graph primitives is exploration of that graph from a particular vertex s. There are many different ways to do this – most popular ones are the depth-first and breath-first search – and depending on applications some of them might be more preferable than the others.

Today, we want to introduce yet another way of exploring the graph – via so-called *random walks* – that is fundamentally different than the most obvious ones, but turns out to provide us with a considerable insight into the structure of the underlying graphs.

2 Random Walks in Graphs

Given an undirected graph $G = (V, E)$ and some *starting vertex* s, a random walk in G of length t is defined as a randomized process in which, starting from the vertex s, we repeat t times a step that consists of choosing at random one of the neighbors v' of the vertex v we are at and moving to it. (If the graph is weighted then we transition to a neighbor v' with probability that is proportional to the weight w_e of the connecting edge $e = (v, v')$.)

Note that this definition is also valid for a directed graph (we just always pick an arc that is leaving v, i.e., has v as its tail), but – for reasons that will become clear later – we focus our treatment on undirected case.

2.1 Random Walk as a Distribution

Clearly, random walks as defined above are naturally viewed as trajectories determined by the outcome of randomized choices we make at each step. However, today we want to focus on a more global view on them and thus we will keep track not of these trajectories directly, but instead of the probability distribution on vertices that random walks induce.

More precisely, for a given vertex v, we define p_v^t to be the probability that we are at vertex v after t steps of the random walk. For notational convenience, we will represent all these probabilities as a single n-dimensional vector $p^t \in \mathbb{R}^V$ with the v^{th} coordinate corresponding to p_v^t. Clearly, if our random walks starts from some vertex s, the initial distribution p^0 of the random walk can be defined as

$$p_v^0 = \begin{cases} 1, & \text{if } v = s \\ 0, & \text{otherwise.} \end{cases}$$

Now, we can express the evolution of the distribution p^t as

$$p_v^{t+1} = \sum_{e \in E, e=(u,v)} \frac{1}{d(u)} p_u^t \quad \text{for each } v \in V, \tag{1}$$

where $d(u)$ is the *degree*, i.e., number of adjacent edges, of vertex u. (If the graph is weighted, we want $d(u)$ denote the *weighted degree* of vertex u, that is, the sum of weights of adjacent edges, and then $p_v^{t+1} = \sum_{e \in E, e=(u,v)} \frac{w_e}{d(u)} p_u^t$.)

Intuitively, the evolution of the probability distribution p^t as a function of t can be seen as describing a diffusion processes in the underlying graph.

2.2 Lazy Random Walk

It will be sometime convenient to consider a slight variation of random walk, in which in each step, with probability 1/2, we stay at the current vertex and only with probability 1/2 we make the usual step of random walk. This variation is called *lazy random walk* and it can be viewed as a vanilla version of random walk in a graph in which we added $d(u)$ self-loops to every vertex u.

Formally, the evolution of the probability distribution $\hat{p}^t$ of such a lazy random walk is given by

$$\hat{p}_v^{t+1} = \frac{1}{2} \hat{p}^t + \frac{1}{2} \left(\sum_{e \in E, e=(u,v)} \frac{1}{d(u)} \hat{p}_u^t \right) \quad \text{for each } v \in V. \tag{2}$$

2.3 Matrix Definitions

As we will see soon, linear algebra is the "right" language for analyzing random walks. So, to enable us to use this language, we want to express the evolution of random walks in purely linear-algebraic terms. To this end, given a graph $G = (V, E)$, let us define the *adjacency matrix* A of G as an n-by-n matrix with rows and columns indexed by vertices and

$$A_{u,v} = \begin{cases} 1 & \text{if } (u,v) \in E \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Now, we define the *walk matrix* W of G as an n-by-n matrix given by

$$W := AD^{-1},$$

where D is the *degree matrix* of G, i.e., a diagonal n-by-n matrix whose each diagonal entry $D_{v,v}$ is equal to the degree $d(v)$ of vertex v.

