BOLYAI SOCIETY
MATHEMATICAL STUDIES, 2

Combinatorics,
Paul Erdős is Eighty (Volume 2)
Keszthely (Hungary), 1993, pp. 1–46.

Random Walks on Graphs: A Survey

L. LOVÁSZ

Dedicated to the marvelous random walk
of Paul Erdős
through universities, continents, and mathematics

Various aspects of the theory of random walks on graphs are surveyed. In particular, estimates on the important parameters of access time, commute time, cover time and mixing time are discussed. Connections with the eigenvalues of graphs and with electrical networks, and the use of these connections in the study of random walks is described. We also sketch recent algorithmic applications of random walks, in particular to the problem of sampling.

0. Introduction

Given a graph and a starting point, we select a neighbor of it at random, and move to this neighbor; then we select a neighbor of this point at random, and move to it etc. The (random) sequence of points selected this way is a *random walk* on the graph.

A random walk is a finite Markov chain that is time-reversible (see below). In fact, there is not much difference between the theory of random walks on graphs and the theory of finite Markov chains; every Markov chain can be viewed as random walk on a directed graph, if we allow weighted edges. Similarly, time-reversible Markov chains can be viewed as random walks on undirected graphs, and symmetric Markov chains, as random walks on regular symmetric graphs. In this paper we'll formulate the results in terms of random walks, and mostly restrict our attention to the undirected case.

Random walks arise in many models in mathematics and physics. In fact, this is one of those notions that tend to pop up everywhere once you begin to look for them. For example, consider the shuffling of a deck of cards. Construct a graph whose nodes are all permutations of the deck, and two of them are adjacent if they come by one shuffle move (depending on how you shuffle). Then repeated shuffle moves correspond to a random walk on this graph (see Diaconis [20]). The Brownian motion of a dust particle is random walk in the room. Models in statistical mechanics can be viewed as random walks on the set of states.

The classical theory of random walks deals with random walks on simple, but infinite graphs, like grids, and studies their qualitative behaviour: does the random walk return to its starting point with probability one? does it return infinitely often? For example, Pólya (1921) proved that *if we do a random walk on a d-dimensional grid, then (with probability 1) we return to the starting point infinitely often if $d = 2$, but only a finite number of times if $d \geq 3$.* See Doyle and Snell [25]; for more recent results on random walks on infinite graphs, see also Thomassen [65].

More recently, random walks on more general, but finite graphs have received much attention, and the aspects studied are more quantitative: how long we have to walk before we return to the starting point? before we see a given node? before we see all nodes? how fast does the distribution of the walking point tend to its limit distribution?

As it turns out, the theory of random walks is very closely related to a number of other branches of graph theory. Basic properties of a random walk are determined by the *spectrum* of the graph, and also by *electrical resistance* of the electric network naturally associated with graphs. There are a number of other processes that can be defined on a graph, mostly describing some sort of "diffusion" (chip-firing, load-balancing in distributed networks etc.), whose basic parameters are closely tied with the above-mentioned parameters of random walks. All these connections are very fruitful and provide both tools for the study and opportunities for applications of random walks. However, in this survey we shall restrict our attention to the connections with eigenvalues and electrical networks.

Much of the recent interest in random walks is motivated by important algorithmic applications. Random walks can be used to reach "obscure" parts of large sets, and also to generate random elements in large and complicated sets, such as the set of lattice points in a convex body or the set of perfect matchings in a graph (which, in turn, can be used to the asymptotic enumeration of these objects). We'll survey some of these applications along with a number of more structural results.

We mention three general references on random walks and finite Markov chains: Doyle and Snell [25], Diaconis [20] and the forthcoming book of Aldous [3].

Acknowledgement. My thanks are due to Peter Winkler, András Lukács and Andrew Kotlov for the careful reading of the manuscript of this paper, and for suggesting many improvements.

1. Basic notions and facts

Let $G = (V, E)$ be a connected graph with n nodes and m edges. Consider a *random walk* on G: we start at a node v_0; if at the t-th step we are at a node v_t, we move neighbor of v_t with probability $1/d(v_t)$. Clearly, the sequence of random nodes $(v_t : t = 0, 1, \ldots)$ is a Markov chain. The node v_0 may be fixed, but may itself be drawn from some initial distribution P_0. We denote by P_t the distribution of v_t:

$$P_t(i) = \mathrm{Prob}(v_t = i).$$

We denote by $M = (p_{ij})_{i,j\in V}$ the matrix of transition probabilities of this Markov chain. So

$$p_{ij} = \begin{cases} 1/d(i), & \text{if } ij \in E, \\ 0, & \text{otherwise.} \end{cases} \tag{1.1}$$

Let A_G be the adjacency matrix of G and let D denote the diagonal matrix with $(D)_{ii} = 1/d(i)$, then $M = DA_G$. If G is d-regular, then $M = (1/d)A_G$. The rule of the walk can be expressed by the simple equation

$$P_{t+1} = M^T P_t,$$

(the distribution of the t-th point is viewed as a vector in $\mathbb{R}^V$), and hence

$$P_t = (M^T)^t P_0.$$

It follows that the probability p_{ij}^t that, starting at i, we reach j in t steps is given by the ij-entry of the matrix M^t.

If G is regular, then this Markov chain is *symmetric:* the probability of moving to u, given that we are at node v, is the same as the probability of moving to node v, given that we are at node u. For a non-regular graph G, this property is replaced by *time-reversibility:* a random walk considered backwards is also a random walk. More exactly, this means that if we look at all random walks $(v_0, \ldots, v_t)$, where v_0 is from some initial distribution P_0, then we get a probability distribution P_t on v_t. We also get a probability distribution Q on the sequences $(v_0, \ldots, v_t)$. If we reverse each sequence, we get another probability distribution Q' on such sequences. Now time-reversibility means that this distribution Q' is the same as the distribution obtained by looking at random walks starting from the distribution P_t. (We'll formulate a more handy characterization of time-reversibility a little later.)

The probability distributions $P_0, P_1, \ldots$ are of course different in general. We say that the distribution P_0 is *stationary* (or *steady-state*) for the graph G if $P_1 = P_0$. In this case, of course, $P_t = P_0$ for all $t \geq 0$; we call this walk the *stationary walk.*

A one-line calculation shows that for every graph G, the distribution

$$\pi(v) = \frac{d(v)}{2m}$$

is stationary. In particular, the uniform distribution on V is stationary if the graph is regular. It is not difficult to show that the stationary distribution is unique (here one has to use that the graph is connected).

The most important property of the stationary distribution is that if G is non-bipartite, then the distribution of v_t tends to a stationary distribution, as $t \to \infty$ (we shall see a proof of this fact, using eigenvalues, a little later). This is not true for bipartite graphs if $n > 1$, since then the distribution P_t is concentrated on one color class or the other, depending on the parity of t.

In terms of the stationary distribution, it is easy to formulate the property of time-reversibility: it is equivalent to saying that for every pair $i, j \in V$, $\pi(i)p_{ij} = \pi(j)p_{ji}$. This means that in a stationary walk, we step as often from i to j as from j to i. From (1.1), we have $\pi(i)p_{ij} = 1/(2m)$ for $ij \in E$, so we see that we move along every edge, in every given direction, with the same frequency. If we are sitting on an edge and the random walk just passed through it, then the expected number of steps before it passes through it in the same direction again is $2m$. There is a similar fact for nodes: if we are sitting at a node i and the random walk just visited this node, then the expected number of steps before it returns is $1/\pi(i) = 2m/d(i)$. If G is regular, then this "return time" is just n, the number of nodes.

2. Main parameters

We now formally introduce the measures of a random walk that play the most important role in the quantitative theory of random walks, already mentioned in the introduction.

(a) The *access time* or *hitting time* H_{ij} is the expected number of steps before node j is visited, starting from node i. The sum

$$\kappa(i,j) = H(i,j) + H(j,i)$$

is called the *commute time:* this is the expected number of steps in a random walk starting at i, before node j is visited and then node i is reached again. There is also a way to express access times in terms of commute times, due to Tetali [63]:

$$H(i,j) = \frac{1}{2}\left(\kappa(i,j) + \sum_u \pi(u)[\kappa(u,j) - \kappa(u,i)]\right). \tag{2.1}$$

This formula can be proved using either eigenvalues or the electrical resistance formulas (sections 3 and 4).

(b) The *cover time* (starting from a given distribution) is the expected number of steps to reach every node. If no starting node (starting distribution) is specified, we mean the worst case, i.e., the node from which the cover time is maximum.

(c) The *mixing rate* is a measure of how fast the random walk converges to its limiting distribution. This can be defined as follows. If the graph is non-bipartite, then $p_{ij}^{(t)} \to d_j/(2m)$ as $t \to \infty$, and the mixing rate is

$$\mu = \limsup_{t\to\infty} \max_{i,j} \left| p_{ij}^{(t)} - \frac{d_j}{2m} \right|^{1/t}.$$

(For a bipartite graph with bipartition $\{V_1, V_2\}$, the distribution of v_t oscillates between "almost proportional to the degrees on V_1'' and "almost proportional to the degrees on V_2''. The results for bipartite graphs are similar, just a bit more complicated to state, so we ignore this case.)

One could define the notion of "mixing time" as the number of steps before the distribution of v_t will be close to uniform (how long should we shuffle a deck of cards?). This number will be about $(\log n)/(1-\mu)$. However, the exact value depends on how (in which distance) the phrase "close" is interpreted, and so we do not introduce this formally here. In section 5 we will discuss a more sophisticated, but "canonical" definition of mixing time.

The surprising fact, allowing the algorithmic applications mentioned in the introduction, is that this "mixing time" may be much less than the number of nodes; for an expander graph, for example, this takes only $O(\log n)$ steps!

Example 1. To warm up, let us determine the access time for two points of a path on nodes $0, \ldots, n-1$.

First, observe that the access time $H(k-1, k)$ is one less than the expected return time of a random walk on a path with $k+1$ nodes, starting at the last node. As remarked, this return time is $2k$, so $H(k-1,k) = 2k-1$.

Next, consider the access times $H(i,k)$ where $0 \le i < k \le n$. In order to reach k, we have to reach node $k-1$; this takes, on the average, $H(i,k-1)$ steps. From here, we have to get to k, which takes, on the average, $2k-1$ steps (the nodes beyond the k-th play no role). This yields the recurrence

$$H(i,k) = H(i,k-1) + 2k - 1,$$

whence $H(i,k) = (2i+1) + (2i+3) + \ldots + (2k-1) = k^2 - i^2$. In particular, $H(0,k) = k^2$ (this formula is closely related to the well-known fact that Brownian motion takes you distance $\sqrt{t}$ in t time).

Assuming that we start from 0, the cover time of the path on n nodes will also be $(n-1)^2$, since it suffices to reach the other endnode. The reader might find it entertaining to figure out the cover time of the path when starting from an internal node.

¿From this it is easy to derive that the access time between two nodes at distance k of a circuit of length n is $k(n-k)$. To determine the cover time $f(n)$ of the circuit, note that it is the same as the time needed on a very long path, starting from the midpoint, to reach n nodes. Now we have to reach first $n-1$ nodes, which takes $f(n-1)$ steps on the average. At this point, we have a subpath with $n-1$ nodes covered, and we are sitting at one of its endpoints. To reach a new node means to reach one of the endnodes of a path with $n+1$ nodes from a neighbor of an endnode. Clearly, this is the same as the access time between two consecutive nodes of a circuit of length n. This leads to the recurrence

$$f(n) = f(n-1) + (n-1),$$

and through this, to the formula $f(n) = n(n-1)/2$.

