

Chapter 9: Geometric Bounds on Eigenvalues¹

9.1. Looking back, looking ahead.

Looking back: one need leads to another.

Our work so far has led us from permutation tests to eigenvalues:

- To test a null hypothesis, we need to be able to generate random data sets with all possibilities equally likely.
- To generate each one of the data sets, we need to carry out a random walk on a graph.
- To use the random walk, we have to know how many steps to take to get to a data set that can be regarded as random.
- To decide the number of steps, we need to know how quickly or slowly the random walk converges² to its limiting distribution.
- And (at last!) to find the rate of convergence, we need to know the eigenvalues of the transition matrix.

Looking ahead

This last need raises the question, “How are the eigenvalues of the transition matrix related to the structure of the graph?” From previous work, you should be able to give examples of graph walks that you would expect to converge rapidly, and others that you would expect to converge slowly. The goal of this chapter is to build from that intuitive base to relationships between measurable features of a graph’s geometric structure and the eigenvalues of its transition matrix.

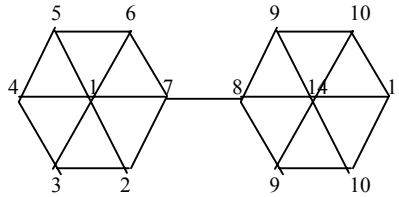
To think about this intuitively, regard the graph as a street map, and think about traffic flowing from vertex to vertex. The equilibrium distribution is given by $\pi_i = d_i / 2|E|$. Suppose you start $2|E|$ cars, with d_i cars at vertex i . How long will it take until the cars are thoroughly “mixed?” The “mixing time” will depend on how freely the cars can move from vertex to vertex. At one extreme, if the graph is complete, with an edge joining every pair of vertices, mixing will occur almost immediately: after just a few steps, it will be impossible to tell where a car has come from based on where it is.³ Other graphs, with fewer vertices than the complete graph, may have bottlenecks that prevent rapid flow of traffic.

¹ This chapter is based on Diaconis, Persi and Daniel Stroock (1991). “Geometric Bounds for Eigenvalues of Markov Chains,” *The Annals of Applied Probability*, vol. 1, pp. 36-61.

² Remember that for Markov chain Monte Carlo, we have two different kinds of convergence: $\hat{p} \rightarrow p$ and $\mathbf{p}^{(n)} \rightarrow \boldsymbol{\pi}$. The second kind is sometimes called “**mixing**,” as in “How long (how many shuffles) does it take to mix a deck of cards?” If this kind of convergence is geometric, with $\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\| \leq M\theta^n$, then the parameter θ is sometimes called the **mixing rate**.

³ You might think that for a complete graph, with all possible edges present, convergence would be immediate, and that after just one step, it would be impossible to tell which vertex a car had come from, but that is not the case. (Explain why.)

A fundamental intuition about mixing rates is that they are related to the presence or absence of such bottlenecks in the graph. A simple instance occurs in a dumbbell-shaped graph like the one in Display 9.1.



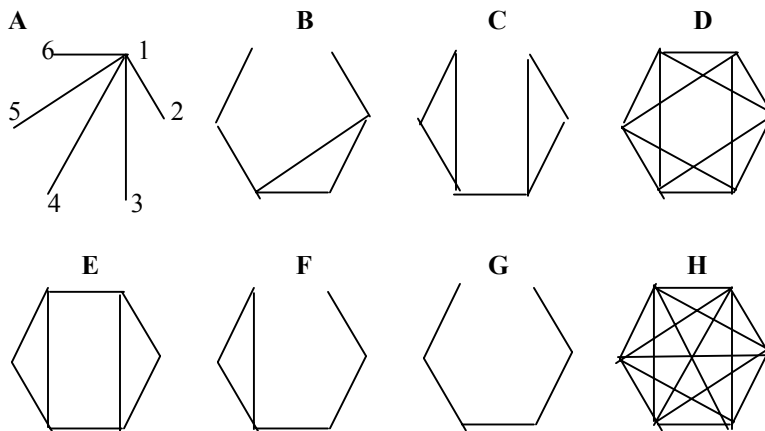
Display 9.1 A 14-point dumbbell with bottleneck edge $\{7,8\}$

Here the middle edge (from 7 to 8) is a bottleneck that makes it hard to go back and forth between the left set (1 – 7) and the right set (8-14). This bottleneck holds down the rate of mixing, a fact that shows up in the eigenvalues of the transition matrix: The second largest eigenvalue is roughly .95, and so the variation distance from the stationary distribution for this random walk is on the order of $(.95)^n$.

Activity: Exploring the geometry and algebra of graphs

A. Informal geometric analysis of graph walks.

Display 9.2 shows eight 6-point graphs:



Display 9.2 Eight 6-point graphs

1. Identify any of the graphs whose corresponding walks don't converge. (How can you tell?)
2. Which two of the remaining graph walks do you expect to show the fastest convergence? Of these two, which has the faster mixing rate?
3. Which two of the remaining graph walks do you expect to show the slowest convergence? Of these two, which one is slower?

4. Based on your answers in A1-A3, put the graphs whose walks converge in order, from fastest mixing to slowest mixing.
5. What features of the graphs are you using to make your judgments?

B. Algebraic analysis of graph walks

Display 9.3 shows eight sets of eigenvalues for the transition matrices of the random walks on the graphs of Display 9.1.

	Graph							
	1	2	3	4	5	6	7	8
Eigenvalues								
1	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
2	0.707107	0.000000	0.809017	0.795334	-0.200000	0.803956	0.000000	0.577350
3	0.193713	0.000000	0.309017	-0.166667	-0.200000	0.113037	0.000000	0.000000
4	-0.333333	0.000000	-0.309017	-0.500000	-0.200000	-0.500000	0.000000	-0.333333
5	-0.707107	-0.500000	-0.809017	-0.500000	-0.200000	-0.500000	0.000000	-0.577350
6	-0.860380	-0.500000	-1.000000	-0.628667	-0.200000	-0.916993	-1.000000	-0.666667

Display 9.3 Eigenvalues of the transition matrices for the graph walks of Display 9.2 (The order of the columns differs from the order of the graphs. See below.)

1. Identify the eigenvalues of the graphs whose walks don't converge.
2. Identify the eigenvalues of the two fastest-mixing graph walks. Of the two, which set of eigenvalues is for the faster-mixing graph?
3. Identify the eigenvalues of the two slowest-mixing graph walks. Of the two, which set if for the slower-mixing walk.
4. Based on your answers in A and B1-3, match each graph A – H with its set of eigenvalues 1- 8.

C. Distance from stationarity

Suppose the graph walk starts in state 1, so that the initial probability vector is $\mathbf{p}^{(0)} = (1,0,0,0,0,0)$. Then $\mathbf{p}^{(n)} = \mathbf{p}^{(0)}\mathbf{P}^n$ gives the probabilities n steps later. Display 9.3 shows, for each of the eight graph walks, the variation distance between the n -step probability distribution $\mathbf{p}^{(n)}$ and the stationary distribution $\boldsymbol{\pi}$. The order is the same as for the eigenvalues in Display 9.2.