It is easy to verify that the walk matrix W can be explicitly written as

$$W_{u,v} = \begin{cases} \frac{1}{d(v)} & \text{if } (u,v) \in E \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

As a result, we can compactly rewrite the evolution of the probability distribution p^t (cf. Equation (1)) as

$$p^{t+1} = Wp^t = W^t p^0. \tag{5}$$

Similarly, in case of lazy random walk, if we define *lazy walk matrix* $\widehat{W}$ of G to be a matrix

$$\widehat{W} := \frac{1}{2}I + \frac{1}{2}W, \tag{6}$$

where I is the identity matrix, we can conveniently express the evolution of the distribution $\hat{p}^t$ (cf. Equation (2)) as

$$\hat{p}^{t+1} = \widehat{W}\hat{p}^t = \widehat{W}^t \hat{p}^0. \tag{7}$$

Finally, if the graph $G = (V, E, w)$ is weighted then one just needs to adjust the definitions of the adjacency matrix A and degree matrix D to make $A_{u,v} := w_e$ if $e = (u, v)$ is an edge in G, and $D_{v,v}$ to be equal to the weighted degree of vertex v.

3 Stationary Distribution

The main goal of today's lecture is understanding the behavior of the (lazy) random walk distributions p^t and $\hat{p}^t$ for large t. At first, given the randomized character of the random walk and its crucial dependence on graph topology, one might expect this behavior to be very complex and hard to describe.

However, somewhat strikingly, it turns out that the shape of the distributions p^t and $\hat{p}^t$ for $t \to \infty$ is very easy to describe and depends only mildly on the topology of the graph. (As we will see later, it is actually the behavior of these distributions for moderate values of t that really carries more substantial information about the graph structure.)

The key object that will be relevant here is the so-called *stationary distribution* π of an (undirected) graph G given by

$$\pi_v := \frac{d(v)}{\sum_{u \in V} d(u)}. \tag{8}$$

Intuitively, in this distribution, the probability of being at vertex v is proportional to its degree.[1]

The crucial property of this distribution is that

$$W\pi = \widehat{W}\pi = \pi, \tag{9}$$

i.e., once the distribution p^t or $\hat{p}^t$ is equal to π, for some t, it will remain equal for all $t' \geq t$. We can thus view the stationary distribution as a "steady state" of a random walk in G.

Now, the surprising fact about random walks is that in every connected non-bipartite graph the random walk distribution p^t on this graph is actually guaranteed to converge to π when $t \to \infty$. No

[1] A stationary distribution can also be defined for directed graphs, but its form and conditions for existence are slightly more involved.

matter what the starting distribution p^0 is. (A graph $G = (V, E)$ is *bipartite* if its set of vertices V can be partitioned into two disjoint sets P and Q such that all the edges in E have one endpoint in P and one in Q, i.e., $E \subseteq P \times Q$.)

The reason why there is no convergence when the graph is not connected is obvious. On the other hand, to see why there might be no convergence to π when the graph is bipartite, consider p^0 to be a distribution that is supported only on one side of the bipartition, i.e., only on P or only on Q. Then, in each step of random walk, the support of p^t will move entirely to the other side and thus, in particular, p^t will be always different for ts with different parity.

It is not hard to see that the above problem with bipartite graphs is no longer present when we consider the lazy variant of random walks. Indeed, for the lazy random walk distribution $\hat{p}^t$ such convergence happens always – even in bipartite graphs – as long as the underlying graph is connected. (This is one of the main reasons why it is more convenient to work with lazy random walks instead of the vanilla ones.)

3.1 Spectral Matrix Theory

Our goal now is to prove the convergence properties stated above. Looking at Equations (5) and (7), one can see that understanding the distributions p^t and $\hat{p}^t$ for large t is equivalent to understanding the product of a power of the walk matrices W^t and $\widehat{W^t}$ with the vector p^0, for large values of t. It turns out that there is a very powerful linear-algebraic machinery – called *spectral matrix theory* – that can be used for this purpose.

To introduce this tool, let us define a $\lambda \in \mathbb{R}$ to be an *eigenvalue* of a square matrix M iff there exists some vector v such that

$$Mv = \lambda v.$$

The vector v is called the *eigenvector* of M corresponding to the eigenvalue λ.

Although every real n-by-n matrix M has n eigenvalues (but not necessarily real ones!), when M is *symmetric*, i.e., $M_{i,j} = M_{j,i}$, for all i, j, then these eigenvalues are especially nice. Namely, we have then that

1. M has exactly n eigenvalues $\lambda_1, \ldots, \lambda_n$ (some of these λ_is can be equal) that are real.
2. One can find a collection $v^1, \ldots, v^n$ of eigenvectors of M, where v^i is the eigenvector corresponding to eigenvalue λ_i, such that all v^is are orthogonal, i.e., $(v^i)^T v^j = 0$ if $i \neq j$.

