Hitting Time Distributions of Random Walks on General Graphs, Vertex Transitive Graphs, and Groups

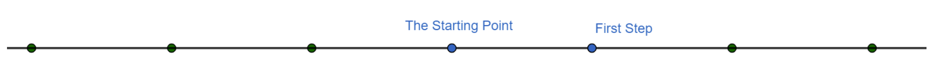
Anuraag Kumar

April 2025

Introduction

Random walks are a fundamental topic in probability theory with wide-ranging applications across physics, engineering, mathematics, and computer science. In fact, there are over 11,000 papers on arXiv that explore them in depth. Random walks can be used to model a variety of real-world phenomena, including fluid dynamics, stock price movements, genetic drift, animal foraging behavior, and search algorithms.

But what exactly is a random walk? At its core, it's a simple model of an indecisive traveler. To understand how such a versatile model works, let's look at its simplest form. Imagine standing at 0 on a number line. You flip a fair coin: if it lands heads, you take a step to the right; if tails, a step to the left. For instance, let's say you flip the coin and get heads—you move to position 1. That's the first step in your random walk.



and if we get tails after that...

A lot of research goes into modeling where we would end up after n steps and with what probability. This process can also be generalized to higher dimensions (moving on a grid or in a 3d space instead of just the number line where we move in each direction with the same probability) and those processes have their own interesting properties. For the purpose of this paper, our random walks will occur on graphs.

A graph is a mathematical structure made up of nodes, called vertices, and the connections between them, called edges. This can be conceived as a system of towns and the roads that connect them. Consider now a random walk on a graph. At each step, the choice of direction depends only on the current position and the available edges, not on the path taken to get there. Whether the edges are directed or undirected, weighted or unweighted, the behavior of the random walk changes accordingly, revealing insights into the structure and dynamics of the graph itself. Through this lens, a graph becomes not just a static diagram, but a playground of movement, chance, and connectivity.

Motivation and Problem Statement

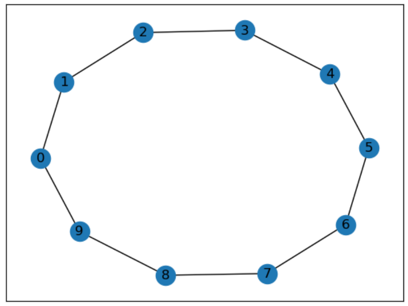
The central focus of this honors thesis is the *hitting time*, which refers to the number of steps required for a random walk to move from one vertex to another. In probabilistic terms, for a graph $G = (V, E)$, we define the hitting time as

$$\tau_{i,j} = \inf\{t \mid X_0 = i, X_t = j\}$$

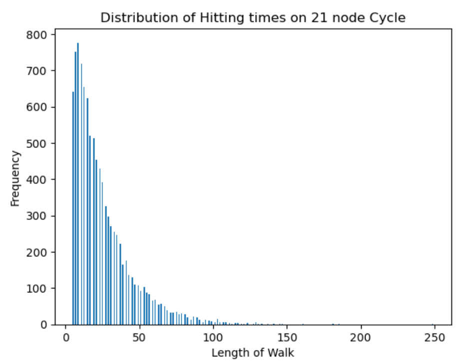
where X_t denotes the position of the random walker at time t, and i and j are two distinct vertices in the graph.

Studying hitting times in Markov chains is essential because they provide deep insights into the behavior and dynamics of stochastic processes over time. Specifically, a hitting time represents the expected number of steps it takes for a Markov chain to reach a particular state for the first time, starting from a given initial state. This concept is crucial across a variety of fields. In network theory, it helps us understand the expected time it takes for a random walk to reach a specific node. In reliability engineering, it can model the time until a system fails or recovers. In algorithm design, hitting times are important for evaluating the efficiency of randomized algorithms, while in economics and biology, they offer tools for modeling transitions between key states or events. By studying hitting times, we gain a clearer understanding of the temporal aspects of randomness, enabling us to predict, optimize, and control complex systems governed by uncertainty. Any system that can be modeled by a random walk benefits from the investigation of hitting times.