1. For each graph, examine the last few rows of the table of variation distances $\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\|$. Decide which of the following is true:
 - a. Convergence of $\mathbf{p}^{(n)}$ to $\boldsymbol{\pi}$ is *linear*: for each of the graph walks, the distance $\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\|$ decreases by a constant *amount* Δ as you go from n to $n+1$, i.e., the *difference* $\Delta = \|\mathbf{p}^{(n)} - \boldsymbol{\pi}\| - \|\mathbf{p}^{(n+1)} - \boldsymbol{\pi}\|$ is roughly constant after the first several steps.

- b. Convergence of $\mathbf{p}^{(n)}$ to $\boldsymbol{\pi}$ is *geometric*: for each of the graph walks, the distance $\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\|$ decreases by a constant *factor* θ as you go from n to $n+1$, i.e., the *ratio* $\theta = \|\mathbf{p}^{(n)} - \boldsymbol{\pi}\| / \|\mathbf{p}^{(n+1)} - \boldsymbol{\pi}\|$ is roughly constant after the first several steps.
2. Compute the relevant one of Δ or θ for each of the graphs. Compare it with the set of eigenvalues for the graph: How is the mixing rate, as measured by the number in C1, related to the eigenvalues? What is the reason for the pattern that relates the two sets of numbers?

	Graph							
	1	2	3	4	5	6	7	8
Distances								
n = 1	0.750000	0.333333	0.800000	0.642857	0.166667	0.833333	0.500000	0.500000
n = 2	0.500000	0.166667	0.700000	0.333333	0.033333	0.750000	0.500000	0.236111
n = 3	0.361111	0.083333	0.600000	0.305556	0.006667	0.583333	0.500000	0.203704
n = 4	0.277778	0.041667	0.575000	0.217593	0.001333	0.583333	0.500000	0.100309
n = 5	0.188272	0.020833	0.537500	0.189043	0.000267	0.500000	0.500000	0.084362
n = 6	0.154321	0.010417	0.500000	0.140303	0.000053	0.458333	0.500000	0.044067
n = 7	0.113340	0.005208	0.500000	0.117906	0.000011	0.416667	0.500000	0.035437
n = 8	0.094822	0.002604	0.500000	0.089803	0.000002	0.364583	0.500000	0.019528
n = 9	0.073179	0.001302	0.500000	0.073920	0.000000	0.345486	0.500000	0.015064
n = 10	0.061614	0.000651	0.500000	0.057221	0.000000	0.293113	0.500000	0.008673
n = 11	0.048809	0.000326	0.500000	0.046497	0.000000	0.286748	0.500000	0.006466
n = 12	0.041320	0.000163	0.500000	0.036360	0.000000	0.239294	0.500000	0.003854
n = 13	0.034405	0.000081	0.500000	0.029308	0.000000	0.238534	0.500000	0.002798
n = 14	0.028442	0.000041	0.500000	0.023065	0.000000	0.203420	0.500000	0.001713
n = 15	0.024664	0.000020	0.500000	0.018498	0.000000	0.198869	0.500000	0.001218

Display 9.4 Total variation distance $\|\mathbf{p}^{(n)} - \boldsymbol{\pi}\|$ between the distribution n steps after starting in state 1 and the stationary distribution $\boldsymbol{\pi}$, for each of the graph walks whose eigenvalues are shown in Display 9.3.

The activity illustrates two important facts about random walks on graphs:

1. The convergence behavior, after the first several steps, is largely determined by the eigenvalue with the second largest absolute value.
2. The convergence behavior is also linked to the geometry of the graph in ways that let you tell the difference between faster-mixing and slower-mixing graphs walks without having to compute eigenvalues.

It is not hard to understand intuitively why the mixing rate depends on the geometry of the graph. (Think about the dumbbell, for example.) In addition, once you have become familiar enough with the spectral decomposition of the transition matrix for a graph walk, it is straightforward to go from there to the fact that the mixing rate is determined by the eigenvalues. In a sense then, even though some of the ideas involved may be hard at

first, neither of the two facts is very surprising. However, taken together, the two not-so-surprising statements have a logical consequence that is quite unexpected: the geometry of the graph and the eigenvalues of its transition matrix \mathbf{P} must be closely related.

The goal of this chapter is to find ways to link the algebraic and geometric analyses, using the geometry of a graph to say something about the eigenvalues of the corresponding walk, most especially about $\lambda^* = \max_{2 \leq i \leq k} |\lambda_i|$.⁴ In the last chapter you saw that the ordered eigenvalues satisfied $1 = \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k \geq -1$, which means that λ^* must be equal to the larger of $|\lambda_2|$ and $|\lambda_k|$. This fact lets us put the general goal of relating the geometry of a graph to its eigenvalues in a more focused form: We want to use the geometry of the graph to find bounds on $|\lambda_2|$ and $|\lambda_k|$, that is, inequalities of the form

$$\lambda_2 \leq 1 - c \text{ and } \lambda_k \geq -1 + c,$$

where the constant c depends on the structure of the graph. In what follows, you will see two closely related approaches to finding such constants. Both approaches take the informal idea of a bottleneck and develop it into a formal measure. The simpler of the two, which leads to a result called Cheeger’s Inequality, comes from looking carefully at what we might mean by a graph’s “greediest” set, the set of vertices with the lowest one-step probability of escaping from the set. The second approach, which leads to a result called Poincare’s Inequality, comes from looking carefully at what we might mean by a graph’s “busiest” edge.

7.2 Cheeger’s Inequality: Bounds Based on the Greediest Set

If you look back to the dumbbell graph in Display 9.1 and ask yourself, “Which set or sets is it hardest to escape from?” you can see right off that the two sets $S = \{1, 2, \dots, 7\}$ and its complement $S^c = \{8, 9, \dots, 14\}$ beg for attention. For each of these sets, there is only one edge, $\{7,8\}$, that leads out of the set. Moreover, S is a “big” set in the sense that there are lots of moves that start from vertices in S .⁵ (Check that the total number of such moves – those that start from vertices in S – equals the sum of the degrees of the vertices in S .) Define the “escape fraction” of a set S of vertices as the ratio of the “moves out of S ” to “total moves starting in S ”.

Def. Let u and v be vertices, and let e_{uv} be the edge from u to v . Then the **escape fraction** for a set S of vertices is

$$h(S) = \frac{\#\{e_{uv} : u \in S, v \in S^c\}}{\#\{e_{uv} : u \in S\}} = \frac{\#\{e_{uv} : u \in S, v \in S^c\}}{\sum_{u \in S} d_u}.$$

Drill.

1. Refer to the graph below. Find the escape fractions for each of the following sets.

⁴ Sometimes λ^* is written as “SLEM”, an acronym for “second largest eigenvalue modulus.”

⁵ The same is true of S^c .