Example 2. As another example, let us determine the access times and cover times for a complete graph on nodes $\{0, \ldots, n-1\}$. Here of course we may assume that we start from 0, and to find the access times, it suffices to determine $H(0,1)$. The probability that we first reach node 1 in the t-th step is clearly $\left(\frac{n-2}{n-1}\right)^{t-1} \frac{1}{n-1}$, and so the expected time this happens is

$$H(0,1) = \sum_{t=1}^{\infty} t \left(\frac{n-2}{n-1}\right)^{t-1} \frac{1}{n-1} = n-1.$$

The cover time for the complete graph is a little more interesting, and is closely related to the so-called Coupon Collector Problem (if you want to collect each of n different coupons, and you get every day a random coupon in the mail, how long do you have to wait?). Let τ_i denote the first time when i vertices have been visited. So $\tau_1 = 0 < \tau_2 = 1 < \tau_3 < \ldots < \tau_n$. Now $\tau_{i+1} - \tau_i$ is the number of steps while we wait for a new vertex to occur — an event with probability $(n-i)/(n-1)$, independently of the previous steps. Hence

$$E(\tau_{i-1} - \tau_i) = \frac{n-1}{n-i},$$

and so the cover time is

$$E(\tau_n) = \sum_{i=1}^{n-1} E(\tau_{i+1} - \tau_i) = \sum_{i=1}^{n-1} \frac{n-1}{n-i} \approx n \log n.$$

A graph with particularly bad random walk properties is obtained by taking a clique of size $n/2$ and attach to it an endpoint of a path of length $n/2$. Let i be any node of the clique and j, the "free" endpoint of the path. Then

$$H(i,j) = \Omega(n^3).$$

In fact, starting from i, it takes, on the average, $n/2-1$ moves to reach the attachment node u; then with probability $1-2/n$, we move to another node of the clique, and we have to come back about $n/2$ times before we can expect to move into the path. But one can argue that on a path of length $n/2$, if we start a random walk from one end, we can expect to return to the starting node $n/2$ times. Each time, we can expect to spend $\Omega(n^2)$ steps to get back on the path.

Bounds on the main parameters

We start with some elementary arguments (as we shall see later, eigenvalues provide more powerful formulas). Recall that if we have just traversed an edge, then the expected number of steps before it is traversed in this direction again is $2m$. In other words, if we start from node i, and j is an adjacent node, then the expected time before the edge ji is traversed in this direction is $2m$. Hence the commute time for two adjacent nodes is bounded by $2m$. It follows that the commute time between two nodes at distance r is at most $2mr < n^3$. A similar bound follows for the cover time, by considering a spanning tree. It is an important consequence of this fact that these times are polynomially bounded. (It should be remarked that this does not remain true on directed graphs.)

The following proposition summarizes some known results about cover and commute times. An $O(n^3)$ upper bound on the access and cover times was first obtained by Aleliunas, Karp, Lipton, Lovász and Rackoff [4]. The upper bound on the access time in (a), which is best possible, is due to Brightwell and Winkler [13].

It is conjectured that the graph with smallest cover time is the complete graph (whose cover time is $\approx n \log n$, as we have seen, and this is of course independent of the starting distribution). Aldous [1] proved that this is true up to a constant factor if the starting point is drawn at random, from the stationary distribution. The asymptotically best possible upper and lower bounds on the cover time given in (b) are recent results of Feige [31,32].

For the case of regular graphs, a quadratic bound on the cover time was first obtained by Kahn, Linial, Nisan and Saks (1989). The bound given in (c) is due to Feige [33].

Theorem 2.1. *(a) The access time between any two nodes of a graph on n nodes is at most*

$$(4/27)n^3 - (1/9)n^2 + (2/3)n - 1 \text{ if } n \equiv 0 \pmod 3,$$

$$(4/27)n^3 - (1/9)n^2 + (2/3)n - (29/27) \text{ if } n \equiv 1 \pmod 3,$$

$$(4/27)n^3 - (1/9)n^2 + (4/9)n - (13/27) \text{ if } n \equiv 2 \pmod 3.$$

(b) *The cover time from any starting node in a graph with n nodes is at least $(1 - o(1))n \log n$ and at most $(4/27 + o(1))n^3$.*

(c) *The cover time of a regular graph on n nodes is at most $2n^2$.*

It is a trivial consequence of these results that the commute time between any two nodes is also bounded by n^3, and for a regular graph, the access time is at most $2n^2$ and the commute time is bounded by $4n^2$.

No non-trivial lower bound on the commute time can be found in terms of the number of nodes: the commute time between the two nodes in the smaller color class of the complete bipartite graph $K_{2,n}$ is 8. It is true, however, that $\kappa(u, v) \geq 2m/d(u)$ for all u and v (cf. Proposition 2.3 below, and also Corollary 3.3). In particular, the commute time between two nodes of a regular graph is always at least n.

The situation is even worse for the access time: this can remain bounded even for regular graphs. Consider a regular graph (of any degree $d \geq 3$) that has a cutnode u; let $G = G_1 \cup G_2$, $V(G_1) \cap V(G_2) = \{u\}$, and let v be a node of G_1 different from u. Then the access time from v to u is the same as the access time from v to u in G_1, which is independent of the size of the rest of the graph.

One class of graphs for which a lower bound of $n/2$ for any access time can be proved is the class of graphs with transitive automorphism group; cf. Corollary 2.6.

Symmetry and access time

The access time from i to j may be different from the access time from j to i, even in a regular graph. There is in fact no way to bound one of these numbers by the other. In the example at the end of the last paragraph, walking from u to v we may, with probability at least $1/d$, step to a node of G_2. Then we have to walk until we return to u; the expected time before this happens more than $|V(G_2)|$. So $\alpha(u, v) > |V(G_2)|$, which can be arbitrarily large independently of $\alpha(v, u)$.

Still, one expects that time-reversibility should give some sort of symmetry of these quantities. We formulate two facts along these lines. The first is easy to verify by looking at the walks “backwards”.

Proposition 2.2. *If u and v have the same degree, then the probability that a random walk starting at u visits v before returning to u is equal to the probability that a random walk starting at v visits u before returning to v.*

(If the degrees of u and v are different, then the ratio of the given probabilities is $\pi(v)/\pi(u) = d(v)/d(u)$.)

The probabilities in Proposition 2.2 are related to the commute time $\kappa(u,v)$ in an interesting way:

Proposition 2.3. *The probability that a random walk starting at u visits v before returning to u is $1/(\kappa(u,v)\pi(u))$.*

Proof. Let q denote the probability in question. Let τ be the first time when a random walk starting at u returns to u and σ, the first time when it returns to u after visiting v. We know that $E(\tau) = 2m/d(u)$ and by definition, $E(\sigma) = \kappa(u,v)$. Clearly $\tau \le \sigma$ and the probability of $\tau = \sigma$ is exactly q. Moreover, if $\tau < \sigma$ then after the first τ steps, we have to walk from u until we reach v and then return to u. Hence $E(\sigma - \tau) = (1-q)E(\sigma)$, and hence

$$q = \frac{E(\tau)}{E(\sigma)} = \frac{2m}{d(u)\kappa(u,v)}. \qquad \blacksquare$$

A deeper symmetry property of access times was discovered by Coppersmith, Tetali and Winkler [19]. This can also be verified by elementary means considering walks visiting three nodes u, v and w, and then reversing them, but the details are not quite simple.

Theorem 2.4. *For any three nodes u, v and w,*

$$H(u,v) + H(v,w) + H(w,u) = H(u,w) + H(w,v) + H(v,u).$$

An important consequence of this symmetry property is the following.

Corollary 2.5. *The nodes of any graph can be ordered so that if u precedes v then $H(u,v) \le H(v,u)$. Such an ordering can be obtained by fixing any node t, and order the nodes according to the value of $H(u,t) - H(t,u)$.*

Proof. Assume that u precedes v in the ordering described. Then $H(u,t) - H(t,u) \le H(v,t) - H(t,v)$ and hence $H(u,t) + H(t,v) \le H(v,t) + H(t,u)$. By Theorem 2.4, this is equivalent to saying that $H(u,v) \le H(v,u)$. $\blacksquare$

This ordering is not unique, because of the ties. But if we partition the nodes by putting u and v in the same class if $H(u,v) = H(v,u)$ (this is an equivalence relation by Proposition 2.4), then there is a well-defined ordering of the equivalence classes, independent of the reference node t. The nodes in the lowest class are "difficult to reach but easy to get out of", the nodes in the highest class are "easy to reach but difficult to get out of". It is worth formulating a consequence of this construction:

Corollary 2.6. *If a graph has a vertex-transitive automorphism group then $H(i,j) = H(j,i)$ for all nodes i and j.*

Access time and cover time

The access times and commute times of a random walk have many nice properties and are relatively easy to handle. The cover time is more elusive. But there is a very tight connection between access times and cover times, discovered by Matthews [56]. (See also Matthews [57]; this issue of the *J. Theor. Probability* contains a number of other results on the cover time.)

Theorem 2.7. *The cover time from any node of a graph with n nodes is at most $(1 + (1/2) + \ldots + (1/n))$ times the maximum access time between any two nodes, and at least $(1 + (1/2) + \ldots + (1/n))$ times the minimum access time between two nodes.*

Let us sketch a simple proof for the somewhat weaker upper bound of $2\log_2 n$ times the maximum access time.

Lemma 2.8. *Let b be the expected number of steps before a random walk visits more than half of the nodes, and let h be the maximum access time between any two nodes. Then $b \leq 2h$.*

¿From this lemma, the theorem is easy. The lemma says that in $2h$ steps we have seen more than half of all nodes; by a similar argument, in another $2h$ steps we have seen more than half of the rest etc.

Proof. Assume, for simplicity, that $n = 2k+1$ is odd. Let α_v be the time when node v is first visited. Then the time β when we reach more than half of the nodes is the $(k+1)$-st largest of the α_v. Hence

$$\sum_v \alpha_v \geq (k+1)\beta,$$

and so

$$b = E(\beta) \leq \frac{1}{k+1}\sum_v E(\alpha_v) \leq \frac{n}{k+1}h < 2h. \qquad \blacksquare$$

Monotonicity

Let G' be obtained from the graph G by adding a new edge ab. Since this new graph is denser, one expects that a random walk on it turns back less frequently, and therefore the access times, commute times, and cover times decrease. As it turns out, this does not hold in general.

First, it is easy to see that some access times may increase dramatically if an edge is added. Let G be a path on n nodes, with endpoints a and b. Let $s = a$ and let t be the unique neighbor of s. Then the access time from s to t is 1. On the other hand, if we add the edge (a, b) then with probability $1/2$, we have to make more than one step, so the access time from s to t will be larger than one; in fact, it jumps up to $n - 1$, as we have seen.

One monotonicity property of access time that does hold is that if an edge incident with t is added, then the access time from s to t is not larger in G' than in G.

The commute time, which is generally the best behaved, is not monotone either. For example, the commute time between two opposite nodes of a 4-cycle is 8; if we add the diagonal connecting the other two nodes, the commute time increases to 10. But the following "almost monotonicity" property is true (we'll return to its proof in section 4).

Theorem 2.9. *If G' arises from a graph G by adding a new edge, and G has m edges, then the commute time between any two nodes in G' is at most $1 + 1/m$ times the commute time in G. In other words, the quantity $\kappa(s, t)/m$ does not decrease.*

We discuss briefly another relation that one intuitively expects to hold: that access time increases with distance. While such intuition is often misleading, the following results show a case when this is true (Keilson [42]).

Theorem 2.10. *Let G be a graph and $t \in V(G)$.*

(a) If we choose s uniformly from the set of neighbors of t, then the expectation of $H(s, t)$ is exactly $(2m/d(t)) - 1$.

(b) If we choose s from the stationary distribution over V, then the expectation of $H(s, t)$ is at least $\frac{2m}{d(t)}\left(1 - \frac{d(t)}{2m}\right)^2$. So if we condition on $s \neq t$, the expectation of $H(s, t)$ is at least $(2m/d(t)) - 1$.

(c) If we choose t from the stationary distribution over V, then the expectation of $H(s, t)$ is at least $n - 2 + 1/n$.

(a) is just a restatement of the formula for the return time. The proof of (b) and (c) uses eigenvalue techniques. It is easy to derive either from (b) or (c) that $\max_{s,t} H(s,t) \geq n-1$. We remark that the expectation in (c) is independent of s (see formula (3.3)).

Applications of the cover time and commute time bounds

Perhaps the first application of random walk techniques in computer science was the following (Aleliunas, Karp, Lipton, Lovász and Rackoff [4]). Let $G = (V, E)$ be a connected d-regular graph, $v_0 \in V(G)$, and assume that at each node, the ends of the edges incident with the node are labelled $1, 2, \ldots, d$. A *traverse sequence* (for this graph, starting point, and labelling) is a sequence $(h_1, h_2, \ldots, h_t) \subseteq \{1, \ldots, d\}^t$ such that if we start a walk at v_0 and at the i^{th} step, we leave the current node through the edge labelled h_i, then we visit every node. A *universal traverse sequence* (for parameters n and d) is a sequence which is a traverse sequence for every d-regular graph on n nodes, every labelling of it, and every starting point.