One effective way to explore hitting times is through simulation. For example, consider the cycle graph with 10 nodes. The following diagram shows the graph:

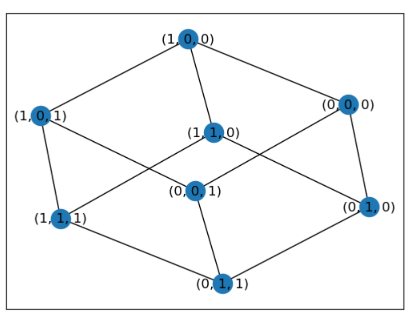


Suppose we are interested in the random variable $\tau_{0,5}$, representing the number of steps needed to go from node 0 to node 5. After running 10,000 trials, we obtain the following frequency graph:

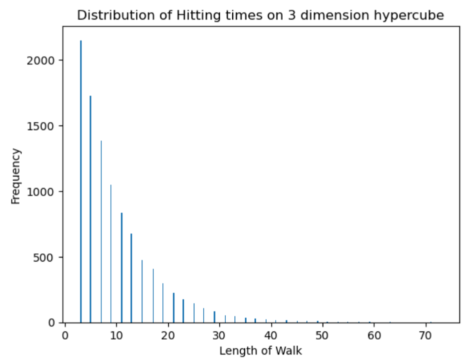


What stands out here is the *large variance* in the distribution. The sample mean is 25.0306, and the sample variance is 410, indicating that the variance is relatively large compared to the mean.

We can observe a similar behavior in the hypercube graph with 8 nodes, which is shown below:



Now, suppose we are interested in $\tau_{(0,0,0),(1,1,1)}$, the number of steps needed to move from node $(0,0,0)$ to node $(1,1,1)$. After running 10,000 trials, we obtain the following frequency graph:



The sample mean in this case is 10, and the sample variance is 63.

This observation leads to the central motivation for this thesis. In the literature, the primary focus in the study of hitting times has been on *expected hitting times*, denoted $E[\tau_{i,j}]$. However, as demonstrated above, the expected value alone is often a poor predictor of how the random walk behaves. In this paper, we will examine two related quantities: $P(\tau_{i,j} = n)$, the probability mass function of hitting times, and $\text{Var}(\tau_{i,j})$, the variance of the hitting time. Despite their importance, these quantities receive less attention due to computational challenges. Our approach will involve general computations, with a subsequent focus on vertex-transitive graphs.

Markov Chains on General Graphs

Many of the ideas from the next section comes from [2] and [3]

Distributions

Let's try to find the distribution of $P(\tau_{i,j} = n)$ on our graph G with Markov matrix A. One relationship becomes clear.

$$P(\tau_{i,j} = n) = \sum_{k \neq j} P(\tau_{i,k} = 1) P(\tau_{k,j} = n - 1)$$

The above formula calculates the probability of reaching an adjacent node to our ending node in $n-1$ steps and then making a step from that adjacent node to the end. By setting $k \neq j$, we make sure that we aren't adding the probability that we arrive to ending node j one move early. The above is true, as each step of a random walk is independent. Let us fix an ending node j, and then let us define a vector

$$P_n = \begin{bmatrix} P(\tau_{1,j} = n) \\ P(\tau_{2,j} = n) \\ \vdots \\ P(\tau_{|V|-1,j} = n) \end{bmatrix}$$

the nodes $1, 2...|V| - 1$ represent some arbitrary numbering of the nodes of the graph once j is removed. Let Q be the matrix such that $Q_{ik} = P(\tau_{ik} = 1)$ such that $i, k \neq j$. We then have

$$P_n = QP_{n-1}$$

as the recursion above is simply matrix multiplication. So then by induction, we have

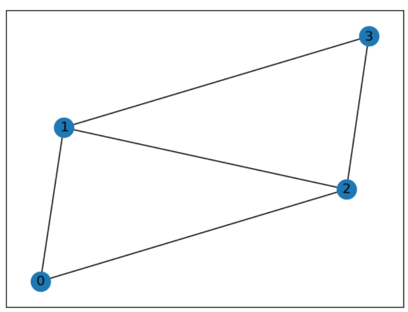
$$P_n = Q^{n-1}P_1$$

One might notice that Q is simply the Markov matrix of our graph but with removed jth row and jth column. Therefore, here we have a solid way of calculating distributions. As we are taking arbitrary powers of a matrix, it often comes down to diagonalizing Q. This is often hard to do by hand, but a computer can help. There are a couple things to note about the eigenvalues of such a matrix.