(a) $S_1 = \{1\}$

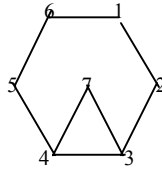
(b) $S_2 = \{1,2\}$

(c) $S_3 = \{1,2,3\}$

(d) $S_4 = \{2,5\}$

(e) $S_5 = \{3,4\}$

(f) $S_6 = \{3,4,7\}$

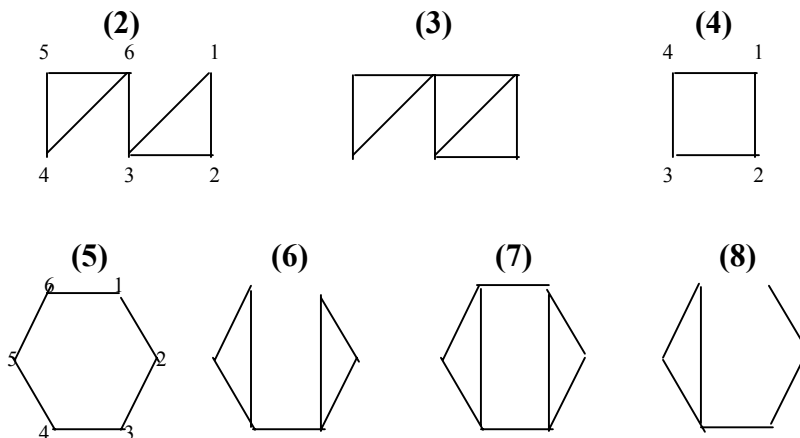


Now that we have defined the escape fraction, we are almost ready to define the greediest set. The basic idea we want to formalize is that the greediest set is the one with the smallest escape fraction. There is a small obstacle to get around first, though. To see this, ask yourself, “What is the escape fraction for the set V of all of a graph’s vertices?” Because there is no vertex to go to from V , by default its escape fraction is 0. We could declare this one set V to be an exception, off limits as a candidate to be greediest set, but you can convince yourself that we could easily encounter similar difficulties with sets that were almost as big as V . One way to avoid these difficulties would be to declare that S cannot contain more than half of the vertices. A better version of the same idea is to measure the importance of a vertex i by its stationary probability π_i , and require that the sum of these probabilities be less than $\frac{1}{2}$.

Def. The set S_* of vertices is **greediest** if (i) the sum of the stationary probabilities of the vertices of S_* , written $\pi(S_*)$, is at most $\frac{1}{2}$, and (ii) the escape fraction $h_* = h(S_*)$ is a minimum among all vertex sets S with $\pi(S) \leq \frac{1}{2}$.

Drill.

2 – 8. For each of the graphs shown in Display 9.5, find the greediest set S_* and its escape fraction h_* .



Display 9.5. Graphs for drill exercises 2 – 8: Find each graph’s greediest set.

Theorem. (Cheeger’s Inequality for simple graph walks) Let \mathbf{P} be the transition matrix for the random walk on a simple graph G . Let λ_2 be the second largest eigenvalue of \mathbf{P} , and let h_* be the escape fraction for the greediest set S_* . Then

$$1 - 2h_* \leq \lambda_2 \leq 1 - h_*^2 .$$

Proof: See Appendix.⁶

Discussion questions.

9. Explain why h_* must be in the interval between 0 and 1.
10. Explain why h_* is not a very meaningful quantity unless the graph is connected.
11. Draw/invent a graph with a very small value of h_* .
12. Can a graph have $h_* > 1/2$? If so, give an example. If not, explain why not.

Exercises.

13. Graph the upper Cheeger bound $1 - h_*^2$ versus h_* for $1 \leq h_* \leq 1$. Then, on the same axes, graph the lower bound $1 - 2h_*$ versus h_* for $1 \leq h_* \leq 1$.
14. (Continuation) For each of the graphs 2 – 8 in Display 9.5, locate the value of h_* for the graph on the horizontal axis, and then draw a vertical line segment above that point to show the range of possible values for λ_2 .
15. (Continuation) The values of λ_2 for these graphs are given below. For each graph, show the value of λ_2 on your graph by adding the point (h_*, λ_2) to the line segment for the graph. Do you notice any pattern relating the graphs to whether the value of λ_2 is closer to $1 - h_*^2$ or $1 - 2h_*$?

Graph	2	3	4	5	6	7	8
λ_2	.7953	.6584	0	.5	.7953	.5774	.8040

Questions for investigation:

16. By exploring graphs of various kinds, try to find what it is about the structure of a graph that makes the upper bound $1 - h_*^2$ be better (closer to λ_2) than the lower bound $1 - 2h_*$.

⁶ This appendix hasn’t been written yet. For a proof, see the paper by Diaconis and Stroock in footnote 1.

17. Can you find a graph for which $\lambda_2 = 1 - h_*^2$? For which $\lambda_2 = 1 - 2h_*$?

Extending Cheeger's Inequality

The version of Cheeger's Inequality you have been working with so far applies to walks on simple graphs. Often, as you have seen, we want to Metropolize a graph walk, which results in a Markov chain that cannot be represented as a random walk on a graph. What then?

Fortunately, there is a generalization of Cheeger's Inequality that applies; in fact this general version applies to any reversible Markov chain.⁷ All we need is (1) to define the graph associated with a reversible Markov chain, and (2) to find the "right" (i.e., workable) generalization of the escape fraction h_* . Both of these turn out to be straightforward. First, the graph G .

Def. Let \mathbf{P} be the transition matrix for a reversible Markov chain. Then the **graph associated with the chain** has as its vertices the states of the chain; the edges are the vertex pairs $\{i, j\}$ for which $p_{ij} > 0$. (Check that because the chain is reversible, it is impossible to have a pair i, j for which $p_{ij} > 0$ and $p_{ji} = 0$, so there is no ambiguity about which pairs correspond to edges.) If $p_{ii} > 0$, the graph has a self-loop from i to i .

To get a generalization of the escape fraction h_* , we keep the basic idea from before, but instead of counting edges to get probabilities, we add values of $\pi_i p_{ij}$. These values define a set of probabilities on the vertex pairs.

Def. Let \mathbf{P} be a reversible Markov chain with stationary distribution π . Then the **reversing probability measure** (reversing measure, for short) for \mathbf{P} is given by

$$\mathbf{Q} = \{q_{ij}\}, \text{ where } q_{ij} = \pi_i p_{ij} \text{ (} = q_{ji} = \pi_j p_{ji} \text{)}.$$

Drill.

18. Consider the random walk on the 3-point linear graph 1—2—3. Find the reversing measure \mathbf{Q} , and verify that \mathbf{Q} is symmetric ($q_{ij} = q_{ji}$ for all pairs) and that \mathbf{Q} is a probability measure on the pairs i, j (the q_{ij} add to 1 over all pairs).

19. Metropolize the random walk in (18) so that the stationary distribution is uniform. Find the reversing measure \mathbf{Q} for this chain, and verify that \mathbf{Q} is once again symmetric and a probability measure.

⁷ Remember that a chain is reversible if for every pair of states i, j , the steady state flow from i to j , $\pi_i p_{ij}$, equals the steady state flow from j to i , $\pi_j p_{ji}$.

Prop. \mathbf{Q} is a probability measure on the ordered pairs (i,j) of states of \mathbf{P} , that is,

$$0 \leq q_{ij} \leq 1 \text{ for all } (i,j) \text{ and } \sum_{i=1}^k \sum_{j=1}^k q_{ij} = 1.$$

Proof: (Drill problem 26.)