It is quite surprising that such sequences exist, and in fact need not be too long:

Theorem 2.11. *For every $d \geq 2$ and $n \geq 3$, there exists a universal traverse sequence of length $O(d^2n^3 \log n)$.*

A consequence of this fact is that the reachability problem on undirected graphs is solvable in non-uniform logspace. We do not discuss the details.

Proof. The "construction" is easy: we consider a random sequence. More exactly, let $t = 8dn^3 \log n$, and let $H = (h_1, \ldots, h_t)$ be randomly chosen from $\{1, \ldots, d\}^t$. For a fixed G, starting point, and labelling, the walk defined by H is just a random walk; so the probability p that H is not a traverse sequence is the same as the probability that a random walk of length t does not visit all nodes.

By Theorem 2.1, the expected time needed to visit all nodes is at most $2n^2$. Hence (by Markov's Inequality) the probability that after $4n^2$ steps we have not seen all nodes is less than $1/2$. Since we may consider the next $4n^2$ steps as another random walk etc., the probability that we have not seen all nodes after t steps is less than $2^{-t/(4n^2)} = n^{-2nd}$.

Now the total number of d-regular graphs G on n nodes, with the ends of the edges labelled, is less than n^{dn} (less than n^d choices at each node), and so the probability that H is not a traverse sequence for one of these graphs, with some starting point, is less than $nn^{nd}n^{-2nd} < 1$. So at least one sequence of length t is a universal traverse sequence. ∎

The results of Coppersmith, Tetali and Winkler [19] discussed above served to solve the following problem: let us start two random walks on a graph simultaneously; how long does it take before they collide? There are variations depending on whether the two random walks step simultaneously, alternatingly, or in one of several other possible ways. Here we only consider the worst case, in which a "schedule daemon" determines which random walk moves at any given time, whose aim is to prevent collision as long as possible.

The motivation of this problem is a self-stabilizing token-management scheme for a distributed computing network. The "token" is the authorization for the processor carrying it to perform some task, and at any time, only one processor is supposed to carry it. Assume that by some disturbance, two processors carry the token. They pass it around randomly, until the two tokens collide; from then on, the system is back to normal. How long does this take?

Let $M(u,v)$ denote the expected number of steps before two random walks, starting from nodes u and v, collide. It is clear that $M(u,v) \geq H(u,v)$ (v may never wake up to move). Coppersmith, Tetali and Winkler [19] prove the nice inequality

$$M(u,v) \leq H(u,v) + H(v,w) - H(w,u)$$

for some vertex w. Thus it follows that the collision time is $O(n^3)$.

3. The eigenvalue connection

Recall that the probability p_{ij}^t of the event that starting at i, the random walk will be at node j after t steps, is an entry of the matrix M^t. This suggests that the powerful methods of the spectral theory of matrices can be used.

The matrix M has largest eigenvalue 1, with corresponding left eigenvalue π and corresponding right eigenvalue $\mathbf{1}$, the all-1 vector on V. In fact, $M^T\pi = \pi$ expresses the fact that π is the stationary distribution, while $M\mathbf{1} = \mathbf{1}$ says that exactly one step is made from each node.

Unfortunately, M is not symmetric unless G is regular; but it is easy to bring it to a symmetric form. In fact, we kow that $M = DA$, where $A = A_G$ is the adjacency matrix of G and D is the diagonal matrix in which the i-th diagonal entry is $1/d(i)$. Consider the matrix $N = D^{1/2}AD^{1/2} = D^{-1/2}MD^{1/2}$. This is symmetric, and hence can be written in a spectral form:

$$N = \sum_{k=1}^{n} \lambda_k v_k v_k^T,$$

where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ are the eigenvalues of N and $v_1, \ldots, v_n$ are the corresponding eigenvectors of unit length. Simple substitution shows that $w_i = \sqrt{d(i)}$ defines an eigenvector of N with eigenvalue 1. Since this eigenvector is positive, it follows from the Frobenius-Perron Theorem that $\lambda_1 = 1 > \lambda_2 \geq \ldots \geq \lambda_n \geq -1$ and that (possibly after flipping signs) $v_1 = (1/\sqrt{2m})w$, i.e., $v_{1i} = \sqrt{d(i)/2m} = \sqrt{\pi(i)}$. It also follows by standard arguments that if G is non-bipartite then $\lambda_n > -1$.

Now we have

$$M^t = D^{1/2}N^tD^{-1/2} = \sum_{k=1}^{n} \lambda_k^t D^{1/2} v_k v_k^T D^{-1/2} = Q + \sum_{k=2}^{n} \lambda_k^t D^{1/2} v_k v_k^T D^{-1/2}$$

where $Q_{ij} = \pi(j)$. In other words,

$$p_{ij}^t = \pi(j) + \sum_{k=2}^{n} \lambda_k^t v_{ki} v_{kj} \sqrt{\frac{d(j)}{d(i)}}. \tag{3.1}$$

If G is not bipartite then $|\lambda_k| < 1$ for $k = 2, \ldots, n$, and hence

$$p_{ij}^t \to \pi(j) \qquad (t \to \infty)$$

as claimed above. We shall return to the rate of this convergence later.

Spectra and access times

We start a more in-depth study of connections between random walks and spectra by deriving a spectral formula for access times. Let $H \in \mathbb{R}^{V\times V}$ denote the matrix in which $H_{ij} = H(i,j)$, the access time from i to j. Let $\Gamma(i)$ be the set of neighbors of node i. The key equation is that if $i \neq j$ then

$$H(i,j) = 1 + \frac{1}{d(i)} \sum_{v\in\Gamma(i)} H(v,j)$$

(since the first step takes us to a neighbor v of i, and then we have to reach j from there). Expressing this equation in matrix notation, we get that $F = J + MH - H$ is a diagonal matrix. Moreover,

$$F^T\pi = J\pi + H^T(M-I)^T\pi = J\pi = \mathbf{1},$$

whence

$$(F)_{ii} = \frac{1}{\pi(i)} = \frac{2m}{d(i)}.$$

Thus $F = 2mD$, i.e.,

$$(I - M)H = J - 2mD. \tag{3.2}$$

We want to solve this "matrix equation" for H. Of course, this is not possible since $I - M$ is singular; in fact, with every X satisfying (3.2) (in place of H), every matrix $X + \mathbf{1}a^T$ also satisfies it for any vector a. But these are all, as elementary linear algebra shows, and so a can be determined using the relations

$$H(i,i) = 0 \qquad (i \in V).$$

So if we find any solution of (3.2), we can obtain H by subtracting the diagonal entry from each column.

Let M^* denote the matrix $\mathbf{1}\pi^T$, i.e., $M^*_{ij} = \pi(j)$ (note that M^* is the limit of M^t as $t \to \infty$). Substitution shows that the matrix $X = (I - M + M^*)^{-1}(J - 2mD)$ satisfies (3.2). Diagonalizing M as above, we get the following formula:

Theorem 3.1.

$$H(s,t) = 2m \sum_{k=2}^{n} \frac{1}{1-\lambda_k} \left(\frac{v_{kt}^2}{d(t)} - \frac{v_{ks}v_{kt}}{\sqrt{d(s)d(t)}} \right).$$

As an immediate corollary we obtain a similar formula for the commute time:

Corollary 3.2.

$$\kappa(s,t) = 2m \sum_{k=2}^{n} \frac{1}{1-\lambda_k} \left(\frac{v_{kt}}{\sqrt{d(t)}} - \frac{v_{ks}}{\sqrt{d(s)}} \right)^2.$$

Using that

$$\frac{1}{2} \le \frac{1}{1-\lambda_k} \le \frac{1}{1-\lambda_2}$$

along with the orthogonality of the matrix (v_{ks}), we get

Corollary 3.3.

$$m\left(\frac{1}{d(s)}+\frac{1}{d(t)}\right)\le \kappa(s,t)\le \frac{2m}{1-\lambda_2}\left(\frac{1}{d(s)}+\frac{1}{d(t)}\right).$$

If the graph is regular, the lower bound is n. If we have a *expander graph,* which can be characterized as a regular graph for which $1/(1-\lambda_2)=O(1)$, then it follows that the commute time between any pair of nodes is $\Theta(n)$.

In these formulas, the appearence of $1-\lambda_k$ in the denominators suggest that it will be necessary to find good bounds on the *spectral gap:* the difference $1-\lambda_2=\lambda_1-\lambda_2$. This is an important parameter for many other studies of graphs, and we shall return to its study in the next section.

To warm up to the many applications of Theorem 3.1, the reader is encouraged to give a proof of the week symmetry property of access times expressed in Theorem 2.4, and of the expression for access times in terms of commute times (2.1). Another easy corollary is obtained if we average the access time over all t. We have

$$\begin{aligned}\sum_t \pi(t)H(s,t) &= \sum_t\sum_{k=2}^n \frac{1}{1-\lambda_k}\left(v_{kt}^2 - v_{kt}v_{ks}\sqrt{\frac{d(t)}{d(s)}}\right)\\ &= \sum_{k=2}^n \frac{1}{1-\lambda_k}\left(\sum_t v_{kt}^2 - v_{ks}\sqrt{\frac{1}{d(s)}}\sum_t v_{kt}\sqrt{d(t)}\right).\end{aligned}$$

Using that v_k is of unit length and it is orthogonal to v_1 for $k\ge 2$, we get the nice formula

$$\sum_t \pi(t)H(s,t)=\sum_{k=2}^n\frac{1}{1-\lambda_k}. \tag{3.3}$$

Note that this value is independent of the starting node s.

As another application, we find the access time between two antipodal nodes of the k-cube Q_k. Let $\mathbf{0}=(0,\dots,0)$ and $\mathbf{1}=(1,\dots,1)$ represent two antipodal nodes of the k-cube. As is well known, we get an eigenvector v_b of M (or A) for every 0-1 vector $b\in\{0,1\}^k$, defined by $(v_b)_x=(-1)^{b\cdot x}$. The corresponding eigenvalue of M is $1-(2/k)b\cdot\mathbf{1}$. Normalizing v_b and substituting in Theorem 3.1, we get that

$$H(\mathbf{0},\mathbf{1})=k\sum_{j=1}^k\binom{k}{j}\frac{1}{2j}(1-(-1)^j).$$

To find the asymptotic value of this expression, we substitute $\binom{k}{j} = \sum_{p=0}^{k-1}\binom{p}{j-1}$, and get

$$H(\mathbf{0},\mathbf{1}) = k\sum_{j=1}^{k}\sum_{p=0}^{k-1}\frac{1}{2j}\binom{p}{j-1}(1-(-1)^j)$$
$$= k\sum_{p=0}^{k-1}\frac{1}{2(p+1)}\sum_{j=1}^{k}\binom{p+1}{j}(1-(-1)^j)$$
$$= k\sum_{p=0}^{k-1}\frac{2^p}{p+1} = 2^{k-1}\sum_{j=0}^{k-1}\frac{1}{2^j}\frac{k}{k-j} \sim 2^k.$$

(It is easy to see that the exact value is always between 2^k and 2^{k+1}.)

As a further application, let us prove that "more distant targets are more difficult to reach" (Theorem 2.10.b). The argument is similar to the proof of (3.3). We have

$$\sum_s \pi(s)H(s,t) = \sum_s\sum_{k=2}^{n}\frac{1}{1-\lambda_k}\left(v_{kt}^2\frac{d(s)}{d(t)} - v_{kt}v_{ks}\sqrt{\frac{d(s)}{d(t)}}\right).$$

Using again that v_k is orthogonal to v_1 for $k \geq 2$, we have

$$\sum_s \pi(s)H(s,t) = \frac{2m}{d(t)}\sum_{k=2}^{n}\frac{1}{1-\lambda_k}v_{kt}^2.$$

By the inequality between arithmetic and harmonic means (considering the v_{kt}^2 as weights), we have

$$\frac{\sum_{k=2}^{n}\frac{1}{1-\lambda_k}v_{kt}^2}{\sum_{k=2}^{n}v_{kt}^2} \geq \frac{\sum_{k=2}^{n}v_{kt}^2}{\sum_{k=2}^{n}(1-\lambda_k)v_{kt}^2}.$$

Now here

$$\sum_{k=2}^{n}v_{kt}^2 = \sum_{k=1}^{n}v_{kt}^2 - \pi(t) = 1-\pi(t)$$

and

$$\sum_{k=2}^{n}(1-\lambda_k)v_{kt}^2 = \sum_{k=1}^{n}(1-\lambda_k)v_{kt}^2 = 1-\sum_{k=1}^{n}\lambda_k v_{kt}^2 = 1-(N)_{t,t} = 1.$$

Thus

$$\sum_s \pi(s)H(s,t) \geq \frac{1}{\pi(t)}(1-\pi(t))^2,$$

which proves the assertion.