1. Q is a substochastic matrix as the sum of every row is less than or equal to 1. This implies that $|\lambda| < 1$. Where λ is an eigenvalue of 1.

2. Q is the adjacency graph of a subgraph of G. Which implies by the Interlacing theorem of Spectral Graph Theory that all the eigenvalues of Q are embedded between the eigenvalues of A.

Consider this graph



. We will use every step on a random walk is independent and assuming that walking across any edge is equally likely. Setting $j = 0$ and then Q is

$$\begin{bmatrix} 0 & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix}$$

and P_1 is

$$\begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \\ 0 \end{bmatrix}$$

. Then

$$Q^{n-1}P_1 = \begin{bmatrix} P(\tau_{1,0} = n) \\ P(\tau_{2,0} = n) \\ P(\tau_{3,0} = n) \end{bmatrix} = \begin{bmatrix} \frac{-(-1)^n*2^{(2*n)*3^n*\sqrt{13}+(2*\sqrt{13}+14)^n*\sqrt{13}}}{13*\left(\sqrt{13}+1\right)^n*2^n*3^n} \\ \frac{-(-1)^n*2^{(2*n)*3^n*\sqrt{13}+(2*\sqrt{13}+14)^n*\sqrt{13}}}{13*\left(\sqrt{13}+1\right)^n*2^n*3^n} \\ \frac{13*(-1)^n*2^{(2*n)*3^n+13*(2*\sqrt{13}+14)^n+(-1)^n*2^{(2*n)*3^n*\sqrt{13}-(2*\sqrt{13}+14)^n*\sqrt{13}}}}{26*\left(\sqrt{13}+1\right)^n*2^n*3^n} \end{bmatrix}$$

We can see that even for relatively simple looking graphs, the distributions can be very complicated and often intractable to compute by hand with larger graphs. Later, we will restrict the graphs we will work with to make sure this process is simpler.

Characteristic Function

Let's consider the value of the $\phi_{\tau_{i,j}}(t)$. We can compute this directly using standard techniques

$$\phi_{\tau_{i,j}}(t) = E[e^{it\tau_{i,j}}] = \sum_{n=-\infty}^{\infty} e^{int} P(\tau_{i,j} = n)$$

As it is impossible to move to another node in negative moves. Also since $i \neq j$, it follows that $n \neq 0$. Therefore, we can write the sum as

$$\sum_{n=1}^{\infty} e^{int} P(\tau_{i,j} = n)$$

$$\sum_{n=1}^{\infty} e^{int} e_i^T Q^{n-1} P_1$$

Instead of just considering just one moment generating function, we consider all the characteristic functions $\phi_{\tau_{i,j}}(t)$ for all $i \neq j$. We will call this vector function $M_{\tau_{*,j}}(t)$.

$$\phi_{\tau_{*,j}}(t) = \sum_{n=1}^{\infty} e^{int} Q^{n-1} P_1$$

$$\phi_{\tau_{*,j}}(t) = e^{it} \sum_{n=0}^{\infty} e^{int} Q^n P_1$$

$$\phi_{\tau_{*,j}}(t) = e^{it}(I - e^{it}Q)^{-1} P_1$$

We can take the derivative of this

$$\phi'_{\tau_{*,j}}(t) = ie^{it}(I - e^{it}Q)^{-1}P_1 + e^{it}(I - e^{it}Q)^{-1} ie^{it}Q(I - e^{it}Q)^{-1}P_1$$

$$\phi'_{\tau_{*,j}}(t) = ie^{it}(I - e^{it}Q)^{-1}(I + e^{it}Q(I - e^{it}Q)^{-1})P_1$$

$$\phi'_{\tau_{*,j}}(t) = ie^{it}(I - e^{it}Q)^{-1}((I - e^{it}Q + e^{it}Q)(I - e^{it}Q)^{-1})P_1$$

$$\phi'_{\tau_{*,j}}(t) = ie^{it}(I - e^{it}Q)^{-2}P_1$$

Since $iE[\tau_{*,j}] = \phi_{\tau_{*,j}}(0)$

$$\phi'_{\tau_{*,j}}(0) = i(I - Q)^{-2}P_1$$

Therefore, we have

$$E[\tau_{*,j}] = (I - Q)^{-2}P_1$$

We have another way of reaching this quantity by considering the quantity agian.