Now, the fundamental importance of existence of the full set of n eigenvalues (and corresponding eigenvectors) of the matrix M lies in that we can express any power (in fact, any real function) of M in a very convenient way via its eigenvalues and eigenvectors. Namely, we have that

$$M^t = \sum_i \lambda_i^t v^i (v^i)^T.$$

(More generally, for any real function f of M, we have $f(M) = \sum_i f(\lambda_i) v^i (v^i)^T$.)

So, a product of M^t and any vector v is just equal to

$$M^t v = \sum_i \lambda_i^t v^i (v^i)^T v,$$

which is an expression whose only parts that depend on t are the powers of eigenvalues and, as a result, understanding the evolution of $M^t v$ as a function of t boils down to understanding the eigenvalues of M.

3.2 Proof of Convergence

In the light of the above discussion, we would like now to analyze the distribution p^t (resp. $\hat{p}^t$) by treating it as a product of t-th power of the matrix W (resp. $\widehat{W}$) and the vector p^0, and applying to it the spectral approach we introduced. Unfortunately, the problem is that the matrices W and $\widehat{W}$ might not necessarily be symmetric and thus they might, in particular, not even have real eigenvalues.

It turns out, however, that there is a way to remedy this problem, provided that the graph is undirected. (When the graph is directed, it might actually really be the case that the walk matrices do not have eigenvalues and thus analyzing them becomes more challenging – this is the main reason why we focus on undirected graph case.) To this end, we consider the following matrix S

$$S := D^{-\frac{1}{2}}AD^{-\frac{1}{2}} = D^{-\frac{1}{2}}WD^{\frac{1}{2}}.$$

It is easy to see that S is symmetric, as for undirected graphs the adjacency matrix A is symmetric. So, S has to have a set of n eigenvalues $\omega_1 \geq \ldots \geq \omega_n$ (for notational convenience, we numbered them in non-increasing order) and corresponding set of orthogonal eigenvectors $v^1, \ldots, v^n$.

Now, the crucial observation is that, for any i,

$$W(D^{\frac{1}{2}}v^i) = D^{\frac{1}{2}}SD^{-\frac{1}{2}}D^{\frac{1}{2}}v^i = D^{\frac{1}{2}}Sv^i = D^{\frac{1}{2}}\omega_i v^i = \omega_i(D^{\frac{1}{2}}v^i),$$

i.e., ω_i is an eigenvalue of the walk matrix W corresponding to an eigenvector $D^{\frac{1}{2}}v^i$.

This proves that W has a full set of n real eigenvalues (even though it is, in general, not symmetric!) and thus we can write p^t as

$$p^t = W^t p^0 = (D^{\frac{1}{2}}SD^{-\frac{1}{2}})^t p^0 = D^{\frac{1}{2}}(S^t D^{-\frac{1}{2}}p^0) = D^{\frac{1}{2}}(\sum_{i=1}^{n} \omega_i^t v^i (v^i)^T D^{-\frac{1}{2}}p^0), \tag{10}$$

in which, again, the only terms depending on t are the powers of the eigenvalues ω_i.

So, we need now to bound the values of ω_i to be able to understand how the limit of p^t for $t \to \infty$ looks like. To this end, we prove the following lemma.

Lemma 1 *For any connected graph $G = (V, E)$, we have that $|\omega_i| \leq 1$, for each i, ω_1 is equal to 1, and π is an eigenvector of W corresponding to this eigenvalue. Furthermore, $\omega_i < 1$ for $i > 1$.*

Proof The fact that $\omega_1 = 1$ and π is the corresponding eigenvector follows directly from the property of the stationary distribution – cf. (9). Now, to see that $|\omega_i| \leq 1$, for all i, let $y^i = D^{\frac{1}{2}}v^i$ be an eigenvector of W corresponding to the eigenvalue ω_i. Let u be the vertex that maximizes

$$\frac{|y^i_u|}{d(u)},$$

among all the vertices of G.