Perhaps the most important applications of eigenvalue techniques concern the mixing rate, which we'll discuss in a separate section.

Spectra and generating functions

One may obtain spectral formulas carrying even more information by introducing the *probability generating function*

$$F(x) = \sum_{t=0}^{\infty} x^t M^t = (I - xM)^{-1}.$$

(the (i,j) entry $F_{ij}(x)$ of this matrix is the generating function for the probabilities p_{ij}^t).

Using this function, we can express other probabilities via standard techniques of generating functions. As an example, let q_{ij}^t denote the probability that the random walk starting at i hits node j for the first time in the t-th step. It is clear that

$$p_{ij}^t = \sum_{s=0}^{t} q_{ij}^s p_{jj}^{t-s},$$

We can translate this relation in terms of generating functions as follows. Let

$$G_{ij}(x) = \sum_{t=0}^{\infty} q_{ij}^t x^t,$$

then

$$F_{ij}(x) = G_{ij}(x)F_{jj}(x).$$

So the matrix $G(x) = (G_{ij}(x))$ arises from $F(x)$ by scaling each column so that the diagonal entry becomes 1.

We may use the spectral decomposition of M to get more explicit formulas. We have

$$F_{ij}(x) = \sqrt{\frac{d(j)}{d(i)}} \sum_{t=0}^{\infty} \sum_{k=1}^{n} (x\lambda_k)^t v_{ki} v_{kj} = \sqrt{\frac{d(j)}{d(i)}} \sum_{k=1}^{n} v_{ki} v_{kj} \frac{1}{1-x\lambda_k}.$$

Hence we also get the generating function

$$G_{ij}(x) = \sqrt{\frac{d(j)}{d(i)}} \sum_{k=1}^{n} v_{ki} v_{kj} \frac{1}{1 - x\lambda_k} \Bigg/ \sum_{k=1}^{n} v_{kj}^2 \frac{1}{1 - \lambda_k x}.$$

¿From this another proof of Theorem 3.1 follows easily, since

$$H(s,t) = G'_{st}(1).$$

By calculating higher derivatives, we can derive similar (though increasingly complicated) formulas for the higher moments of the time a node t is first visited.

4. The electrical connection

Let $G = (V, E)$ be a connected graph and $S \subseteq V$. A function $\phi : V \to \mathbb{R}$ is called a "harmonic function with set of poles S" if

$$\frac{1}{d(v)} \sum_{u \in \Gamma(v)} \phi(u) = \phi(v)$$

holds for every $v \notin S$ (the set S is also called the boundary of the harmonic function). Not surprisingly, harmonic functions play an important role in the study of random walks: after all, the averaging in the definition can be interpreted as expectation after one move. They also come up in the theory of electrical networks, and also in statics. This provides a connection between these fields, which can be exploited. In particular, various methods and results from the theory of electricity and statics, often motivated by physics, can be applied to provide results about random walks.

We start with describing three constructions of harmonic functions, one in each field mentioned.

(a) Let $\phi(v)$ denote the probability that a random walk starting at node v hits s before it hits t. Clearly, ϕ is a harmonic function with poles s and t. We have $\phi(s) = 1$ and $\phi(t) = 0$.
More generally, if we have a set $S \subseteq V$ and a function $\phi_0 : S \to \mathbb{R}$, then we define $\phi(v)$ for $v \in V \setminus S$ as the expectation of $\phi_0(s)$, where s is the (random) node where a random walk starting at v first hits S. Then

$\phi(v)$ is a harmonic function with pole set S. Moreover, $\phi(s) = \phi_0(s)$ for all $s \in S$.

(b) Consider the graph G as an electrical network, where each edge represents a unit resistance. Assume that an electric current is flowing through G, entering at s and leaving at t. Let $\phi(v)$ be the voltage of node v. Then ϕ is a harmonic function with poles s and t.

(c) Consider the edges of the graph G as ideal springs with unit Hooke constant (i.e., it takes h units of force to stretch them to length h). Let us nail down nodes s and t to points 1 and 0 on the real line, and let the graph find its equilibrium. The energy is a positive definite quadratic form of the positions of the nodes, and so there is a unique minimizing position, which is the equilibrium. Clearly all nodes will lie on the segment between 0 and 1, and the positions of the nodes define a harmonic function with poles s and t.
More generally, if we have a set $S \subseteq V$ and we fix the positions of the nodes in S (in any dimension), and let the remaining nodes find their equilibrium, then any coordinate of the nodes defines a harmonic function with pole set S.

Let us sum up some trivial properties of harmonic functions. Clearly, $\phi(v)$ lies between the minimum and maximum of ϕ over S. Moreover, given $S \subseteq V$ and $\phi_0 : S \to \mathbb{R}$, there is a unique harmonic function on G with pole set S extending ϕ_0. (The existence follows by either construction (a) or (c); the uniqueness follows by considering the maximum of the difference of two such functions.)

In particular it follows that every harmonic function with at most one pole is constant. We denote by ϕ_{st} the (unique) harmonic function with poles s and t such that $\phi_{st}(s) = 1$ and $\phi_{st}(t) = 0$.

Another consequence of the uniqueness property is that the harmonic functions constructed in (a) and (c), and (for the case $|S| = 2$) in (b) are the same. As an application of this idea, we show the following useful characterizations of commute times (see Nash-Williams [60], Chandra, Raghavan, Ruzzo, Smolensky and Tiwari [16]).

Theorem 4.1. (i) *Consider the graph G as an electrical network as in (b) and let R_{st} denote the resistance between nodes s and t. Then the commute time between nodes s and t is exactly $2mR_{st}$.*

(ii) *Consider the graph G as a spring structure in equilibrium, as in example (c), with two nodes s and t nailed down at 1 and 0. Then the force* *pulling the nails is*

$$\frac{1}{R_{st}} = \frac{2m}{\kappa(s,t)}.$$

The energy of the system is

$$\frac{1}{2R_{st}} = \frac{m}{\kappa(s,t)}.$$

Note that equation (2.1) can be used to express access times in terms of resistances or spring forces (Tetali [63]).

Proof. By construction (b), $\phi_{st}(v)$ is the voltage of v if we put a current through G from s to t, where the voltage of s is 0 and the voltage of t is 1. The total current through the network is $\sum_{u\in\Gamma(t)} \phi_{st}(u)$, and so the resistence is

$$R_{st} = \left(\sum_{u\in\Gamma(s)} \phi_{st}(u)\right)^{-1}.$$

On the other hand, (a) says that $\phi_{st}(u)$ is the probability that a random walk starting at u visits s before t, and hence $\frac{1}{d(t)}\sum_{u\in\Gamma(t)} \phi_{st}(u)$ is the probability that a random walk starting at t hits s before returning to t. By Proposition 2.3, this probability is $2m/d(t)\kappa(s,t)$. This proves assertion (i). The proof of (ii) is similar. ∎

Using the "topological formulas" from the theory of electrical networks for the resistance, we get the following characterization of commute time:

Corollary 4.2. *Let G be a graph and $s,t \in V$. Let G' denote the graph obtained from G by identifying s and t, and let $T(G)$ denote the number of spanning trees of G. Then*

$$\kappa(s,t) = 2m\frac{T(G')}{T(G)}.$$

The following fact is called *Raleigh's Principle* in the theory of electrical networks. We derive it as a consequence of Theorem 4.1.

Corollary 4.3. *Adding any edge to a graph G does not increase any resistance R_{st}. Consequently, no commute time $\kappa(s,t)$ is increased by more than a factor of $(m+1)/m$.*

In fact, it suffices to prove that deleting an edge from a graph G cannot increase the energy of the equilibrium configuration in the spring structure (c). Clearly, deleting an edge while keeping the positions of the nodes fixed cannot increase the energy. If we let the new graph find its equilibrium then the energy can only further decrease.

Combining Corollaries 4.2 and 4.3, a little algebraic manipulation gives the following inequality for the numbers of spanning trees in a graph G and in its subgraphs $G-e$, $G-f$, and $G-e-f$, where e and f are two edges of G:

$$T(G-e)T(G-f) \geq T(G)T(G-e-f). \tag{4.1}$$

5. Mixing rate

In several recent applications of random walks, the most important parameter is the mixing rate. Using eigenvalues, it is an easy task to determine the mixing rate in polynomial time (see below), but this result does not tell the whole story, since, as we shall see, the underlying graph in the cases of interest is exponentially large, and the computation of the eigenvalues by the tools of linear algebra is hopeless. Therefore, combinatorial techniques that lead to approximations only but are more manageable are often preferable. Two main techniques that have been used are *coupling* and *conductance.* In this section we discuss these techniques; in the next, we give several applications in algorithm design.

Mixing rate and coupling

We shall illustrate the methods for bounding the mixing rate on a special class of graphs. (For reasons of comparison, we will also apply the other methods to the same graph.) These graphs are the cartesian sum C_n^k of k circuits of length n, where n is odd. The node set of this graph is $\{0,\dots,n-1\}^k$, and two nodes $(x_1,\dots,x_k)$ and $(y_1,\dots y_k)$ are adjacent iff there exists an i, $1 \leq i \leq k$, such that $x_j = y_j$ for $j \neq i$ and $x_i \equiv y_i \pm 1 \pmod{n}$.

Let us start a random walk $(v_0, v_1, \dots)$ on C_n^k from an arbitrary initial distribution P_0. To estimate how long we have to walk to get close to the stationary distribution (which is uniform in this case), let us start another random walk $(w_0, w_1, \dots)$, in which w_0 is drawn from the uniform distribution. Of course, w_t is then uniformly distributed for all t.

The two walks are not independent; we "couple" them as follows. The vertices of C_n^k are vectors of length k, and a step in the random walk consists of changing a randomly chosen coordinate by one. We first generate the step in the first walk, by selecting a random coordinate j, $1 \le j \le k$, and a random $\varepsilon \in \{-1, 1\}$. The point v_{t+1} is obtained by adding ε to the j-th coordinate of v_t. Now the trick is that if v_t and w_t agree in the j-th coordinate, we generate w_{t+1} by adding ε to the j-th coordinate of w_t; else, we subtract ε from the j-th coordinate of w_t. (All operations are modulo n.)

The important fact is that viewing $(w_0, w_1, \dots)$ in itself, it is an entirely legitimate random walk. On the other hand, the "coupling" rule above implies that once a coordinate of v_t becomes equal to the corresponding coordinate of w_t, it remains so forever. Sooner or later all coordinates become equal, then v_t will have the same distribution as w_t, i.e., uniform.

To make this argument precise, let us look at the steps when the j-th coordinate is selected. The expected number of such steps before the two walks will have equal j-th coordinate is the average access time between two nodes of the circuit on length n, which is $(n^2-1)/6$. So the expected number of steps before *all* coordinates become equal is $k(n^2-1)/6$. By Markov's inequality, the probability that after kn^2 steps v_t and w_t are still different is less than $1/6$, and so the probability that after ckn^2 steps these points are still different is less than 6^{-c}. Hence for any T that is large enough,

$$|P(v_T \in S) - \frac{|S|}{n^k}| = |P(v_T \in S) - P(w_T \in S)| \le P(w_T \ne v_T) < 6^{-T/(kn^2)}.$$

We obtain that the mixing rate is at most $6^{-1/(kn^2)} < 1 - \frac{1}{kn^2}$.

This method is elegant but it seems that for most applications of interest, there is no simple way to find a coupling rule, and so it applies only in lucky circumstances.

Mixing rate and the eigenvalue gap

An algebraic formula for the mixing rate is easily obtained. Let $\lambda = \min\{|\lambda_2|, |\lambda_n|\}$, then from (3.1) it is easy to derive:

Theorem 5.1. *For a random walk starting at node i,*

$$|P_t(j) - \pi(j)| \le \sqrt{\frac{d(j)}{d(i)}}\lambda^t.$$

More generally,

$$|P_t(S) - \pi(S)| \le \sqrt{\frac{\pi(S)}{\pi(i)}}\lambda^t.$$

So the mixing rate is at most λ; it is not difficult to argue that equality must hold here. Thus we obtain:

Corollary 5.2. *The mixing rate of a random walk on a non-bipartite graph G is $\lambda = \max\{|\lambda_2|, |\lambda_n|\}$.*

In most applications, we don't have to worry about λ_n; for example, we can add $d(i)$ loops at each point i, which only slows down the walk by a factor of 2, but results in a graph with positive semidefinite adjacency matrix. The crucial parameter is λ_2, or rather, the "spectral gap" $1-\lambda_2$. Note that $\log(1/\lambda) \approx (1-\lambda)^{-1}$.