$$\phi_{\tau_{*,j}}(t) = e^{it}P_1$$

$$(I - e^{it}Q)\phi_{\tau_{*,j}}(t) = e^{it}P_1$$

We take the derivative of both sides

$$(I - e^{it}Q)\phi'_{\tau_{*,j}}(t) - ie^{it}Q\phi_{\tau_{*,j}}(t) = ie^{it}P_1$$

$$(I - e^{it}Q)\phi'_{\tau_{*,j}}(t) = ie^{it}P_1 + ie^{it}Q\phi_{\tau_{*,j}}(t)$$

We now plug in $t = 0$.

$$(I - Q)\phi'_{\tau_{*,j}}(0) = iP_1 + iQ\phi_{\tau_{*,j}}(0)$$

Since $\phi_{\tau_{i,j}}(0) = \sum_{n=1}^{\infty} e^{i0t}P(\tau_{i,j} = n) = 1$. If $\mathbf{1}$ is the vector of all 1's, then we have that

$$(I - Q)\phi'_{\tau_{*,j}}(0) = iP_1 + Q\mathbf{1}$$

$$\phi'_{\tau_{*,j}}(0) = i(I - Q)^{-1}(P_1 + Q\mathbf{1})$$

It follows that $Q\mathbf{1}_i = \sum_{k\neq j} P(\tau_{i,k} = 1)$. Therefore, $(P_1 + Q\mathbf{1}) = \mathbf{1}$

$$\phi'_{\tau_{*,j}}(0) = i(I - Q)^{-1}\mathbf{1}$$

So then

$$E[\tau_{*,j}] = (I - Q)^{-1}\mathbf{1} = \sum_{n=0}^{\infty} Q^n\mathbf{1}$$

We then have that second derivative of this is

$$\phi''_{\tau_{*,j}}(t) = -2Q(I - Q)^{-2}\mathbf{1}$$

Which means that

$$E[\tau_{*,j}^2] = 2Q(I - Q)^{-2}\mathbf{1}$$

$$E[\tau_{*,j}^2] = 2\sum_{n=0}^{\infty} nQ^n\mathbf{1}$$

When it comes to actually computing these quantities. The above calculations exist for approximate values found by a computer. In some cases, it is feasible to use the above formulae to find the distributions for general classes of graphs. The cases below are generally easier to compute. Later in this paper, we will introduce machinery to tackle harder cases.

Complete Case

The complete graph is where each node is connected to every other node. It looks like this. In this case, it is clear that for a complete graph of k nodes, we have

$$P(\tau_{i,j} = 1) = \frac{1}{k-1}$$

if we have a simple random walk. Then, it is clear that for all vertices i, j in the graph

$$P(\tau_{i,j} = n) = \left(\frac{k-2}{k-1}\right)^{n-1} \frac{1}{k-1}$$

Where the walker makes $n-1$ "wrong" moves and then makes the right one once. We can see above that the distribution is geometric with success probability of $\frac{1}{k-1}$ so

$$E[\tau_{i,j}] = k - 1$$

$$Var[\tau_{i,j}] = (k-1)(k-2)$$

Complete Bipartite Case

A complete bipartite graph consists of two disjoint sets of nodes, say of sizes k_1 and k_2, where every node in one set is connected to every node in the other, but there are no edges within a set.

Suppose the walker starts at a node in set A (with k_1 nodes), and aims to reach a node in set B (with k_2 nodes). The walker alternates between the two sets on each step due to the bipartite structure.