By definition of ω_i, we have that

$$\omega_i y^i_u = \sum_{(w,u)\in E} \frac{y^i_w}{d(w)}, \tag{11}$$

and thus

$$|\omega_i||y^i_u| \leq \sum_{(w,u)\in E} \frac{|y^i_w|}{d(w)} \leq \sum_{(w,u)\in E} \frac{|y^i_u|}{d(u)} = |y^i_u|,$$

where in the second inequality we used the maximality of u.

So, indeed $|\omega_i| \leq 1$.

To show that $\omega_i < 1$ when $i > 1$, let u be the vertex that maximizes

$$\frac{y^i_u}{d(u)},$$

among all the vertices of G. Note that if $\omega_i = 1$ then the only way the Equality (11) can hold is if

$$\frac{y^i_u}{d(u)} = \frac{y^i_w}{d(w)},$$

for all the neighbors of u in G. But, by repeating this argument iteratively and using the fact that G is connected, we can conclude that the above equality holds for any two vertices of the graph. This, in turn, means that

$$y^i = C\pi,$$

for some C, i.e., y^i has to be a scalar multiple of the stationary distribution vector π. In this way, we have shown that any eigenvector of W corresponding to the eigenvalue of 1 has to be co-linear with π, which implies that there can be only one eigenvalue equal to 1. ■

In one of the problem sets, you will be asked to prove that W has an eigenvalue equal to -1 only if the graph G is bipartite. So, in the case of G being connected and non-bipartite, we have that $|\omega_i| < 1$ for $i > 2$ and thus in the limit of $t \to \infty$ the powers of these eigenvalues will vanish. That is, (10) becomes

$$p^t = W^t p^0 = D^{\frac{1}{2}}(\sum_{i=1}^{n} \omega_i^t v^i (v^i)^T D^{-\frac{1}{2}} p^0) \overset{t\to\infty}{=} D^{\frac{1}{2}} v^1 (v^1)^T D^{-\frac{1}{2}} p^0 = C\pi = \pi,$$

where the last two equalities follow as any eigenvector of W corresponding to $\omega_1 = 1$ has to be of the form $C\pi$, for some C, (cf. Lemma 1) and $W^t p^0$ has to be a valid probability distribution, which means that $C = 1$.

This concludes the proof of convergence for the vanilla version of the random walks. To see how the convergence for the lazy variant follows, note that by definition of the lazy walk matrix $\widehat{W}$ (6), we have that

$$\hat{\omega}_i = \frac{1}{2} + \frac{1}{2}\omega_i,$$

for any i, where $\hat{\omega}_1 \geq \ldots \geq \hat{\omega}_n$ are the eigenvalues of the matrix $\widehat{W}$. Also, it must be the case that the eigenvectors of $\widehat{W}$ and W coincide.

So, by Lemma 1, we have that $0 \leq \hat{\omega}_i \leq 1$ and, if G is connected, $\hat{\omega}_1 = 1$ is the only eigenvalue of value 1 (with π being it corresponding eigenvector). Thus, by repeating the reasoning we applied to W above, we can conclude that again

$$\hat{p}^t = \widehat{W}^t p^0 = D^{\frac{1}{2}}(\sum_{i=1}^{n} \hat{\omega}_i^t v^i (v^i)^T D^{-\frac{1}{2}} p^0) \overset{t\to\infty}{=} D^{\frac{1}{2}} v^1 (v^1)^T D^{-\frac{1}{2}} p^0 = \pi,$$

as desired. (Note that we do not need to assume here that G is non-bipartite.)

4 The Convergence Rate

Once we established the convergence result, it is natural to ask what is the rate at which this convergence happens. By analyzing Equation (10), one can establish the following lemma. (For convenience, we focus here on lazy random walk case.)

Lemma 2 *For any graph $G = (V, E)$ and any starting distribution $\hat{p}^0$, we have*

$$\|\hat{p}^t - \pi\| \leq \sqrt{\max_{v,u \in V} \frac{d(v)}{d(u)}} \hat{\omega}_2^t,$$

for any $t \geq 0$.