Theorem 5.1 concerns the convergence to the stationary distribution in terms of the total variation distance, which seems to be the most important for applications. Other measures have other, sometimes technical, adventages. For example, using the χ^2 measure has the adventage that the distance is improving after each step (Fill [34]):

$$\sum_j \frac{(P_{t+1}(j)-\pi(j))^2}{\pi(j)} \leq \lambda \sum_j \frac{(P_t(j)-\pi(j))^2}{\pi(j)}.$$

As an application of Theorem 5.1, let us determine the mixing rate of a random walk on an n-dimensional cube. This graph is bipartite, so we add loops; let's add n loops at each node. The eigenvalues of the resulting graph are $0, 2, 4, \ldots, 2n$, and so the eigenvalues of the transition matrix are $0, 1/n, 2/n, \ldots, (n-1)/n, 1$. Hence the mixing rate is $(n-1)/n$.

Next, let us do the graph C_n^k, where n is odd. The eigenvalues of C_n are $2\cos(2r\pi/n)$, $0 \leq r < n$. Hence the eigenvalues of the adjacency matrix C_n^k are all numbers

$$2\cos(2r_1\pi/n) + 2\cos(2r_2\pi/n) + \ldots + 2\cos(2r_k\pi/n)$$

(see e.g. Lovász [48], exercise 11.7). In particular, the largest eigenvalue is (of course) $2k$, the second largest is $2(k-1)+2\cos(2\pi/n)$, and the smallest is $2k\cos((n-1)\pi/n)$. From this it follows that the mixing rate is

$$1-\frac{1}{k}\left(1-\cos\frac{2\pi}{n}\right) \approx 1-\frac{2\pi^2}{kn^2}.$$

The eigenvalue gap and conductance

Let G be a graph and $S \subset V$, $S \neq \emptyset$. Let $\nabla(S)$ denote the set of edges connecting S to $V \setminus S$. We define the *conductance* of the set $S \subset V$, $S \neq \emptyset$ by

$$\Phi(S) = \frac{|\nabla(S)|}{2m\pi(S)\pi(V \setminus S)}$$

and the conductance of the graph by

$$\Phi = \min_S \Phi(S),$$

where the minimum is taken over all non-empty proper subsets $S \subset V$. If the graph is d-regular, then the conductance of S is

$$\Phi(S) = \frac{n|\nabla(S)|}{d|S| \cdot |V \setminus S|}.$$

To digest this quantity a little, note that $|\nabla(S)|/2m$ is the frequency with which a stationary random walk switches from S to $V \setminus S$; while $\pi(S)\pi(V \setminus S)$ is the frequency with which a sequence of independent random elements of V, drawn from the stationary distribution π, switches from S to $V \setminus S$. So Φ can be viewed as a certain measure of how independent consecutive nodes of the random walk are.

Sinclair and Jerrum [62] established a connection between the spectral gap and the conductance of the graph. A similar result for the related, but somewhat different parameter called *expansion rate* was proved by Alon [5] and, independently, by Dodziuk and Kendall [24]; cf. also Diaconis and Stroock [21]. All these results may be considered as discrete versions of Cheeger's inequality in differential geometry.

Theorem 5.3. $\dfrac{\Phi^2}{8} \leq 1 - \lambda_2 \leq \Phi.$

We'll sketch the proof of this fundamental inequality; but first, we state (without proof) a simple lemma that is very useful in the study of the spectral gap.

Lemma 5.4.

$$1 - \lambda_2 = \min\left\{ \sum_{ij \in E(G)} (x_i - x_j)^2 \ :\ \sum_i \pi(i) x_i = 0,\ \sum_i \pi(i) x_i^2 = \frac{1}{2m} \right\}$$

(each edge ij is considered only once in the sum).

Proof. Proof of Theorem 5.3 To warm up, let us prove the upper bound first. By Lemma 5.4, it suffices to exhibit a vector $x \in \mathbb{R}^V$ such that

$$\sum_i \pi(i)x_i = 0, \qquad \sum_i \pi(i)x_i^2 = 1/(2m), \tag{5.1}$$

and

$$\sum_{ij \in E(G)} (x_i - x_j)^2 = \Phi. \tag{5.2}$$

Let S be a set with minimum conductance, and consider a vector of the type

$$x_i = \begin{cases} a, & \text{if } i \in S, \\ b, & \text{if } i \in V \setminus S. \end{cases}$$

Such a vector satisfies (5.1) if

$$a = \sqrt{\frac{\pi(V \setminus S)}{2m\pi(S)}}, \qquad b = -\sqrt{\frac{\pi(S)}{2m\pi(V \setminus S)}},$$

and then straightforward substitution shows that (5.2) is also satisfied.

To prove the lower bound, we again invoke Lemma 5.4: we prove that for every vector $x \in \mathbb{R}^V$ satisfying (5.1), we have

$$\sum_{ij \in E(G)} (x_i - x_j)^2 \geq \frac{\Phi^2}{8}. \tag{5.3}$$

Conductance enters the picture through the following inequality, which is, in a sense, the "ℓ_1-version" of (5.3).

Lemma 5.5. *Let G be a graph with conductance Φ. Let $y \in \mathbb{R}^V$ and assume that $\pi(\{i : y_i > 0\}) \leq 1/2$, $\pi(\{i : y_i < 0\}) \leq 1/2$ and $\sum_i \pi(i)|y_i| = 1$. Then*

$$\sum_{(i,j) \in E} |y_i - y_j| \geq m\Phi.$$

Proof. Proof of the Lemma Label the points by $1, \ldots, n$ so that

$$y_1 \leq y_2 \leq y_t < 0 = y_{t+1} = \ldots = y_s < y_{s+1} \leq \ldots \leq y_n.$$

Set $S_i = \{1,\ldots,i\}$. Substituting $y_j - y_i = (y_{i+1} - y_i) + \cdots + (y_j - y_{j-1})$, we get

$$\sum_{(i,j)\in E} |y_i - y_j| = \sum_{i=1}^{n-1} |\nabla(S_i)|(y_{i+1} - y_i) \geq 2m\Phi \sum_{i=1}^{n-1}(y_{i+1} - y_i)\pi(S_i)\pi(V \setminus S_i).$$

Using that $\pi(S_i) \leq 1/2$ for $i \leq t$, $\pi(S_i) \geq 1/2$ for $i \geq s+1$, and that $y_{i+1} - y_i = 0$ for $t < i < s$, we obtain

$$\begin{aligned}\sum_{(i,j)\in E} |y_i - y_j| &\geq m\Phi \sum_{i=1}^{t}(y_{i+1} - y_i)\pi(S_i) + m\Phi \sum_{i=t+1}^{n-1} (y_{i+1} - y_i)\pi(V \setminus S_i)\\ &= m\Phi \sum_i \pi(i)|y_i| = m\Phi. \qquad \blacksquare\end{aligned}$$

Now we return to the proof of the theorem. Let x be any vector satisfying (5.1). We may assume that $x_1 \geq x_2 \geq \ldots \geq x_n$. Let k $(1 \leq k \leq n)$ be the index for which $\pi(\{1,\ldots,k-1\}) \leq 1/2$ and $\pi(\{k+1,\ldots,n\}) < 1/2$. Setting $z_i = \max\{0, x_i - x_k\}$ and choosing the sign of x appropriately, we may assume that

$$\begin{aligned}\sum_i \pi(i)z_i^2 &\geq \frac{1}{2}\sum_i \pi(i)(x_i - x_k)^2 = \frac{1}{2}\sum_i \pi(i)x_i^2 - x_k \sum_i \pi(i)x_i + \frac{1}{2}x_k^2\\ &= \frac{1}{2m} + \frac{1}{2}x_k^2 \geq \frac{1}{2m}.\end{aligned}$$

Now Lemma 5.5 can be applied to the numbers $y_i = z_i^2 / \sum_i \pi(i)z_i^2$, and we obtain that

$$\sum_{(i,j)\in E} |z_i^2 - z_j^2| \geq m\Phi \sum_i \pi(i)z_i^2.$$

On the other hand, using the Cauchy-Schwartz inequality,

$$\sum_{(i,j)\in E} |z_i^2 - z_j^2| \leq \left(\sum_{(i,j)\in E} (z_i - z_j)^2\right)^{1/2} \left(\sum_{(i,j)\in E} (z_i + z_j)^2\right)^{1/2}.$$

Here the second factor can be estimated as follows:

$$\sum_{(i,j)\in E} (z_i + z_j)^2 \leq 2 \sum_{(i,j)\in E} (z_i^2 + z_j^2) = 4m \sum_i \pi(i)z_i^2.$$

Combining these inequalities, we obtain

$$\begin{aligned}\sum_{(i,j)\in E}(z_i-z_j)^2 &\geq \left(\sum_{(i,j)\in E}|z_i^2-z_j^2|\right)^2 \Big/ \sum_{(i,j)\in E}(z_i+z_j)^2\\ &\geq \Phi^2 m^2\left(\sum_i \pi(i)z_i^2\right)^2 \Big/ 4m\sum_i \pi(i)z_i^2\\ &= \frac{\Phi^2 m}{4}\sum_i \pi(i)z_i^2 \geq \frac{\Phi^2}{8}.\end{aligned}$$

Since trivially

$$\sum_{(i,j)\in E}(x_i-x_j)^2 \geq \sum_{(i,j)\in E}(z_i-z_j)^2,$$

the theorem follows. ∎

Corollary 5.6. *For any starting node i, any node j and any $t \geq 0$,*

$$\left|P^t(j)-\pi(j)\right| \leq \sqrt{\frac{d(j)}{d(i)}}\left(1-\frac{\Phi^2}{8}\right)^t.$$

In another direction, Chung and Yau [17] considered a refined notion of conductance, replacing $\pi(S)\pi(V\setminus S)$ in the denominator by some power of it, and showed how this relates to higher eigenvalues. Diaconis and Saloff-Coste [23] used similar inequalities to get improved bounds on the mixing time, in particular on the early part when the distribution is highly concentrated. Theorem 5.3 is a discrete analogue of Cheeger's inequality from differential geometry, and these inequalities are discrete analogues of the Harnack, Sobolev and Nash inequalities known from the theory of the heat equation, and in fact, these results represent first steps in the exciting area of studying "difference equations" on graphs as discrete analogues of differential equations.

Conductance and multicommodity flows

Conductance itself is not an easy parameter to handle; it is NP-hard to determine it even for an explicitly given graph. But there are some methods to obtain good estimates. The most important such method is the construction of *multicommodity flows.* Let us illustrate this by a result of Babai and Szegedy [11].

Theorem 5.7. *Let G be a connected graph with a node-transitive automorphism group, with diameter D. Then the conductance of G is at least $1/(dD)$. If the graph is edge-transitive, its conductance is at least $1/D$.*

Proof. For each pair i, j of points, select a shortest path P_{ij} connecting them. Let $\mathcal{P}$ denote the family of these paths and all their images under automorphisms of G. The total number of paths in $\mathcal{P}$ (conting multiplicities) is $\binom{n}{2}g$, where g is the number of automorphisms of G. Moreover, $\mathcal{P}$ contains exactly g paths connecting any given pair of points.

We claim that every edge occurs in at most $Dg(n-1)$ paths of $\mathcal{P}$. In fact, if an edge e occurs in p paths then so does every image of e under the automorphisms, and there are at least $n/2$ distinct images by node-transitivity. This gives $pn/2$ edges, but the total number of edges of paths in $\mathcal{P}$ is at most $Dg\binom{n}{2}$, which proves the claim.

Now let $S \subseteq V(G)$, $|S| = s \le |V(G)|/2$. The number of paths in $\mathcal{P}$ connecting S to $V(G) \setminus S$ is exactly $gs(n-s)$. On the other hand, this number is at most $|\nabla(S)| \cdot Dg(n-1)$, and hence

$$|\nabla(S)| \ge \frac{gs(n-s)}{Dg(n-1)} = \frac{s}{D} \cdot \frac{n-s}{n-1}.$$

Hence the conductance of S is

$$\frac{n|\nabla(S)|}{ds(n-s)} \ge \frac{n}{n-1}\frac{1}{dD} > \frac{1}{dD}.$$

This proves the first assertion. The second follows by a similar argument.