The fact that \mathbf{Q} is a probability measure means that we can use \mathbf{Q} to assign probabilities to sets of pairs (i,j) . In particular, if A and B are sets of states of \mathbf{P} , let $Q(A \times B)$ be the probability of the Cartesian product $A \times B$, that is, of the set of pairs (i,j) with $i \in A$ and $j \in B$:

$$Q(A \times B) = \sum_{i \in A} \sum_{j \in B} q_{ij}.$$

If $V = \{1, 2, \dots, k\}$ is the entire set of states and A is any other set of states, define $Q(A)$ to mean $Q(A \times V)$

We can now define the escape fraction for sets of states of reversible Markov chains. Whereas before, for random walks on simple graphs, we used the ratio “number of edges from S to S^c over the total number of edges originating in S ,” now we use the ratio “probability of the set of edges from S to S^c over the probability of the set of edges originating in S ,” where “probability” refers to the reversing measure \mathbf{Q} :

Def. Let \mathbf{P} be a reversible Markov chain with stationary distribution π and reversing measure $\mathbf{Q} = \{q_{ij}\}$. Let S be a set of states of \mathbf{P} . Then the **escape fraction** for S is given

by $h(S) = \frac{Q(S \times S^c)}{Q(S)}$. As before, the **greediest** set is the set S_* with the smallest escape

fraction among all S with $\pi(S) \leq 1/2$, and $h_* = h(S_*)$.

Drill.

20. Consider the random walk on the 3-point linear graph 1—2—3.
- Find the greediest set S_* and its escape fraction $h_* = h(S_*)$ the old way, that is using the definition of escape fraction h for random walks on simple graphs.
 - Now find the greediest set the new way, using the definition of escape fraction for reversible Markov chains, and the reversing measure \mathbf{Q} from Exercise 18.
 - Finally, find the greediest set for the Metropolized chain, using the reversing measure from Exercise 19.

Using our newly generalized definitions of escape fraction and greediest set, we can now state the more general version of Cheeger’s Inequality.

Theorem. (Cheeger’s Inequality for Reversible Markov Chains.) Let \mathbf{P} be the transition matrix for a reversible Markov chain. Let λ_2 be the second largest eigenvalue of \mathbf{P} , and let h_* be the escape fraction for the greediest set S_* . Then

$$1 - 2h_* \leq \lambda_2 \leq 1 - h_*^2.$$

Drill

21. Find h_* for a 3-state Markov chain with $p_{11} = p_{22} = p_{33} = .5$ and $p_{ij} = .25$ if $i \neq j$.
22. Start with a random walk on the 4-point linear graph $1-2-3-4$. Metropolize to give a uniform stationary distribution. For the Metropolized walk, find the greediest set S_* and the value of h_* .
23. Repeat the steps of 22, this time with the 5-point linear graph $1-2-3-4-5$.
24. Repeat the steps of 22, this time for a k -point linear graph.
25. Start with a random walk on a triangle-with-tail, as shown below. Metropolize to give a uniform stationary distribution. For the Metropolized walk, find the greediest set S_* and the value of h_* .

**Exercises**

26. Prove that the reversing measure \mathbf{Q} is a probability measure, i.e.,

$$0 \leq q_{ij} \leq 1 \text{ for all } (i,j) \text{ and } \sum_{i=1}^k \sum_{j=1}^k q_{ij} = 1.$$

27. Show that if \mathbf{P} is the transition matrix for a simple graph walk with stationary distribution π and reversing measure $\mathbf{Q} = \{q_{ij}\}$, then for any set of vertices S ,
- $\pi(S) = Q(S)$, and
 - the two definitions of $h(S)$ give the same value.

Investigations:

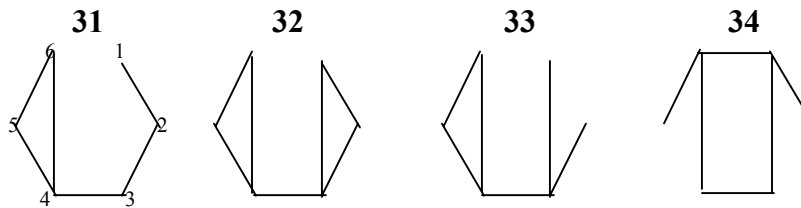
28. Is greediness invariant with respect to Metropolizing? Let S_* be the greediest set for a random walk on a simple connected graph, and let ${}_M S_*$ be the greediest set for the Markov chain that you get by Metropolizing the walk to give uniform stationary distribution. Is it always the case that $S_* = {}_M S_*$?
29. What is the effect of Metropolizing on the escape fraction for the greediest set? Let h_* be the escape fraction for the greediest set of a walk on a simple graph, and let ${}_M h_*$ be the escape fraction for the greediest set of the random walk Metropolized to give uniform limiting probabilities. Which of the following is true:
- $h_* \geq {}_M h_*$ for all connected graphs
 - $h_* \leq {}_M h_*$ for all connected graphs
 - $h_* < {}_M h_*$ for some graphs, and $h_* > {}_M h_*$ for others
30. Let h_* and ${}_M h_*$ be as in the previous problem. For which graphs, if any, will $h_* = {}_M h_*$?

9.3 Poincare's Inequality: Bounds Based on the Busiest Edge

Cheeger's Inequality relates mixing to getting trapped in a greedy set, and bounds the mixing rate using the traffic flow out of that set – the more restricted the flow, the slower the mixing. Poincare's Inequality is similar in that it tries to quantify the idea of a bottleneck, but its focus is on the busiest edge of the graph.

Discussion questions

31 - 34. Separately for each of the graphs shown in Display 9.6, decide which edges you think should be regarded as busiest. Then put the edges of the graphs into groups according to how busy you judge them to be. What principles or informal rules are you using as the basis for your groupings?



Display 9.6 Four 6-point graphs: Find the busiest edges.

The next few pages develop a measure of the traffic flow along an edge. The basic idea is simple: list the paths that use the edge, and find their average length. However, several adjustments to the basic idea are needed to make it workable. Rather than present these all at once, hare-from-a-hat, we start with the basic idea, and build in the adjustments one at a time via examples and exercises that show why the adjustments are needed in order to make the basic idea workable.

Measure # 1: Average length of paths containing the edge.

For fast mixing, it must be possible to go quickly from any vertex i to any other vertex j . One natural measure of how quickly you can get from i to j is the length of the shortest path from i to j . This suggests a measure like the following:

0. For each ordered pair (i, j) list a path γ_{ij} from i to j . Often γ_{ij} is the shortest path, but the measure works for any choice of paths. Let Γ stand for the collection of all these paths, one for each ordered pair of vertices.
1. For any edge e , measure the traffic flow across e by the average length of the paths of Γ that contain e .

The following drill exercises will walk you through the computations, and at the same time will show you a shortcoming of the first measure of busy-ness.

Drill.

35. Consider the 4-point linear graph 1—2—3—4
- Which of the three edges do you consider busiest? Explain your reasoning.
 - List all the vertex pairs with $i < j$.
 - Next to each pair, list the shortest path from i to j , and its length.
 - List all edges $\{i,j\}$ in a new list.
 - Next to each edge, list all paths from your list in (c) that contain the edge.
 - Use that list to find, for each edge, the average length of the paths that contain it.
 - Explain why the averages wouldn't change if you listed *all* possible vertex pairs instead of just those with $i < j$.
 - Does average length, as defined by this measure, give values for the three edges that fit with your intuition? Explain.

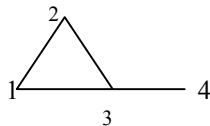
Measure #1'. Average travel time.

It will turn out that the connection between our measure of busy-ness and the eigenvalues of \mathbf{P} will be cleaner if we replace average length of the paths by the average value of $2|E| \times (\text{length})$, where $|E|$ is the number of edges in the graph. Step 0 is the same as before.

- For any edge e , measure the traffic flow across e by $2|E|$ times the average length of the paths of Γ that contain e .

