1. Random walks on graphs

We call a simple random walk on a graph the Markov chain with state space the vertices of a graph and transition probability defined as:

\[ P_{x,y} = \begin{cases} \frac{1}{\deg(x)}, & \text{if } (x,y) \in E \\ 0, & \text{otherwise} \end{cases} \]

The chain is irreducible and aperiodic if and only if the graph is connected and non-bipartite. We can check that in this case the stationary distribution is \( \pi(v) = \frac{\deg(v)}{2m} \) for every \( v \in V \) and that the chain is reversible.

The cover time of a graph \( C(G) \) is the expected time to reach all vertices (the worst case over the starting vertex). Recall that for two vertices \( u, v \), the hitting time \( E_u(\tau_v) \) is the expected time to get from \( u \) to \( v \).

**Theorem 1.** For any \( (u, v) \in E \), \( E_u(\tau_v) + E_v(\tau_u) \leq 2m \).

**Theorem 2.** For a connected graph \( C(G) \leq 2m(n-1) \).

**Corollary 3.** There is a log-space \( O(n^3) \)-time randomized algorithm for st-connectivity.

2. Electric networks

If the graph is interpreted as an electric network, where on each edge there is a resistor of 1 ohm, for two vertices \( u, v \) the effective resistance between these vertices \( R(u \leftrightarrow v) \) (i.e. the voltage difference that has to be applied to \( u \) and \( v \) to get a unit current flow) is related to the escape probability.

**Theorem 4.**

\[ P_u \{ \tau_v < \tau_u^{+} \} = \frac{1}{\deg(u)R(u \leftrightarrow v)} \]

The proof uses the concept of harmonic function. A function \( h : \Omega \to \mathbb{R} \) is harmonic for \( P \) at a vertex \( x \in \Omega \) if \( h(x) = \sum_{y \in \Omega} P(x, y)h(y) \). For example, notice that the hitting time \( h_{x,a} \), from \( x \) to a fixed vertex \( a \in \Omega \) (or to a set \( A \subset \Omega \)), considered as a function of \( x \), is a harmonic function on \( \Omega \backslash \{a\} \) (or \( \Omega \backslash A \)). Another example is the voltage. Suppose a voltage difference of 1 is applied to vertices \( s \) and \( t \), then the resulting voltage on all of the vertices is a harmonic function on \( \Omega \backslash \{s, t\} \).

The important property of harmonic functions that we use is that if we fix the values of a function on a set \( B \subset \Omega \) it has a unique extension which is harmonic on \( \Omega \backslash B \). This is because the harmonicity constraints form a system of equations with \( |\Omega| - |B| \) linear equations and \( |\Omega| - |B| \) unknowns.

We also get an exact relationship between the effective resistance and the commute time.

**Theorem 5.** \( E_u(\tau_v) + E_v(\tau_u) = 2mR(u \leftrightarrow v) \).
This theorem can be used to give a simpler proof of Theorem 1. It also gives us a tighter bound for the cover time.

**Theorem 6.** Let \( R(G) = \max_{u,v \in V} R(u \leftrightarrow v) \). Then \( mR(G) \leq C(G) \leq mR(G) \times 2e^3 \ln n + n \).

### 3. Applications of random walks

We will look at local search algorithms for the 2-SAT and 3-SAT problems. The problem is to find a satisfying assignment for a logical formula in conjunctive normal form with clauses of 2 or 3 literals respectively.

For example,

\[ \phi(x_1, \ldots, x_n) = (x_2 \lor x_5) \land (x_1 \lor x_2) \land \ldots \]

Consider the following local search algorithm for 2-SAT. We start with an arbitrary assignment to the variables. If it’s a satisfying assignment, it is output, otherwise we find a clause which is not satisfied and flip the value of one of the variables in the clause uniformly at random. This is a random walk on the space of assignments to the variables which has size \( 2^n \).

**Theorem 7.** The local search algorithm for 2SAT finds a satisfying assignment, if one exists, with probability \( 1 - \frac{1}{2^b} \) within \( n^2 b \) steps.

To prove this we just need to consider the evolution of the random variable \( X \) indicating the Hamming distance between the current assignment and some fixed satisfying assignment. This turns out to be a walk on the line of length \( n \) with probability at least \( 1/2 \) of going to the left every time. This walk reaches 0 in expected time \( n^2 \) so within \( 2n^2 \) steps it reaches with probability at least \( 1/2 \) (which can be boosted by going \( bn^2 \) steps).

The same analysis applied to 3-SAT leads to a random walk on the line with probability of \( 1/3 \) of going left. This, together with the observation that with high probability the search starts with \( X \) close to \( n/2 \), gives an algorithm running in time \( O((\sqrt{3})^n) \), which is better than the trivial algorithm which takes \( 2^n \) steps.

We can improve the algorithm by letting the search run only \( 3n \) steps but restarting the search \( \Theta((\frac{4}{3})^n n^{O(1)}) \) times.

**Theorem 8.** The local search algorithm with restarts finds a satisfying assignment with probability \( 1 - \frac{1}{2^b} \) within time \( O((\frac{4}{3})^n n^{O(1)}) \).