Let us use Theorem 5.7 to estimate the mixing rate of C_n^k (where n is odd). This graph has an edge-transitive automorphism group, and its diameter is $k(n-1)/2$. Hence its conductance is more than $2/(kn)$, and so its mixing rate is at most

$$1 - \frac{1}{2k^2n^2}$$

We see that the bound is worse than the coupling and eigenvalue bounds; in fact, depending on the relative value of n and k, the mixing rate may be close to either the upper or the lower bound in Theorem 5.3. ∎

If we look in the proof of Theorem 5.7 at all paths connecting a given pair $\{u, v\}$ of nodes, and take each such path with weight $1/n^2g$, we get a flow from u to v with value $1/n^2$. The little argument given shows that these $\binom{n}{2}$ flows load each edge with at most $D(n-1)/n^2$. The rest of the argument applies to any graph and shows the following:

Proposition 5.8. *If we can construct in G a flow f_{uv} of value $\pi(u)\pi(v)$ from u to v for each $u \neq v$, and the maximum total load of these $\binom{n}{2}$ flows on any edge is at most γ, then the conductance of G is at least $1/(2m\gamma)$.*

This proof method has many applications (Jerrum and Sinclair [36], Diaconis and Stroock [21], Fill [34]). But what are its limitations, i.e., how close can we get to the true conductance? An important theorem of Leighton and Rao [47] shows that we never lose more than a factor of $O(\log n)$.

Theorem 5.9. *Let G be a graph with conductance Φ. Then there exists a system of flows f_{uv} of value $\pi(u)\pi(v)$ from u to v for each $u \neq v$, loading any edge by at most $O(\log n)/m\Phi$.*

There are many refinements and extensions of this fundamental result (see e.g. Klein, Agraval, Ravi and Rao [45]; Leighton et al [46]), but these focus on multicommodity flows and not on conductance, so we do not discuss them here.

Shortcuts

In the last paragraphs we have sketched the following steps in estimating the mixing rate:

mixing rate $\rightarrow$ eigenvalue gap $\rightarrow$ conductance $\rightarrow$ multicommodity flows.

It is possible to make "shortcuts" here, thereby obtaining bounds that are often sharper and more flexible.

Diaconis and Stroock [21] and Fill [34] prove the following lower bound on the eigenvalue gap, shortcutting the notion of conductance. We define the *cost* of a flow f as $\sum_e f(e)$.

Theorem 5.10. *Assume that there exists a flow f_{uv} of value $\pi(u)\pi(v)$ from u to v for each $u \neq v$, such that the maximum total load of these $\binom{n}{2}$ flows on any edge is at most γ, and the cost of each flow f_{uv} is at most $\beta\pi(u)\pi(v)$. Then*

$$\lambda_2 \leq 1 - \frac{1}{2m\beta\gamma}.$$

Lovász and Simonovits [50,51] introduced a method that estimates the mixing rate directly using conductance or related parameters, without the use of eigenvalue techniques. This makes the method more flexible. We formulate one application that is implicit in these papers:

Theorem 5.11. *Let $t \in \mathbb{Z}_+$ and assume that for each $0 \le s \le t$ and $0 \le x \le 1$, every level set $A = \{v \in V : P^s(v) \ge x\}$ has conductance at least ψ. Then for every $S \subseteq V$,*

$$|P^t(S) - \pi(S)| \le \sqrt{|V|}\left(1 - \frac{\psi^2}{4}\right)^t.$$

In other words, if the convergence $P^t \to \pi$ is slow, then among the level sets of the P^t there is one with small conductance. Other applications of this method include results where sets S with "small" measure $\pi(S)$ are allowed to have small conductance.

6. Sampling by random walks

Probably the most important applications of random walks in algorithm design make use of the fact that (for connected, non-bipartite graphs) the distribution of v_t tends to the stationary distribution π as $t \to \infty$. In most (though not all) cases, G is regular of some degree d, so π is the uniform distribution. A node of the random walk after sufficiently many steps is therefore essentially uniformly distributed.

It is perhaps surprising that there is any need for a non-trivial way of generating an element from such a simple distribution as the uniform. But think of the first application of random walk techniques in real world, namely shuffling a deck of cards, as generating a random permutation of 52 elements from the uniform distribution over all permutations. The problem is that the set we want a random element from is exponentially large (with respect to the natural size of the problem). In many applications, it has in addition a complicated structure; say, we consider the set of lattice points in a convex body or the set of linear extensions of a partial order.

Enumeration and volume computation

The following general scheme for approximately solving enumeration problems, called the *product estimator,* is due to Jerrum, Valiant and Vazirani [39], and also to Babai [10] for the case of finding the size of a group. Let V be the set of elements we want to enumerate. The size of V is typically exponentially large in terms of the natural "size" k of the problem. Assume that we can find a chain of subsets $V_0 \subset V_1 \subset \dots V_m = V$ such that for each i,

(a) $|V_0|$ is known (usually $|V_0| = 1$);

(b) $|V_{i+1}|/|V_i|$ is polynomially bounded (in k);

(c) m is polynomially bounded;

(d) we have a subroutine to generate a random element uniformly distributed over V_i, for each $1 \le i \le m$.

Then we can estimate the ratios $|V_{i+1}|/|V_i|$ by generating a polynomial number of elements of V_{i+1} and counting how often we hit V_i. The product of these estimates and of $|V_0|$ gives an estimate for $|V|$. This scheme leads to a randomized polynomial time approximation algorithm (provided (a), (b), (c) and (d) are satisfied and the subroutine in (d) is polynomial).

The crucial issue is how to generate a random element of V_i in polynomial time. We discuss this question for $V_m = V$; in virtually all applications of the method, every V_i itself is of the same type as V, and so the same arguments apply (this phenomenon is called "self-reducibility").

As mentioned above, random walks provide a general scheme for this. We define a connected graph $G = (V, E)$ on which a random walk can be implemented, i.e., a random neighbor of a given node can be generated (most often, the nodes have small (polynomial) maximum degree). By adding loops, we can make the graph regular and non-bipartite. Then we know that if we stop after a large number of steps, the distribution of the last node is very close to uniform. Our results about the mixing rate tell us how long we have to follow the random walk; but to find good estimates of the mixing rate (on the spectral gap, or on the conductance) is usually the hard part.

This method for generating a random element from a combinatorial structure was initiated by Broder [14] for the problem of approximating the number of perfect matchings in a graph. A proof of the polynomiality of the method was given by Jerrum and Sinclair [36] for the case of graphs with minimum degree at least $n/2$. Whether the method can be modified to handle the case of sparse graph is an open problem.

Let us sketch this important result. Let G be a graph on n nodes with all degrees at least $n/2$. We want to generate a random perfect matching of a graph G on n nodes (n even), approximately uniformly. Therefore, we want to define a graph whose nodes are the perfect matchings, and do a random walk on this graph. However, there is no easy way to step from one perfect matching to another; therefore, we extend the set we consider and include also all *near-perfect* matchings (i.e., matchings with $n/2-1$ edges). We connect two near-perfect matchings by an edge if they have $n/2-2$ edges in common, and connect a perfect matching to all near-perfect matchings contained in it, to obtain a graph H. The degrees in H are bounded by $3n$; we add loops at the nodes to make H regular of degree $3n$.

Now one can construct a multicommodity flow (basically following the tranformation of one matching to the other by alternating paths) to show that $1/\Phi(H)$ is polynomially bounded in n. Hence we can generate an essentially uniformly distributed random node of H by walking a polynomial number of steps. If this node corresponds to a perfect matching, we stop. Else, we start again. The assumption about the degrees can be used to show that the number of perfect matchings is at least a polynomial fraction of the number of near-perfect matchings, and hence the expected number of iterations before a perfect matching is obtained is polynomially bounded.

Other applications of this method involve counting the number of linear extensions of a partial order (Khachiyan and Karzanov [41]), eulerian orientations of a graph (Mihail and Winkler [59]), forests in dense graphs (Annan [7]), and certain partition functions in statistical mechanics (Jerrum and Sinclair [37]). See Welsh [66] for a detailed account of fully polynomial randomized approximation schemes for enumeration problems.

As another example, consider the fundamental problem of finding the volume of a convex body. The exact computation of the volume is difficult, which can be stated, and in some sense proved, in a mathematically exact way. Dyer and Frieze [26] and Khachiyan [43] proved that computing the volume of an n-dimensional convex polytope is #P-hard. Other results by Elekes [29] and Bárány and Füredi [12] show that for general convex bodies (given by, say, a separation oracle; see (Grötschel, Lovász and Schrijver [35]) for background information on the complexity of geometric algorithms) even to compute an estimate with bounded relative error takes exponential time, and the relative error of any polynomial time computable estimate grows exponentially with the dimension.

It was a breakthrough in the opposite direction when Dyer, Frieze and Kannan [27] designed a *randomized* polynomial time algorithm (i.e., an algorithm making use of a random number generator) which computes an estimate of the volume such that the probability that the relative error is larger than any prescribed positive number is arbitrarily small. Randomization reduces the relative error of a polynomial time approximation algorithm from exponentially large to arbitrarily small!

Several improvements of the original algorithm followed; here are some contributions and their running time estimates (we count the number of calls on the separation oracle; the $*$ after the O means that we suppress factors of $\log n$, as well as factors depending on the error bounds): Dyer, Frieze and Kannan [27] $O^*(n^{27})$, Lovász and Simonovits [50] $O^*(n^{16})$, Applegate and Kannan [8] $O^*(n^{10})$, Lovász [49] $O^*(n^{10})$, Dyer and Frieze [28] $O^*(n^8)$, Lovász and Simonovits [52] $O^*(n^7)$, Kannan, Lovász and Simonovits [40] $O^*(n^5)$.

Here is the general idea. Let K be a convex body in $\mathbb{R}^n$. Using known techniques from optimization, we may assume that K contains the unit ball and is contained in a ball with radius $R \le n^{3/2}$. Let K_i be the intersection of K and the ball about 0 with radius $2^{i/n}$ ($i = 0, 1, \ldots, m = \lceil 2n \log n \rceil$). Then $K_0 \subseteq K_1 \subseteq \ldots \subseteq K_m = K$, $\mathrm{vol}(K_{i+1})/\mathrm{vol}(K_i) \le 2$, and $\mathrm{vol}(K_0)$ is known. Thus the general scheme for enumeration described above can be adapted, provided we know how to generate a random point uniformly distributed in a convex body.

For this, we use random walk techniques. There is some technical difficulty here, since the set of points in a convex body is infinite. One can either consider a sufficiently fine grid and generate a random gridpoint in K, or extend the notions and methods discussed above to the case of an infinite underlying set. Both options are viable; the second takes more work but leads to geometrically clearer arguments about mixing rates.

We define the random walk as follows. The first point is generated uniformly from the unit ball B. Given v_t, we generate a random point u uniformly from the ball $v_t + B'$ with center v_t and radius δ (here the parameter δ depends on the version of the algorithm, but typically it is about $\varepsilon/\sqrt{n}$ with some small positive constant ε; $B' = \delta B$). If $u \in K$ then we let $v_{t+1} = u$; else, we generate a new point u and try again. This procedure corresponds to the random walk on the graph whose vertex set is K, with two points $x, y \in K$ connected by an edge iff $|x - y| \le \delta$.

The stationary distribution of this random walk is not the uniform distribution, but a distribution whose density function is proportional to the "degrees" $\ell(x) = \mathrm{vol}(K \cap (x + B'))/\mathrm{vol}(B')$. This quantity $\ell(x)$ is also called the "local conductance" at x; it is the probability that we can make a move after a single trial. If the stepsize is sufficiently small then this quantity, however, is constant on most of K, and the error committed is negligible.

(In several versions of the algorithm, the graph is padded with "loops" to make it regular. More exactly this means that if u is chosen uniformly from $v_t + B'$ and $u \notin K$, then we set $v_{t+1} = v_t$. So the two random walks produce the same set of points, but in one, repetition is also counted. It turns out that for the description as given above, the conductance can be estimated in a very elegant way as in Theorem 6.2 below, while in the other version, points with small local conductance cause a lot of headache.)

Putting these together, we have the outline of the volume algorithm. The analysis of it is, however, not quite easy. The main part of the analysis is the estimation of the conductance of the random walk in K. The proof of the following theorem involves substantial geometric arguments, in particular isoperimetric inequalities.