Where does the extra factor $2|E|$ come from? In one sense, the answer can only come from understanding the proof that establishes the connection to eigenvalues. That proof comes later. (See Appendix.⁸) Meanwhile, here is a partial explanation that is faithful to the logic of the relationship. I'll give a summary sentence first, then illustrate what it means using an example. In brief, $2|E| \cdot |\gamma_{ij}|$ is the average time it would take you to travel the path from i to j if you were to draw edges at random from a box of labeled tickets.

Example 9.1 For the 4-point graph containing a 3-cycle, as shown below, consider the path $\gamma_{14} = 134$.



The length of the path is $|\gamma_{14}| = 2$. For this graph, the number of edges is $|E| = 4$, so $2|E| = 8$, and $2|E| \cdot |\gamma_{14}| = 8 \cdot 2 = 16$. According to the summary sentence, if you “draw edges at random,” on average it will take you 16 draws to travel the path γ_{14} . Here's why: Because there are $|E| = 4$ edges (*undirected* edges), there are $2|E| = 8$ directed edges. If I draw at random from these 8, I have a $1/8$ chance of getting any particular edge. I want to go from 1 to 4 by way of 3. To do this takes two steps, first from 1 to 3, and then from 3 to 4. Now imagine how it goes if I draw directed edges at random. At any given step, there is only one edge I can use; if I draw any other, I have to throw it away and stay

⁸ This appendix hasn't been written yet. For a proof, see the paper by Diaconis and Stroock in footnote 1.

where I am for another step. How long will it take me to get from 1 to 4 according to these rules?

First, I have to draw $(1 \rightarrow 3)$. Each time I draw, I have a chance of $1/8 = 1/2|E|$ of getting the edge I need, so on average it will take $8 = 2|E|$ draws to get $(1 \rightarrow 3)$. By the same argument, on average it will take $2|E| = 8$ draws to go from 3 to 4. In all, then, it takes $8 + 8 = 16 = 2|E| \cdot |\gamma_{14}|$ draws to go from 1 to 4, and in general, it takes $2|E| \cdot |\gamma_{ij}|$ steps, on average, to travel the path γ_{ij} .

Measure #2. Number of paths that use an edge

The first measure, and its rescaled version l' , both ignore important information, namely: how many paths use the edge in question? In Exercise 35, the left-hand edge 12 is used by three paths, 12, 123, and 1234; the middle edge is used by four paths, 123, 1234, 23, and 234. Intuitively, then, it is natural to regard edge 23 as busier than edge 12, even though by Measure #1 (and l') they both have the same average path length (and travel time). Here is the new alternative. Step 0, constructing the set Γ of paths is the same as before.

1. For any edge e , measure the busy-ness of e by the number of the paths of Γ that contain e .

Because all our measures of edge traffic depend on creating a list Γ of paths joining all vertex pairs, it is worth doing a quick example to illustrate an efficient way to generate the list.

Example 9.2: Listing paths.

Refer back to the graph for Exercise 26 in Display 9.6. The graph has 6 vertices, so in all there are $6 \times 6 = 36$ vertex pairs. We can ignore the pairs (i,i) , which leaves 30 pairs (i,j) with $i \neq j$. Moreover, because any path from i to j will also take us from j to i , we need only consider pairs (i,j) with $i < j$, which brings us down to 15 pairs.⁹ We can list these systematically in a table with one row for each first vertex i and one column for each second vertex j :

Pair	Path	Pair	Path	Pair	Path	Pair	Path	Pair	Path
12	12								
13	123	23	23						
14	1234	24	234	34	34				
15	12345	25	2345	35	345	45	45		
16	12346	26	2346	36	346	46	45	56	56

Display 9.7 A list of paths for the graph A of Display 9.6

⁹ It is possible, and sometimes actually desirable, to use a path from j to i that is different from the path from i to j . Here, however, for simplicity the example assumes that each path is used twice, once in each direction.

Notice that in this table, it isn't really necessary to list the vertex pairs separately, because you can read them from the starting and ending vertices of the path. This lets you compress the table:

From	To				
	2	3	4	5	6
1	12				
2	123	23			
3	1234	234	34		
4	12345	2345	345	45	
5	12346	2346	346	45	56

Display 9.8 A compact version of the list of paths

With this listing, you can now practice computing Measure #2:

Discussion question .

36. Refer back to the graph for Exercise 31 in Display 9.6
 - a. Review your decisions about which edges you thought were busiest, next busiest, ... least busy.
 - b. Then use Display 9.8 to find the number of paths that use each edge.
 - c. Compare (a) and (c): How well does this measure fit with your intuitive judgments?

Measure #3: Average travel time, with 0s for unused paths.

The examples so far suggest that we might get a more reasonable measure than either of the first two by combining them. One simple way to do this would be to add up all the travel times of the paths that use the edge we are interested in. This approach gives an edge two ways to get a high score, by having long paths that use the edge, or by having a large number of paths that use it. We can make the measure a little easier to interpret if we divide by the total number of vertex pairs in the graph, to get an average travel time that assigns time 0 to paths that don't use the edge. Assuming, as always, that we have listed a set of paths Γ this gives:

1. For any edge e , assign an "edge use indicator" $\delta_e(i, j)$ to every vertex pair (i, j) :

$$\delta_e(i, j) = \begin{cases} 1 & \text{if the path } \gamma_{ij} \text{ uses the edge } e \\ 0 & \text{otherwise} \end{cases}$$

2. Compute the average, over all vertex pairs, of $2|E| \cdot |\gamma_{ij}| \cdot \delta_e(i, j)$.

Example 9.3. Consider once again the 4-point linear graph 1—2—3—4

The table in Display 9.9 shows a systematic way to compute Measure #3: List the vertex pairs in the left column. For each pair, list the shortest path (column 2) and its length (column 3). Then for each edge (columns 4 – 6) record a 1 if the path for that row uses the edge in that column. Finally, for each edge, sum the values of (edge use)×(length) to

get the sum of the lengths of the paths that use the edge, multiply by $2|E| = 6$ to get the sum of the travel times, and divide by the number of vertex pairs (here also 6) to get the average.

Vertex pair	Path	Length	Edge use indicators		
			e = 12	e = 23	e = 34
1,2	12	1	1	0	0
1,3	123	2	1	1	0
1,4	1234	3	1	1	1
2,3	23	1	0	1	0
2,4	234	2	0	1	1
3,4	34	1	0	0	1
Sum			6	8	6

Sum = sum of (edge use)x(length)

Display 9.9 Computing average travel times for the 4-point linear graph.

For this particular example, because the number of directed edges equals the number of vertex pairs with $i < j$, the average travels times are the same as the sums: 6 for the left and right edges, and 8 for the middle edge.

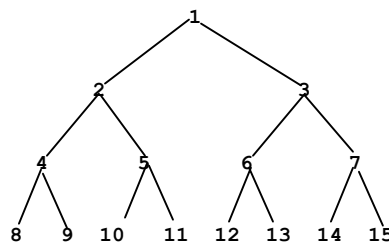
Drill.

37. Compute values of Measure # 3 for edge 34 of the graphs in Exercises 31 and 32. (Do you find that the slower-mixing graph has the larger value?)