In this case, the probability of hitting a particular node in the opposite set in one step is:

$$P(\tau_{i,j} = 1) = \frac{1}{k_2} \quad \text{if } i \in A,\ j \in B$$

Because the walker must switch sets each time, for i, j in opposite sets, $\tau_{i,j}$ takes only odd values. The probability of reaching j in exactly $2n-1$ steps involves making $n-1$ failed visits to other nodes in the target set, interspersed with returns to the starting set. This yields a geometric-like structure with a modified step size, so then we have

$$P(\tau_{i,j} = 2n+1) = \left(\frac{k_2-1}{k_2}\right)^{n-1} \frac{1}{k_2}, \text{if } i \in A, j \in B$$

$$P(\tau_{i,j} = 2n) = \left(\frac{k_2-1}{k_2}\right)^{n-1} \frac{1}{k_2}, \text{if } i \in A, j \in A$$

The hitting time distribution is again geometric (over odd steps), and the expected hitting time from one set to the other is:

$$\mathbb{E}[\tau_{i,j}] = 2k_2 - 1 \quad \text{if } i \in A,\ j \in B$$

Similarly, for transitions within the same set (which require at least two steps), the expected hitting time is:

$$\mathbb{E}[\tau_{i,j}] = 2k_2 \quad \text{if } i,j \in A,\ i \neq j$$

So then variance follows

$$Var[\tau_{i,j}] = 4(k_2)(k_2 - 1) \quad \text{if } i \in A,\ j \in B$$

$$Var[\tau_{i,j}] = 4(k_2)(k_2 - 1) \quad \text{if } i \in A,\ j \in A$$

The variances are the same as both of these processes are the same as there is full probability that a walker moves from A to B or back every move. The first case where $i \in A, j \in B$ is offset from when $i, j \in A$ by one move.

Cycle Case

Let's perform these computations for a cycle. It clearly follows for a k-cycle (denoted as C_k)

$$Q = \begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 & ... & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & ... & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & \frac{1}{2} & 0 \end{bmatrix}, P_1 = \begin{bmatrix} \frac{1}{2} \\ 0 \\ \vdots \\ 0 \\ \frac{1}{2} \end{bmatrix}$$

So then we have

$$\begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 & ... & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & ... & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & \frac{1}{2} & 0 \end{bmatrix} \begin{bmatrix} P(\tau_{1,0} = n-1) \\ P(\tau_{2,0} = n-1) \\ ... \\ P(\tau_{k-1,0} = n-1) \end{bmatrix} = \begin{bmatrix} P(\tau_{1,0} = n) \\ P(\tau_{2,0} = n) \\ ... \\ P(\tau_{k-1,0} = n) \end{bmatrix}$$

Or in other words

$$\begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 & ... & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & ... & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & \frac{1}{2} & 0 \end{bmatrix}^{n-1} \begin{bmatrix} P(\tau_{1,0} = 1) \\ P(\tau_{2,0} = 1) \\ ... \\ P(\tau_{k-1,0} = 1) \end{bmatrix} = \begin{bmatrix} m_n(1) \\ m_n(2) \\ ... \\ m_n(k-1) \end{bmatrix}$$

So it follows we want to take the diagonalization of the Toeplitz Matrix above. Let us call that matrix H and its diagonlization LDL^{-1}. Thankfully, the eigenvectors and eigenvalues for tridiagonal toeplitz matrices are well known and with that we have the following as the diagonalization. These eigenvalues were found in [1]

$$\begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 & ... & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & ... & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & \frac{1}{2} & 0 \end{bmatrix}$$

$$= \frac{2}{k} \begin{bmatrix} \sin(\frac{\pi}{k}) & \sin(\frac{2\pi}{k}) & ... & \sin(\frac{(k-1)\pi}{k}) \\ \sin(\frac{2\pi}{k}) & \sin(\frac{4\pi}{k}) & ... & \sin(\frac{2(k-1)\pi}{k}) \\ ... & ... & ... & ... \\ \sin(\frac{(k-1)\pi}{k}) & \sin(\frac{2(k-1)\pi}{k}) & ... & \sin(\frac{(k-1)^2\pi}{k}) \end{bmatrix} \begin{bmatrix} \cos(\frac{\pi}{k}) & & 0 & ... & 0 \\ 0 & \cos(\frac{2\pi}{k}) & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & \cos(\frac{(k-1)\pi}{k}) \end{bmatrix}$$

$$\begin{bmatrix} \sin(\frac{\pi}{k}) & \sin(\frac{2\pi}{k}) & ... & \sin(\frac{(k-1)\pi}{k}) \\ \sin(\frac{2\pi}{k}) & \sin(\frac{4\pi}{k}) & ... & \sin(\frac{2(k-1)\pi}{k}) \\ ... & ... & ... & ... \\ \sin(\frac{(k-1)\pi}{k}) & \sin(\frac{2(k-1)\pi}{k}) & ... & \sin(\frac{(k-1)^2\pi}{k}) \end{bmatrix}$$