Theorem 6.2. *The conductance of the random walk in a convex body K with diameter D is at least* $\mathrm{const} \cdot \delta/(\sqrt{n}D)$.

This implies that it takes only $O^*(nR^2/\delta^2)$ steps to generate a random point in K.

This theorem suggests that one should choose the stepsize as large as possible. In fact, choosing $\delta = R$ would give us a random point in K in a single step! The problem is that if δ is large, we have to make too many trials before we can move to the next point. It is easy to calculate that in a stationary walk, the average "waiting time", i.e., the average number of points u to generate before we get one in K is

$$\mathrm{vol}(K) \Big/ \int_K \ell(x)\,dx$$

One can prove that this quantity is bounded from above by $1/(1 - \delta\sqrt{n})$, and hence it is $O(1)$ if δ is chosen less than $1/(2\sqrt{n})$. This means that the number of unsuccessful trials is only a constant factor more than that the number of steps in the random walk, which is $O^*(R^2n^2)$ for this choice of the stepsize.

The issue of achieving an R that is as small as possible is crucial but does not belong to this survey. With somewhat elaborate tricks, we can achieve $R = O(\sqrt{n})$ and hence the cost of generating a random point in K is $O^*(n^3)$. One has to generate $O^*(n)$ points to estimate each ratio $\mathrm{vol}(K_i)/\mathrm{vol}(K_{i+1})$ with sufficient accuracy, and there are $O^*(n)$ such ratios. This gives the total of $O^*(n^5)$ steps (oracle calls).

In virtually all applications of this method, the key issue is to estimate the conductance of the appropriate graph. This is usually a hard problem, and there are many unsolved problems. For example, is the conductance of a "matroid basis graph" polynomially bounded from below? (A matroid basis graph has all bases of a matroid $(E, \mathcal{M})$ as nodes, two being connected iff their symmetric difference has cardinality 2.) This is proved for graphic matroids (Aldous [2], Broder [15], cf. the proof of Theorem 6.6), and for a larger class of matroids called *balanced* (Mihail and Feder [30]). It is interesting to note that the property of graphic matroids that allows this proof to go through is inequality (4.1) for the number of spanning trees.

Metropolis filter

In many applications of random walks, the distribution we want to generate a random element from is not uniform. For example, a randomized optimization algorithm may be considered as a method of generating a random feasible solution from some probability distribution Q that is heavily concentrated on optimal and near-optimal solutions. To be more specific, let $f : V \to \mathbb{R}_+$ be the objective function; then maximizing f over V is just the extreme case when we want to generate a random element from a distribution concentrated on the set of optimum solutions. If, instead, we generate a random point w from the distibution Q in which $Q(v)$ is proportional to (say) $\exp(f(v)/T)$, where T is a very small positive number, then with large probability w will maximize f.

The elegant method of *random walk with Metropolis filter* (Metropolis, Rosenbluth, Rosenbluth, Teller and Teller [58]) describes a simple way to modify the random walk, so that it converges to an arbitrary prescribed probability distribution.

Let $G = (V, E)$ be a graph; for simplicity, assume that G is d-regular. Let $F : V \to \mathbb{R}_+$, and let v_0 be any starting point for the random walk. Let v_t be the node where we are after t steps. We choose a random neighbor u of v_t. If $F(u) \geq F(v_t)$ then we move to u; else, we flip a biased coin and move to u only with probability $F(u)/F(v_t)$, and stay at v with probability $1 - F(u)/F(v_t)$.

It is clear that this modified random walk is again a Markov chain; in fact, it is easy to check that it is also time-reversible (and so it can be considered as a random walk on a graph with edge-weights). The "miraculous" property of it is the following:

Theorem 6.3. *The stationary distribution Q_F of the random walk on a graph G filtered by a function F is given by the formula*

$$Q_F(v) = \frac{F(v)}{\sum_{w\in V} F(w)}.$$

An additional important property of this algorithm is that in order to carry it out, we do not even have to compute the probabilities $Q_F(v)$; it suffices to be able to compute the ratios $F(u)/F(v_t) = Q_F(u)/Q_F(v_t)$. This property of the Metropolis filter is fundamental in some of its applications.

Unfortunately, techniques to estimate the mixing time (or the conductance) of a Metropolis-filtered walk are not general enough, and not too many succesful examples are known. One notable exception is the work of Applegate and Kannan [8], who proved that random walks on the lattice points in a convex body, filtered by a smooth log-concave function, mix essentially as fast as the corresponding unfiltered walk. They applied this technique to volume computation. Diaconis and Hanlon [22] extended certain eigenvalue techniques to walks on highly symmetric graphs, filtered by a function which is "smooth" and "log-concave" in some sense. Some negative results are also known (Jerrum [38]).

Exact stopping rules

Let us start with the following funny fact.

Fact 6.4. *Let G be a circuit of length n and u any starting node. Then the probability that a random walk starting at u visits every node before hitting v is the same for each $v \neq u$.*

Clearly, if we replace the circuit with the complete graph, we get a similar result. Answering a question of Graham, it was proved by Lovász and Winkler [53] that no other graph has such a property. This follows from the next result, which verifies in a sense the intuition that the last node visited is more likely to be "far" than "near". Let $p(u, v)$ denote the probability that a random walk starting at u visits every node before v.

Theorem 6.5. *If u and v are two non-adjacent nodes of a connected graph G and $\{u, v\}$ is not a cutset, then there is a neighbor w of u such that $p(w, v) < p(u, v)$.*

Consequently, if G is e.g. 3-connected, then for each v, the nodes u for which $p(u, v)$ is minimal are neighbors of v.

As another result leading up the question of "exact stopping rules", let us describe a method due to Aldous [2] and Broder [15], generating a random spanning tree in a graph, so that each spanning tree is returned with *exactly* the same probability.

Theorem 6.6. *Consider a random walk on a graph G starting at node u, and mark, for each node different from u, the edge through which the node was first entered. Let T denote the set of marked edges. With probability 1, T is a spanning tree, and every spanning tree occurs with the same probability.*

Of course, only the second assertion needs proof, but this is not quite trivial. Our discussion below contains a proof based on a certain coupling idea; for a more direct proof, see Lovász [48], problem 11.58 (or work it out yourself!)

Consider a spanning tree T with root u, and draw a (directed) edge to each spanning tree T' with root v if $uv \in E(G)$ and T' arises from T by deleting the first edge on the path from v to u and adding the edge uv. Let H denote the resulting digraph. Clearly each tree with root v has indegree and outdegree $d(v)$ in H, and hence in the stationary distribution of a random walk on H, the probability of a spanning tree with a given root is proportional to the degree of the root (in G). If we draw a spanning tree from this distribution, and then forget about the root, we get every spanning tree with the same probability.

Now observe that a random walk on G induces a random walk on H as follows. Assume that we are at a node v of G, and at a node (T, v) in H, where T is a spanning tree. If we move along an edge vw in G, then we can move to a node (T', w) in H by removing the first edge of the path from w to v and adding the edge vw to the current spanning tree.

Also observe that by the time the random walk in G has visited all nodes (or at any time thereafter), the current spanning tree in H will be the tree formed by the last exits from each node, and the root is the last node visited. To relate this procedure to Theorem 6.6, let us consider the random walk on G for N steps (where N is much larger than the cover time of G. Viewing this backward is also a legal random walk on G, since G is undirected. If we follow that corresponding random walk on H, then it ends up with a rooted tree (T, v_N), which is the tree of first entries for this reverse walk, unless not all nodes of G were visited during the N returns to

v_0. Letting $N \to \infty$, the probability of this exception tends to 0, and the distribution of (T, v_N) tends to the stationary distribution on H which, for fixed v_N, is uniform on spanning trees. This proves Theorem 6.6.

Looking at this proof, it is natural to ask: can we get rid of the small error arising from the possibility that not all nodes are visited during N steps? After all, this is easily recognized, so perhaps in these cases we should walk a bit longer. More generally, given a random walk on a graph (or a Markov chain), can we define a "stopping rule", i.e., a function that assigns to every walk on the graph (starting at a given node u) either "STOP" or "GO", so that (a) with probability 1, every random walk is stopped eventually and (b) the distribution of the node where the random walk is stopped is the stationary distribution. We also consider randomized stopping rules, where coins may be flipped to determine whether we should stop.

Our first example above shows that for circuits and complete graphs, the "last node visited" rule provides an answer to the problem (we have to modify it a bit if we want to include the starting node too). In the case of the second example, we want to make the stopping time N dependent on the history: we only want to stop after we have seen all nodes of the graph G, but also want to maintain that the walk backward from the last node could be considered a random walk. Such a rule can be devised with some work (we omit its details). In what follows, we give some general considerations about this problem.

Of course, one has to be careful and avoid trivial rules like generating a node v from the stationary distribution, and then stopping when we first visit v. I don't know of any clean-cut condition to rule out such trivial solutions, but one should aim at rules that don't use global computations, in particular, don't make use of an *a priori* knowledge of the stationary distribution.

Stopping rules exist for quite general Markov chains. Asmussen, Glynn and Thorisson [9] describe a randomized algorithm that generates an element from the stationary distribution of a finite irreducible Markov chain, which needs only the number of states and a "black box" that accepts a state as an input and then simulates a step from this state. Lovász and Winkler [54] have found a randomized stopping rule that generates an element from the stationary distribution of any irreducible Markov chain, and only needs to know the number of states. This rule can be made deterministic under the assumption that the chain is aperiodic.

To indicate the flavor of the result, let us describe the case when the Markov chain has two states. The general case follows by a (not quite trivial) recursive construction (similarly as in the work of Asmussen, Glynn and Thorisson [9]).

So let

$$v_0, v_1, v_2, \dots \tag{6.1}$$

be an irreducible aperiodic Markov chain on states $\{u, v\}$. Irreducible means that the transition probabilities p_{uv}, p_{vu} are positive; aperiodocity means that at least one of p_{uu} and p_{vv} is also positive. It is easy to check that the stationary distribution is given by

$$\pi(u) = \frac{p_{vu}}{p_{uv} + p_{vu}}, \qquad \pi(v) = \frac{p_{uv}}{p_{uv} + p_{vu}},$$

The following randomized stopping rule generates a random element from π, without knowing any value p_{ij} or $\pi(i)$, only looking at the sequence (6.1):

Rule 1. *Flip a coin. If the result is head, let $i = 0$; else, let i be the first index for which $v_i \neq v_0$. If $v_{i+1} \neq v_i$ then output v_{i+1}; else, discard the first $i+1$ elements and repeat.*

If you don't like that we use coin flipping, you can use the Markov chain itself to simulate it, making the rule entirely deterministic.

Rule 2. *Wait for the first pair $i < j$ with the following properties:* (i) $v_j = v_i$, (ii) $v_{j+1} \neq v_{i+1}$, (iii) $v_{j+2} \neq v_{j+1}$, *and moreover,* (iv) *the state v_i occurs an even number of times before v_i and* (v) *not at all between v_i and v_j. Output v_{j+2}.*

If this sounds mysterious, note that for each of the first, second, etc. occurence of a pair of indices with (i), (ii), (iv) and (v), v_{j+1} can be either of the states with probability $1/2$.

The stopping rule sketched above takes a lot of time; we don't even know how to make the expected number of steps of the random walk polynomial in the maximum access time, let alone comparable with the mixing time (that we know may be logarithmic in n). On the other hand, if we allow global computation, we can get a stopping rule which needs, on the average, at most twice as many steps as the mixing time τ. We follow the random walk for τ steps, then "flip a biased coin"; with probability $\pi(v_\tau)/2P_\tau(v_\tau)$, we stop; with probability $1 - \pi(v_\tau)/2P_\tau(v_\tau)$, we forget about the past and start from v_τ a random walk of length τ etc. It is easy to see that the probability that we stop at v after k rounds is $2^{-k}\pi(v)$, which adds up to $\pi(v)$. Also, the expected number of steps is 2τ.

A *threshold rule* is a (relatively) simple kind of stopping rule. It is specified by a function $t : V \to \mathbb{R}_+$, depending on the staring point v_0, and works as follows:

if $t(v_k) \le k$, then stop;

if $t(v_k) \ge k+1$, go on;

if $k < t(v_k) < k+1$ then "flip a biased coin" and move with probability $t(v_k) - k$ but stop with probability $k + 1 - t(v_k)$.