38. Compute the value of Measure # 3 for all four edges of the graph of Example 9.2 (the 4-point graph with a 3-cycle).

There is only one more adjustment still to make. Our newest measure of busy-ness takes an all-or-nothing approach to the paths: If a path uses an edge, it gets counted; if it doesn't use the edge, it gets ignored. You can think of the resulting measure as based on a weighted sum with weight $\delta_e(i, j) = 1$ if the path uses the edge, and weight $\delta_e(i, j) = 0$ if not. Regarding Measure #3 as a weighted average in this way opens up the possibility of using other weights instead of the 0,1 all-or-nothing approach. To see why this flexibility is valuable, notice that Measure #3 gives equal weight to all paths that use a given edge, and consider the possibility that some paths may in fact matter more than others. Here's an example:

Example 9.4. Consider a random walk on the tree shown in Display 9.10.



Display 9.10 A complete tree of depth 3

Paths 849 and 842 both have length 2, but 849 is not nearly as important as 842: Whereas 849 is needed to connect vertex 8 to just one other vertex, namely 9, path 842 is needed to connect vertex 8 to 12 other vertices: 1, 2, 3, 5, 6, 7, 10, 11, ..., 15.

Measure # 4. Weighted average travel time

Our final adjustment is this. Measure the busy-ness of an edge using a weighted average of the travel times of the paths that use the edge, with weights determined by the stationary distribution: the weight assigned to vertex pair (i, j) , and to the path γ_{ij} joining them, is $\pi_i\pi_j$. The resulting measure is denoted $\kappa(e)$.

1. For each edge, assign the "edge use indicator" $\delta_e(i, j)$ to every vertex pair (i, j) .
2. To each ordered pair (i, j) assign the weight $\pi_i\pi_j$.
3. Compute $\kappa(e)$ as the weighted average, over all vertex pairs, of $2|E|\cdot|\gamma_{ij}|\cdot\delta_e(i, j)$:

$$\kappa(e) = 2|E|\sum_{i=1}^k\sum_{j=1}^k|\gamma_{ij}|\delta_e(i, j)\pi_i\pi_j.$$

Example 9.5. Find $\kappa(e)$ for the three edges of the 4-point linear graph 1—2—3—4. Notice that the calculations here are very much like those in Example 9.3 and Display 9.9. In fact, there are only three changes. First, we insert a column of weights $\pi_i\pi_j$ for the paths γ_{ij} . Second, the sums are sums of triple products (path length)(path weight)(edge use). Finally, if the vertex list contains only pairs with $i < j$, we double the sum to get $\kappa(e)$, because each row should get counted twice, once for the direction $i \rightarrow j$, and once for $j \rightarrow i$.

Vertex pair	Path	Length	Weight	Edge use indicators		
				e = 12	e = 23	e = 34
1,2	12	1	(1/6)(2/6)=2/36	1	0	0
1,3	123	2	(1/6)(2/6)=2/36	1	1	0
1,4	1234	3	(1/6)(1/6)=1/36	1	1	1
2,3	23	1	(2/6)(2/6)=4/36	0	1	0
2,4	234	2	(2/6)(1/6)=2/36	0	1	1
3,4	34	1	(2/6)(1/6)=2/36	0	0	1
K(e)=2xSum				18/36	30/36	18/36
Sum = sum of (length)x(weight)x(edge use)						

Display 9.11 Computing $\kappa(e)$ = weighted average travel time for the 4-point linear graph

Def. The **busiest edge** of a graph is the edge e^* for which $\kappa(e)$ is maximum. Denote the value of $\kappa(e^*)$ by κ^* .

Drill.

39. Find $\kappa(e)$ for each of the edges of the 4-point graph with 3-cycle. Which edge is busiest?

40. The busiest edge in each of the graphs of Exercises 31 – 33 is the edge 34 at the bottom in the middle. Find $\kappa(e)$ for edge 34 of each of those three graphs.

We now have a measure of busy-ness that leads to a simple bound on the second largest eigenvalue of a graph walk.

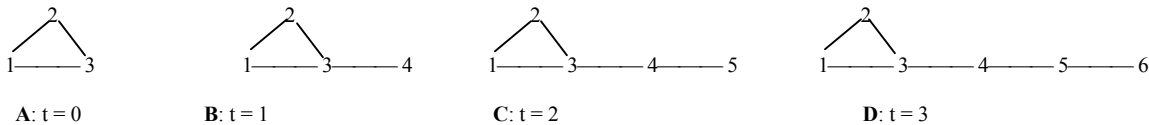
Theorem. (Poincare’s Inequality for simple graphs.) Let λ_2 be the second largest eigenvalue of the transition matrix \mathbf{P} for the random walk on a simple connected graph. Then

$$\lambda_2 \leq 1 - \frac{1}{\kappa^*}.$$

Proof: See Appendix.¹⁰

Drill.

41 – 44. Consider graphs A – E, each of the form “3-cyle with tail of length t .”



Display 9.12 Graphs for Exercises 41 - 44

41. a. Compute $\kappa(e)$ for each edge of graph A, following the format in Display 9.11.
 b. Find the Poincare bound for λ_2 , and compare it with the actual value of λ_2 , which is -0.5 .

42. The busiest edge of graph C is 34. Compute κ for this edge, following the format in Display 9.11, and use it to compute the Poincare bound for λ_2 . (The actual value of λ_2 is 0.654.)

43. For graph D, the busiest edge is also 34. Compute κ for that edge, but rather than follow the format in Display 9.11, use the following shortcut: (1) List all the paths that use the edge. (2) Next to each path, write the weight $\pi_i \pi_j$. (3) Compute the sum of (length)x(weight) for these paths. (4) Multiply your total by $2|E|$ to get the average travel time.

44. Consider a 3-cyle with tail of length t . Find a formula for κ^* as a function of t .

45. Show that the weights $\pi_i \pi_j$, summed over *all* pairs (i, j) add to 1. (This shows that $\kappa(e)$ is a true weighted average, or, in the language of probability, an expected value.)

¹⁰ This appendix hasn’t been written yet. For a proof, see the paper by Diaconis and Stroock in footnote 1.

Extending Poincare’s Inequality

Here for Poincare, just as for Cheeger earlier, we have an initial result that applies to random walks on simple graphs, but we need a result that applies more generally. Once again, it is possible to extend the initial result to a version that applies to any reversible Markov chain. All that’s required is to replace $2|E| \cdot |\gamma_{ij}|$, which is the average time to travel a path γ_{ij} if you draw paths at random with *uniform* probabilities, with a different quantity, $|\gamma_{ij}|_Q$, which is the average time to travel the path γ_{ij} if you draw at random with probabilities given by the *reversing measure* $\mathbf{Q} = \{q_{ij}\}$ defined by $q_{ij} = \pi_i p_{ij}$.

Remind yourself why $2|E| \cdot |\gamma_{ij}|$ gives the average time (number of draws) to travel the path γ_{ij} : There are $2|E|$ directed edges, so the chance of getting the edge you need at any given time is $p = 1/(2|E|)$. On average then, it takes $1/p = 2|E|$ draws to get each edge you need, and you have $|\gamma_{ij}|$ edges to travel. In other words, you add up $1/p + 1/p + \dots + 1/p$, with one term in the sum for each edge in the path. The only difference for a reversible chain is that you draw edges according to the reversing measure \mathbf{Q} , so the chance of getting edge $u \rightarrow v$ is q_{uv} , and the average number of draws it takes to get that edge is $1/q_{uv}$. To get $|\gamma_{ij}|_Q$, you simply add up values of $(1/q_{uv})$ for all the edges uv in the path γ_{ij} .¹¹

Theorem. (Poincare’s Inequality for Reversible Markov chains). Let \mathbf{P} be a reversible Markov chain with stationary distribution π , reversing measure \mathbf{Q} , and associated graph G . Then the second largest eigenvalue λ_2 of \mathbf{P} satisfies

$$\lambda_2 \leq 1 - \frac{1}{\kappa(e^*)}$$

where, for any edge e of G ,

$$\kappa(e) = \sum_{i=1}^k \sum_{j=1}^k |\gamma_{ij}|_Q \delta_e(i, j) \pi_i \pi_j.$$

and e^* is an edge for which $\kappa(e)$ is maximum.