As we know from earlier,

$$\begin{bmatrix} P(\tau_{1,0} = 1) \\ P(\tau_{2,0} = 1) \\ ... \\ P(\tau_{k-1,0} = 1) \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ 0 \\ ... \\ \frac{1}{2} \end{bmatrix}$$

So, then we first multiply by L^{-1} or the last matrix in the diagonalization.

$$\begin{bmatrix} \sin(\frac{\pi}{k}) & \sin(\frac{2\pi}{k}) & ... & \sin(\frac{(k-1)\pi}{k}) \\ \sin(\frac{2\pi}{k}) & \sin(\frac{4\pi}{k}) & ... & \sin(\frac{2(k-1)\pi}{k}) \\ ... & ... & ... & ... \\ \sin(\frac{(k-1)\pi}{k}) & \sin(\frac{2(k-1)\pi}{k}) & ... & \sin(\frac{(k-1)^2\pi}{k}) \end{bmatrix} \begin{bmatrix} \frac{1}{2} \\ 0 \\ ... \\ \frac{1}{2} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \sin(\frac{\pi}{k}) + \sin(\frac{(k-1)\pi}{k}) \\ \sin(\frac{2\pi}{k}) + \sin(\frac{2(k-1)\pi}{k}) \\ ... \\ \sin(\frac{(k-1)\pi}{k}) + \sin(\frac{(k-1)^2\pi}{k}) \end{bmatrix}$$

We next multiply by the diagonal matrix.

$$\frac{1}{2} \begin{bmatrix} \cos(\frac{\pi}{k}) & & 0 & ... & 0 \\ 0 & \cos(\frac{2\pi}{k}) & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & \cos(\frac{(k-1)\pi}{k}) \end{bmatrix}^{n-1} \begin{bmatrix} 0 \\ \sin(\frac{2\pi}{k}) \\ 0 \\ \sin(\frac{4\pi}{k}) \\ ... \end{bmatrix} = \frac{1}{2} \begin{bmatrix} cos(\frac{\pi}{k}) \sin(\frac{\pi}{k})^{n-1} + \sin(\frac{(k-1)\pi}{k})) \\ cos(\frac{2\pi}{k})^{n-1} \sin(\frac{2\pi}{k}) + \sin(\frac{2(k-1)\pi}{k}) \\ ... \\ cos(\frac{(k-1)\pi}{k})^{n-1}(\sin(\frac{(k-1)\pi}{k}) + \sin(\frac{(k-1)^2\pi}{k})) \end{bmatrix}$$

And lastly multiplying by L.

$$\frac{2}{k} \begin{bmatrix} \sin(\frac{\pi}{k}) & \sin(\frac{2\pi}{k}) & ... & \sin(\frac{(k-1)\pi}{k}) \\ \sin(\frac{2\pi}{k}) & \sin(\frac{4\pi}{k}) & ... & \sin(\frac{2(k-1)\pi}{k}) \\ ... & ... & ... & ... \\ \sin(\frac{(k-1)\pi}{k}) & \sin(\frac{2(k-1)\pi}{k}) & ... & \sin(\frac{(k-1)^2\pi}{k}) \end{bmatrix} \frac{1}{2} \begin{bmatrix} cos(\frac{\pi}{k}) \sin(\frac{\pi}{k})^{n-1} + \sin(\frac{(k-1)\pi}{k})) \\ cos(\frac{2\pi}{k})^{n-1} \sin(\frac{2\pi}{k}) + \sin(\frac{2(k-1)\pi}{k}) \\ ... \\ cos(\frac{(k-1)\pi}{k})^{n-1}(\sin(\frac{(k-1)\pi}{k}) + \sin(\frac{(k-1)^2\pi}{k})) \end{bmatrix}$$