Lovász and Winkler [55] have shown that there is a function t that gives a threshold rule that is optimal among all stopping rules in a very strong sense: it minimizes the expected number of steps among all randomized stopping rules (for a fixed starting node). It also minimizes the expected number of times any given node is visited. Every threshold rule is of course finite, in the sense that there is a finite time T such that it is guaranteed to stop within T steps (in fact, $T \le \max_i t(i)$). The optimal threshold rule minimizes this bound among all finite rules.

The expected number of steps for the optimal threshold rule, starting at node v, is

$$\tau^* = \max_u H(u, v) - \sum_u \pi(u)H(u, v).$$

It follows from the description of the stopping rule using the mixing time that

$$\tau^* \le 2\tau.$$

Since the definition of the mixing time τ has an arbitrarily chosen constant $1/2$ in it, while the definition of τ^* is "canonical", it should be more natural to call the quantity τ^* the mixing time.

Since this optimal stopping rule has many nice properties, it would be good to have an efficient implementation. The threshold function is polynomially computable; but this is not good enough since we want to apply these rules to exponentially large graphs. However, one can describe simple, easily implementable stopping rules with comparable expected time that achieve approximate mixing on the exponentially large graphs of interest discussed above.

References

[1] D. J. Aldous, Lower bounds for covering times for reversible Markov chains and random walks on graphs, *J. Theoretical Probability* **2**(1989), 91–100.

[2] D. J. Aldous, The random walk construction for spanning trees and uniform labelled trees, *SIAM J. Discrete Math.* **3**(1990), 450–465.

[3] D. J. Aldous and J. A. Fill, *Reversible Markov Chains and Random Walks on Graphs* (book in preparation).

[4] R. Aleliunas, R. M. Karp, R. J. Lipton, L. Lovász, C. W. Rackoff, Random walks, universal travelling sequences, and the complexity of maze problems, in: *Proc. 20th Ann. Symp. on Foundations of Computer Science* (1979), 218–223.

[5] N. Alon, Eigenvalues and expanders, *Combinatorica* **6**(1986), 83–96.

[6] N. Alon and V. D. Milman, λ_1, isoperimetric inequalities for graphs and superconcentrators, *J. Combinatorial Theory B* **38**(1985), 73–88.

[7] J. D. Annan, A randomised approximation algorithm for counting the number of forests in dense graphs, *Combin. Probab. Comput.* **3**(1994) 273–283.

[8] D. Applegate and R. Kannan, Sampling and integration of near log-concave functions, *Proc. 23th ACM STOC* (1990), 156–163.

[9] S. Asmussen, P. W. Glynn and H. Thorisson, Stationary detection in the initial transient problem, *ACM Transactions on Modeling and Computer Simulation* **2**(1992), 130–157.

[10] L. Babai, Monte Carlo algorithms in graph isomorphism testing, Université de Montréal Tech. Rep. DMS 79-10, 1979 (42).

[11] L. Babai and M. Szegedy, Local expansion of symmetrical graphs, *Combinatorics, Probability, and Computing* **1**(1992), 1–11.

[12] I. Bárány and Z. Füredi, Computing the volume is difficult, *Proc. of the 18th Annual ACM Symposium on Theory of Computing* (1986), 442–447.

[13] G. Brightwell and P. Winkler, Maximum hitting time for random walks on graphs, *J. Random Structures and Algorithms* **1**(1990), 263–276.

[14] A. Broder, How hard is it to marry at random? (On the approximation of the permanent), *Proc. 18th Annual ACM Symposium on Theory of Computing* (1986), 50–58.

[15] A. Broder, Generating random spanning trees, *Proc. 30th Annual Symp. on Found. of Computer Science* IEEE Computer Soc., (1989), 442–447.

[16] A. K. Chandra, P. Raghavan, W. L. Ruzzo, R. Smolensky and P. Tiwari, The electrical resistance of a graph captures its commute and cover times, *Proc. 21st ACM STOC,* (1989), 574–586.

[17] F. R. K. Chung and S. T. Yau, Eigenvalues of graphs and Sobolev inequalities, (1993) preprint.

[18] D. Coppersmith, U. Feige, and J. Shearer, Random Walks on Regular and Irregular Graphs, *Technical report CS93-15 of the Weizmann Institute* 1993.

[19] D. Coppersmith, P. Tetali and P. Winkler, Collisions among random walks on a graph, *SIAM J. Discr. Math.* **6**(1993), 363–374.

[20] P. Diaconis, *Group Representations in Probability and Statistics,* Inst. of Math. Statistics, Hayward, Californis, 1988.

[21] P. Diaconis and D. Stroock, Geometric bounds for eigenvalues of Markov chains, *Annals of Appl. Prob.* **1**(1991), 36–62.

[22] P. Diaconis and P. Hanlon, Eigen analysis for some examples of the Metropolis algorithm, *Hypergeometric functions on domains of positivity, Jack polynomials, and applications,* Contemporary Math. **138**(1992) Amer. Math. Soc., Providence, RI

[23] P. Diaconis and L. Saloff-Coste, Comparison theorems for random walk on finite groups, *Ann. Prob.* **21**(1993), 2131–2156.

[24] J. Dodziuk and W. S. Kendall, Combinatorial Laplacians and isoperimetric inequality, in: *From Local Times to Global Geometry, Control and Physics* (ed.: K. D. Ellworthy), Pitman Res. Notes in Math. Series **150**(1986), 68–74.

[25] P. G. Doyle and J. L. Snell, *Random walks and Electric Networks,* MAA, 1984.

[26] M. Dyer and A. Frieze, On the complexity of computing the volume of a polytope. *SIAM J. Comp.* **17**(1988), 967–974.

[27] M. Dyer, A. Frieze and R. Kannan, A Random Polynomial Time Algorithm for Approximating the Volume of Convex Bodies, *Proc. of the 21st Annual ACM Symposium on Theory of Computing* (1989), 375–381.

[28] M. Dyer and A. Frieze, Computing the volume of convex bodies: a case where randomness provably helps, in: *Probabilistic Combinatorics and Its Applications* (ed.: B. Bollobás), Proceedings of Symposia in Applied Mathematics, **44**(1992), 123–170.

[29] G. Elekes, A geometric inequality and the complexity of computing volume, *Discrete and Computational Geometry* **1**(1986), 289–292.

[30] T. Feder and M. Mihail, Balanced matroids, *Proc. 24rd ACM Symp. on Theory of Comp.* (1992), 26–38.

[31] U. Feige, A Tight Upper Bound on the Cover Time for Random Walks on Graphs, *Random Structures and Algorithms* **6**(1995), 51–54.

[32] U. Feige, A Tight Lower Bound on the Cover Time for Random Walks on Graphs, *Random Structures and Algorithms* **6**(1995), 433–438.

[33] U. Feige, Collecting Coupons on Trees, and the Analysis of Random Walks, *Technical report CS93-20 of the Weizmann Institute* 1993.

[34] J. A. Fill, Eigenvalue bounds on the convergence to stationarity for nonreversible Markov chains, with an application to the exclusion process, *The Annals of Appl. Prob.* **1**(1991), 62–87.

[35] M. Grötschel, L. Lovász and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization,* Springer-Verlag, 1988.

[36] M. R. Jerrum and A. Sinclair, Approximating the permanent, *SIAM J. Comput.* **18**(1989), 1149–1178.

[37] M. R. Jerrum and A. Sinclair, Polynomial-time approximation algorithms for the Ising model, *Proc. 17th ICALP, EATCS* (1990), 462–475.

[38] M. R. Jerrum, Large cliques elude the Metropolis process, *Random Structures and Algorithms* **3**(1992), 347–359.

[39] M. R. Jerrum, L. G. Valiant and V. V. Vazirani, Random generation of combinatorial structures from a uniform distribution, *Theoretical Computer Science* **43**(1986), 169–188.

[40] R. Kannan, L. Lovász and M. Simonovits, Random walks and a faster volume algorithm, (in preparation).

[41] A. Karzanov and L. G. Khachiyan, On the conductance of order Markov chains, *Order* **8**(1991), 7–15.

[42] J. Keilson, *Markov Chain Models – Rarity and Exponentiality,* Springer-Verlag, 1979.

[43] L. G. Khachiyan, The problem of computing the volume of polytopes is NP-hard, *Uspekhi Mat. Nauk 44*(1989), 199–200.

[44] L. G. Khachiyan, Complexity of polytope volume computation, in: *New Trends in Discrete and Computational Geometry* (ed.: J. Pach), Springer, (1993), 91–101.

[45] P. Klein, A. Agraval, R. Ravi and S. Rao, Approximation through multicommodity flow, *Proc. 31st Annual Symp. on Found. of Computer Science,* IEEE Computer Soc., (1990), 726–727.

[46] T. Leighton, F. Makedon, S. Plotkin, C. Stein, É. Tardos, and S. Tragoudas, Fast approximation algorithms for multicommodity flow problem, *Proc. 23rd ACM Symp. on Theory of Comp.* (1991), 101-111.

[47] F. T. Leighton and S. Rao, An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms, *Proc. 29th Annual Symp. on Found. of Computer Science,* IEEE Computer Soc., (1988), 422-431.

[48] L. Lovász, *Combinatorial Problems and Exercises,* Akadémiai Kiadó, Budapest – North Holland, Amsterdam, 1979, 1993.

[49] L. Lovász, How to compute the volume? *Jber. d. Dt. Math.-Verein, Jubiläumstagung 1990,* B. G. Teubner, Stuttgart, (1992), 138–151.

[50] L. Lovász and M. Simonovits, Mixing rate of Markov chains, an isoperimetric inequality, and computing the volume, *Proc. 31st Annual Symp. on Found. of Computer Science,* IEEE Computer Soc., (1990), 346–355.

[51] L. Lovász and M. Simonovits, On the randomized complexity of volume and diameter, *Proc. 33rd IEEE FOCS* (1992), 482–491.

[52] L. Lovász and M. Simonovits, Random walks in a convex body and an improved volume algorithm *Random Structures and Alg.* **4**(1993), 359–412.

[53] L. Lovász and P. Winkler, On the last new vertex visited by a random walk, *J. Graph Theory* **17**(1993), 593–596.

[54] L. Lovász and P. Winkler, Exact mixing in an unknown Markov chain, *The Electronic Journal of Combinatorics* **2**(1995), paper R15, 1–14.

[55] L. Lovász and P. Winkler, Efficient stopping rules for Markov chains, *Proc. 27th ACM Symp. Theory of Comput.* (1995), 76–82.

[56] P. Matthews, Covering problems for Brownian motion on spheres, *Ann. Prob.* **16** (1988), 189–199.

[57] P. Matthews, Some sample path properties of a random walk on the cube, *J. Theoretical Probability* **2**(1989), 129–146.

[58] N. Metropolis, A. Rosenblut, M. Rosenbluth, A. Teller amd E. Teller, Equation of state calculation by fast computing machines, *J. Chem. Physics* **21**(1953), 1087–1092.

[59] M. Mihail and P. Winkler, On the number of Eulerian orientations of a graph, Extended Abstract, *Proc. 3rd ACM-SIAM Symposium on Discrete Algorithms* (1992), 138–145; full version to appear in *Algorithmica.*

[60] C. St.J. A. Nash-Williams, Random walks and electric currents in networks, *Proc. Cambridge Phil. Soc.* **55**(1959), 181–194.

[61] G. Polya, Über eine Aufgabe der Wahrscheinlichkeitsrechnung betreffend die Irrfahrt im Strassennetz, *Math. Annalen* **84**(1921), 149–160.

[62] A. Sinclair and M. Jerrum, Conductance and the rapid mixing property for Markov chains: the approximation of the permanent resolved, *Proc. 20th ACM STOC,* (1988), 235–244.

[63] P. Tetali, Random walks and effective resistance of networks, *J. Theoretical Probability* **1**(1991), 101–109.

[64] P. Tetali and P. Winkler, Simultaneous reversible Markov chains, in: *Combinatorics, Paul Erdős is Eighty,* Vol. 1 (eds:. D. Miklós, V. T. Sós, T. Szőnyi), Bolyai Society Mathematical Studies, Budapest, **1**(1993), 422–452.

[65] C. Thomassen, Resistences and currents in infinite electrical networks, *J. Comb. Theory* **49**(1990), 87–102.

[66] D. J. A. Welsh, *Complexity: Knots, Colourings and Counting,* London Math. Soc. Lecture Note Series **186**, Cambridge University Press, 1993.

László Lovász

Department of Computer Science,
Yale University,
New Haven, CT 06520
USA

e-mail: lovasz@cs.yale.edu