¹¹ Question: Why is it reasonable in this context to draw edges at random with probabilities given by the reversing measure? Answer: Reversing measure tells the steady state probabilities that go with the following two step scheme: First, pick a starting vertex at random with probabilities given by the stationary distribution π . this means you start at i with probability π_i . (At equilibrium, the fraction of time spent at vertex i is π_i , so this is a reasonable set of probabilities.) Now pick a destination vertex from the i^{th} row of \mathbf{P} : this means you go to j with probability p_{ij} . Putting the two steps together, the chance is $q_{ij} = \pi_i p_{ij}$ that your randomly chosen edge will be $i \rightarrow j$.

Prop. If \mathbf{P} is the random walk on a simple connected graph G , then for any edge e of G , the value of $\kappa(e)$ as defined above is the same as the value defined previously.

Proof: Exercise 49.

Example 9.6. Start with the random walk on the 4-point graph with a 3-cycle, from Example 9.1. Metropolize to get a uniform stationary distribution. Find the busiest edge, the value of κ^* , and the Poincare bound on λ_2 .

Solution: The original and Metropolized transition matrices are shown below. Because the stationary distribution π is uniform, the reversing measure \mathbf{Q} is just a scalar multiple of \mathbf{P} , with the scalar chosen to make the entries of \mathbf{Q} add to 1.

$$\mathbf{P} = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 \\ 1/3 & 0 & 1/3 & 1/3 \\ 1/3 & 0 & 1/3 & 1/3 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \tilde{\mathbf{P}} = \begin{pmatrix} 1/4 & 1/2 & 1/3 & 0 \\ 1/6 & 1/3 & 0 & 1/3 \\ 1/3 & 0 & 1/3 & 1/3 \\ 0 & 1/3 & 2/3 & 0 \end{pmatrix} \quad \mathbf{Q} = \begin{pmatrix} 1/3 & 2 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 24 & 2 & 0 & 2 \\ 0 & 2 & 4 & 0 \end{pmatrix}$$

Display 9.13 shows the calculations needed to find the values of $\kappa(e)$.

Vertex pair	Path	Length	Weight	Edge use indicators				
				e = 12	e = 13	e = 24	e = 34	
1,2	12	24/3	= 8	(1/4)(1/4)=1/16	1	0	0	0
1,3	13	24/2	= 12	(1/4)(1/4)=1/16	0	1	0	0
1,4	134	24/2 + 24/2	= 24	(1/4)(1/4)=1/16	0	1	0	1
2,3	23	24/2	= 12	(1/4)(1/4)=1/16	0	0	1	0
2,4	234	24/2 + 24/2	= 24	(1/4)(1/4)=1/16	0	0	1	1
3,4	34	24/2	= 12	(1/4)(1/4)=1/16	0	0	0	1
K(e)=2xSum					16/16	72/16	72/16	120/16
Sum = sum of (length)x(weight)x(edge use)								

Display 9.13 Calculating $\kappa(e)$ for the Metropolized 4-point graph with 3-cycle

The busiest edge is 34, just as for the un-Metropolized walk, and the value of κ^* is 15/2.

Poincare's Inequality guarantees that $\lambda_2 \leq 1 - \frac{2}{15} = \frac{13}{15} \approx 0.867$.

Drill.

46. Consider the Markov chain with transition matrix $\mathbf{P} = \begin{pmatrix} 0 & 1/2 & 1/2 \\ 1/3 & 1/3 & 1/3 \\ 1/2 & 0 & 0 \end{pmatrix}$.

- Check that the limiting distribution is $\pi = \frac{2}{3} \mathbf{1}$.
- Check that the reversing measure \mathbf{Q} assigns probability 0 to pairs (1,1) and (3,3), and assigns probability 1/7 to all other pairs.
- Following Display 9.13 as a model, find $\kappa(e)$ for all edges $e = \{i, j\}$ with $i \neq j$.

- d. Find the busiest edge e^* , the value of $\kappa^* = \kappa(e^*)$, and the Poincare bound.
47. Find the busiest edge e^* , the value of $\kappa^* = \kappa(e^*)$, and the Poincare bound for the 4-point linear graph, Metropolized to have uniform stationary distribution.
48. Consider the 5-point walk on the 3-cycle with tail of length 2 (Graph C in Display 9.12), Metropolized to uniform stationary distribution. Find κ^* and the Poincare bound.
49. Prove that if \mathbf{P} is the random walk on a simple connected graph G , then for any edge e of G , the old and new definitions of $\kappa(e)$ give the same value.
50. Remind yourself of the random walk on the co-occurrence matrices with row sums 2,1,2 and column sums 2,1,2. Metropolize to get a uniform stationary distribution, then find κ^* and the Poincare bound.

9.4 Poincare Bounds on the Smallest Eigenvalue

Step back from all the details of the previous two sections – admittedly, there have been many! -- to review the agenda for this chapter. Our main goal is to use the geometry of a graph to say something about the mixing rate for a random walk on the graph. So far, we have two different ways to put bounds on the second largest eigenvalue of \mathbf{P} . This is good progress, but it still leaves us short of our goal.

As you saw in the Activity of Section 9.1, the mixing rate depends, ultimately, on the eigenvalue (apart from $\lambda_1 = 1$) with the largest absolute value. When the eigenvalues are ordered, with $1 = \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k \geq -1$, this eigenvalue is either λ_2 or λ_k . Let θ be the larger of the two absolute values: $\theta = \max\{|\lambda_2|, |\lambda_k|\}$. Then, as you saw in Part C of the Activity, for large enough n , the ratio of successive distances $\|\mathbf{p}^{(n+1)} - \boldsymbol{\pi}\| / \|\mathbf{p}^{(n)} - \boldsymbol{\pi}\|$ is roughly equal to θ . In other words, convergence (mixing) is geometric, with rate $\theta = \max\{|\lambda_2|, |\lambda_k|\}$.¹²

If you regard bounds on θ as the goal of the chapter, you can see that we have more to do: so far, we have bounds on $|\lambda_2|$, but nothing for $|\lambda_k|$. Fortunately, there is a straightforward modification of the Poincare Inequality that gives a lower bound for λ_k . This new bound, together with the ones from Sections 9.2 and 9.3, gives bounds on θ .

Here are the required modifications:

Replace the list of vertex pairs with a list of vertices.

Replace the paths γ_{ij} from i to j with paths σ_i from i back to i , requiring that each such path be of odd length, with no repeated edges.

For any edge e , define $\iota(e) = \sum_{i=1}^k |\sigma_i|_Q \pi_i \delta_e(i)$, where the edge use indicator $\delta_e(i) = 1$ if the path σ_i uses edge e , and $\delta_e(i) = 0$ otherwise.