For a connected graph G, the bound on convergence rate of a (lazy) random walk given by the above lemma is mainly dependent on how much smaller than 1 is the value of $\hat{\omega}_2$. In particular, if we define the *spectral gap* $\hat{\lambda}(G)$ of the graph G to be

$$\hat{\lambda}(G) := 1 - \omega_2 = 2(1 - \hat{\omega}_2)$$

then we will have that, for any $\varepsilon > 0$, the number of steps t_ε needed for the lazy random walk distribution $\hat{p}^t$ to be within ε of the stationary distribution π is $O(\frac{1}{\hat{\lambda}(G)} \log \frac{\bar{d}(G)}{\varepsilon})$, where $\bar{d}(G) := \max_{v,u \in V} \frac{d(v)}{d(u)}$ is the *degree ratio* of G. (The time $t_{\frac{1}{2}}$ is sometimes called the *mixing time* of the graph G.)

As we will see in coming lectures, the spectral gap of the graph is a very important characteristic of the graph and captures a lot of its structural properties. So, the connection between $\hat{\lambda}(G)$ and the mixing time of random walks implies that observing the behavior of the random walks for moderate values of t, i.e., how fast it converges to stationary distribution, provides us with a lot of information about the graph. Indeed, a lot of powerful graph algorithms is based on this random walk paradigm.

5 Examples of Mixing Times

To get a feel for how different can the convergence rate (i.e., the mixing time) be in different graphs, we provide a heuristic analysis of random walk convergence on a few important examples.

1. Consider a *path graph* P_n on n vertices – cf. Figure 2(a), and assume that our starting vertex is in the middle. To understand the behavior of random walk in this graph, think about the random variable X^t that describes the offset (from the middle) of the vertex random walk is at at step t.

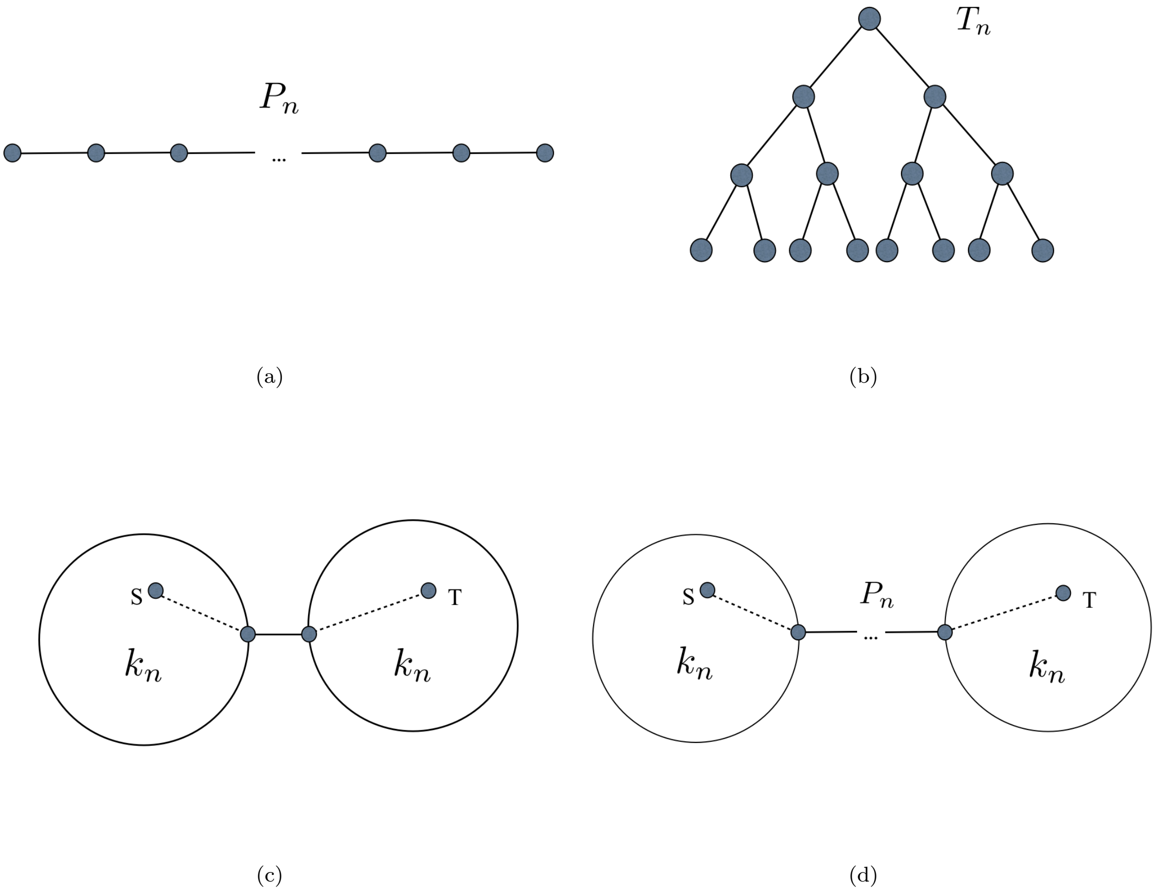
   Clearly, $X^t \in [-n/2, n/2]$ and, ignoring the boundary cases, we can view X^t as a sum of t independent random variables that are equally probable to be -1 and 1. Now, it is easy to see that the standard deviation of the variable X^t is $\sqrt{t}$. So, heuristically, to have a reasonable chance of reaching one of the path endpoints (which is a a necessary condition for good mixing), we need this standard deviation to be $\Omega(n)$. This implies that the mixing time should be $\Theta(n^2)$ and thus the spectral gap $\hat{\lambda}(P_n)$ has to be $O(1/n^2)$ (note that even a tight bound on mixing time provides only an upperbound on the spectral gap). As we will see later, even though our argument wasn't fully formal, the bounds we get are actually tight. In particular, $\hat{\lambda}(P_n)$ is $\Theta(1/n^2)$.