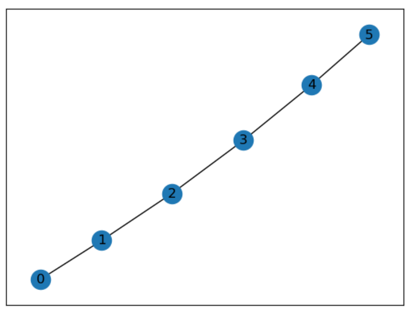
$$= \begin{bmatrix} \frac{1}{k}\sum_{j=0}^{k-1}\cos(\frac{j\pi}{k})^{n-1}(\sin(\frac{j\pi}{k})+\sin(\frac{j(k-1)\pi}{k}))\sin(\frac{j\pi}{k}) \\ \frac{1}{k}\sum_{j=0}^{k-1}\cos(\frac{j\pi}{k})^{n-1}(\sin(\frac{j\pi}{k})+\sin(\frac{j(k-1)\pi}{k}))\sin(\frac{2j\pi}{k}) \\ \frac{1}{k}\sum_{j=0}^{k-1}\cos(\frac{j\pi}{k})^{n-1}(\sin(\frac{j\pi}{k})+\sin(\frac{j(k-1)\pi}{k}))\sin(\frac{3j\pi}{k}) \\ \dots \\ \frac{1}{k}\sum_{j=0}^{k-1}\cos(\frac{j\pi}{k})^{n-1}(\sin(\frac{j\pi}{k})+\sin(\frac{j(k-1)\pi}{k}))\sin(\frac{(k-1)j\pi}{k}) \end{bmatrix}$$

Therefore

$$P(\tau_{i,j}=n)=\frac{1}{k}\sum_{j=0}^{k-1}\cos(\frac{j\pi}{k})^{n-1}(\sin(\frac{j\pi}{k})+\sin(\frac{j(k-1)\pi}{k}))\sin(\frac{ij\pi}{k})$$

Extensions

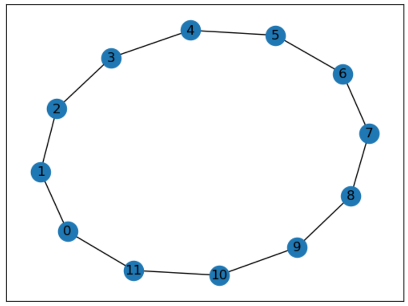
One important class of graphs where hitting time distributions can be derived explicitly is the cycle. Consider the path graph P_k, which is a chain of k nodes connected in sequence.



For generality, we take one of the endpoints as the absorbing state (i.e., the destination node).

This assumption does not result in a loss of generality. Suppose instead we selected an interior node (e.g., node 3 in a 5-node path) as the destination. Then, any walk from one end (e.g., node 1) that eventually reaches the other end (e.g., node 5) without hitting the interior node is irrelevant to the hitting time to that interior node. Thus, conditioning on reaching the interior node without escape effectively partitions the walk into two disjoint path subgraphs. Hence, the analysis of hitting times to an endpoint captures the general behavior.

Now consider the cycle graph C_{2k}, where nodes are connected in a closed loop of $2k$ vertices. By folding the cycle along the line of symmetry, we can pair nodes that are equidistant from a chosen target node. (The line of symmetry would be a hypothetical line from 0 to 6.



Under this pairing, the probability of stepping toward or away from the target remains unchanged, and the resulting random walk is statistically identical (in terms of hitting time) to that of a path graph with k nodes. Therefore, the hitting time distribution for a path of k nodes corresponds exactly to that of a cycle of $2k$ nodes under this symmetry.

In the context of Cayley graphs, the cycle graph on k nodes can be seen as the Cayley graph of the cyclic group $\mathbb{Z}_k$ with generating set $\{\pm 1\}$. Extending this idea, we now consider the group $\mathbb{Z}_p^2$, where p is an odd prime. A standard set of generators for its Cayley graph is:

$$\{(\pm 1, 0), (0, \pm 1)\},$$

which gives rise to the 2D torus graph — a grid with wrap-around edges.