Theorem (Poincare's Inequality for the Smallest Eigenvalue) Let \mathbf{P} be the transition matrix for a reversible Markov chain with stationary distribution $\boldsymbol{\pi}$ and reversing measure \mathbf{Q} . With ι defined as above, let $\iota^* = \iota(e^*)$, where e^* is the edge for which $\iota(e)$ is maximum. Then the smallest eigenvalue λ_k of \mathbf{P} satisfies

$$\lambda_k \geq -1 + \frac{2}{\iota^*}.$$

Example 9.7. Find the Poincare bound on the smallest eigenvalue λ_4 for the random walk on the 3-cycle with tail of length 1.

¹² The distance from θ to 1 is called the **spectral gap**.

Solution:

Vertex	Path	Length	Weight	Edge use indicators			
				12	13	23	34
1	1231	3	2/8	1	1	1	0
2	2312	3	2/8	1	1	1	0
3	3123	3	3/8	1	1	1	0
4	431234	5	1/8	1	1	1	1
Sum				26/8	26/8	26/8	5/8

$$\text{Sum} = \text{sum of (length)} \times (\text{weight}) \times (\text{edge use})$$

Display 9.14 Computing the Poincare bound for the smallest eigenvalue

To get the value of $t(e)$ from the sum in the table, multiply by $2|E|$ to convert the average path length to the average travel time. The edges 12, 13 and 23 are equally busy according to this measure, with $i^* = 26$. This gives a Poincare bound

$$\lambda_4 \geq -1 + \frac{2}{i^*} = -1 + \frac{2}{26} = \frac{12}{13} \approx -0.923.$$

Example 9.8. Find the Poincare bound on the smallest eigenvalue λ_4 for the random walk on the 4-point linear graph, Metropolized to have uniform stationary distribution.

Solution. The transition matrix, Metropolized transition matrix, and reversing measure for the Metropolized chain are given below:

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \tilde{\mathbf{P}} = \begin{bmatrix} 1/2 & 1/2 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 1/2 & 1/2 \end{bmatrix} \quad \mathbf{Q} = \left(\frac{1}{8}\right) \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

The calculations needed to compute average travel times are shown in Display 9.15. Notice that Metropolizing the walk has added self-loops at the end vertices, so there are two additional edges to consider, 11 and 44.

Vertex	Path	Travel time	Weight	Edge Use Indicators				
				12	23	34	11	44
1	11	8	1/4	0	0	0	1	0
2	2112	8 + 8 + 8	1/4	1	0	0	1	0
3	3443	8 + 8 + 8	1/4	0	0	1	0	1
4	44	8	1/4	0	0	1	0	1
Sum				6	0	6	8	8

$$\text{Sum} = \text{sum of (travel time)} \times (\text{weight}) \times (\text{edge use})$$

Display 9.15 Computing the Poincare bound for the smallest eigenvalue

The busiest edges are the self-loops 11 and 44, and the value of i^* is 8. The resulting bound is $\lambda_4 \geq -1 + \frac{2}{8} = -0.75$. The actual value is $\lambda_4 = -.7072$.

Drill.

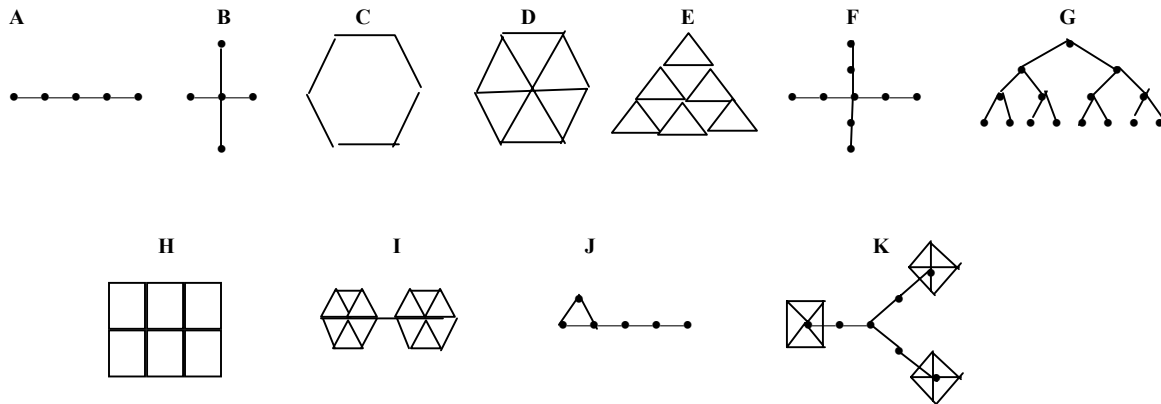
51. Compute $\iota(e)$ for each edge of graph A in Display 9.12 (Exercise 41), following Example 9.7 as a guide. Find the Poincare bound for the smallest eigenvalue, and compare it with the actual value, which is -0.5 .
52. The busiest edges of graph C in Display 9.12 (Exercise 42) are 12, 13, and 23, which are equally busy. Compute $\iota(e)$ for these edges, again following Example 9.7. Then compute the Poincare bound for the smallest eigenvalue. (The actual value is -0.857 .)
53. Consider a 3-cycle with tail of length t . Find the Poincare bound for the smallest eigenvalue, as a function of t .
54. Compute $\iota(e)$ for each edge 12, 23 and 13 of the graph associated with the Markov chain whose transition matrix is given in Exercise 41. Then find the Poincare bound on the smallest eigenvalue.
55. Find $\iota(e)$ for the edges of the Metropolized random walk on the co-occurrence matrices with row sums 1,2,1 and column sums 1,2,1. Then find the Poincare bound on the smallest eigenvalue.

Investigation

Display 9.16 shows 11 different graphs. Each graph is in fact intended to represent an entire family of graphs with similar structure. For example, the 5-point line (A) belongs to the family that includes the 2-point line, 3-point line, etc. The generic member of this family is the k -point line. Similarly, the star with 4 exterior points (B) represents the family of stars with k exterior points, and the hexagon (C) represents the family of k -gons. Graphs (A-E) all represent one-parameter families – there is only one number (parameter) to vary to generate the whole family. Graph F, the “starfish” with 8 arms, each of length 2, represents a two-parameter family. You can vary both the number of arms and their length. Graph G, the binary tree of depth 3 represents another two-parameter family. If you “read” the graph from top to bottom, you see that at each vertex the tree splits into 2 branches (making it a binary tree), and that this occurs 3 times before you get to the bottom – the depth is three. Graphs H, I and J represent other two-parameter families. Finally, graph K represents a 3-parameter family: You can vary the number of arms, the length of each arm, and the number of points in each terminal k -gon-with-center.

56. Choose a parametric family of graphs. As you vary the parameters, look for patterns and generalizations. What can you say about the greediest set? the escape fraction? the busiest edge? the value of κ^* ? of ι^* ? For which graphs is the second eigenvalue λ_2 closer to the lower Cheeger bound $1 - 2h_*$ than it is to the upper Cheeger bound $1 - h_*^2$? For which graphs is λ_2 closer to the upper Poincare bound $1 - 1/\kappa^*$ than it is to the upper

Cheeger bound $1 - h_*^2$? What are the geometric features that affect whether one of these bounds is good or bad?



Display 9.16 Representatives of various parametric families of graphs