2. Consider now a full *binary tree graph* T_n on n vertices, as depicted at Figure 2(b). Let us consider a random walk in this graph that starts from one of the leaves. Clearly, in order to mix, we need to be able to reach the root of the tree with good probability. (In fact, once we reach the root, it will take only little time to mix.) The root is at distance of only $\log_2(n)$ away from the leaf, however, every time we "climb" towards the root we are twice as likely to go down rather than go further up. One can see that such a process requires $\Theta(2^d)$ steps, in expectation, to reach a point that is d levels from the leaf. Therefore, we expect the mixing time to be $\Theta(2^{\log_2 n}) = \Theta(n)$ and thus spectral gap to be $O(1/n)$. It turns out that the spectral gap $\hat{\lambda}(T_n)$ is indeed $\Theta(1/n)$.

3. Let us analyze now the *dumbbell graph* D_n shown in Figure 2(c). It consists of two complete graphs $K_{n/2}$ on $n/2$ vertices that are connected by only one bridge edge. (*Complete graph* K_n is a graph in which every vertex is connected to every other vertex.) Consider a random walk starting at one

of the vertices that are not endpoints of the bridge edge. For this walk to mix, we need to be able to, in particular, pass through the bridge edge to reach the other complete graph. However, in each step, we have only $\Theta(1/n)$ chance of moving to the endpoint of the bridge edge, and then we have only $\Theta(1/n)$ chance to move to its other endpoint and thus cross the bridge. So, crossing to other side in any two consecutive steps happens with probability of $\Theta(1/n^2)$ and thus the mixing time should be roughly $\Theta(n^2)$. This leads to upper bound on the spectral gap of this graph being roughly $O(1/n^2)$ and, again, this bound turns out to be asymptotically tight.

4. Finally, let us consider the *bolas graph* B_n depicted in Figure 2(d). The only difference from the dumbbell graph is that in this graph we have a path graph $P_{n/3}$ that connect the two complete graphs $K_{n/3}$ instead of just one bridge edge. Again, our heuristic for estimation of the mixing time is to consider the expected time to cross from one complete graph to the other. By similar reasoning, we get that the probability for us to enter the path is $\Theta(1/n^2)$, but now one can show that we have only probability of $\Theta(1/n)$ of reaching the other end of the path before exiting it through the endpoint we entered. This hints at the mixing time of $\Theta(n^3)$ and, once more, the resulting upperbound on the spectral gap of this graph turns out to be tight, i.e., $\hat{\lambda}(B_n) = \Theta(1/n^3)$. (As we will see later, $O(n^3)$ is the worse-case bound on the number of steps (in expectation) required to explore any graph of n vertices. So, bolas graph is an example of an graph that has worst bottlenecks from the point of view of random walks.)



(a)

(b)

(c)

(d)

Figure 2: Some example graphs. (a) Path graph P_n (b) Binary tree graph T_n (c) Dumbbell graph D_n (d) Bolas graph B_n