However, an alternative set of generators is:

$$\{(\pm 1, \pm 1)\},$$

which corresponds to diagonal steps in the lattice. These generators also produce a valid Cayley graph, albeit with a rotated geometry. Importantly, these generator sets are related via a linear automorphism:

$$\phi(a, b) = \left(\frac{a+b}{2}, \frac{a-b}{2} \right),$$

which is an automorphism of $\mathbb{Z}_p^2$ since 2 is invertible in $\mathbb{Z}_p$ (as p is odd). This transformation maps the standard coordinate basis to the new one spanned by the diagonal generators.

Let $c_n(i)$ denote the probability that a walk on a p-cycle starting at 0 hits i for the first time at step n. If we define $\phi^{-1}(a-c, b-d) = (a', b')$, then (a', b') represents the displacement between (a, b) and (c, d) expressed in terms of the diagonal generator basis. Under the assumption of independence in each coordinate (which holds due to the structure of the walk), the hitting time distribution on $\mathbb{Z}_p^2$ can be expressed as:

$$P(\tau_{(a,b),(c,d)} = n) = \sum_{i=0}^{n} c_i(a') \cdot c_{n-i}(b').$$

That is, the distribution of the hitting time is a convolution of the 1D hitting time distributions along the transformed coordinates.

Fourier View for Finite Groups

Much of the ideas of the section orginated from the paper [4] Let's suppose we have a random walk on a group G. For each successive time step, we have that for $x, y \in G$ $p^*(x, y)$ is the probability of moving from x to y. For random walks on groups, we are assuming a time-independent increment distribution. Therefore, we can define a new function $p(g) = p^*(x, xg) = P(\tau_{x,xg} = 1)$.

For this portion of the paper, we are concerned with the distribution of $\tau_{g,e}$ on finite groups G. For the ease of writing, we will define that $P(\tau_{g,e} = n) = m_n(g)$. It follows that $p(g) = m_1(g)$ It follows that we have

$$m_n(g) = \sum_{s \in G} p^*(g, gs^{-1}) m_{n-1}(gs^{-1})$$

and

$$m_n(e) = 0$$

for $n \geq 1$. So, we can design a new function c_n

$$c_n(g) = \begin{cases} 0 & g \neq e \\ \sum_{s \in G} m_{n-1}(s) m_1(s) & g = e \end{cases}$$

$$m_n(g) = -c_n(g) + \sum_{s \in G} m_1(s^{-1}) m_{n-1}(gs^{-1})$$

For the purposes of this write-up, we will assume that our random walk is symmetric.

$$m_n(g) = -c_n(g) + \sum_{s \in G} m_1(s) m_{n-1}(gs^{-1})$$

$$\widehat{m_n}(\rho_j) = -I \sum_{s \in G} m_{n-1}(s) p(s) + \widehat{p}(\rho_j) \widehat{m_{n-1}}(\rho_j)$$

$$\widehat{m_n}(\rho_j) = -I \sum_{s \in G} m_{n-1}(s) m_1(s^{-1}) + \widehat{m_1}(\rho_j) \widehat{m_{n-1}}(\rho_j)$$

By Plancharel's Theorem on Abelian Groups and the fact that Abelain groups have exactly the same number of irreducible representations and group elements.

$$\widehat{m_n}(\rho_j) = -\frac{1}{k} \sum_{a=0}^{k-1} \widehat{m_{n-1}}(\rho_a) \widehat{m_1}(\rho_a) + \widehat{m_1}(\rho_j) \widehat{m_{n-1}}(\rho_j)$$

and so then we have that

$$\widehat{m_n}(\rho_j) = -\frac{1}{k} \sum_{a=0}^{k-1} \widehat{m_{n-1}}(\rho_a) \widehat{m_1}(\rho_a) + \widehat{m_1}(\rho_j) \widehat{m_{n-1}}(\rho_j)$$

So then we have a recurrence relation such that.

$$A \begin{bmatrix} \widehat{m_{n-1}(\rho_0)} \\ \widehat{m_{n-1}(\rho_1)} \\ ... \\ \widehat{m_{n-1}(\rho_{k-1})} \end{bmatrix} = \begin{bmatrix} \widehat{m_n(\rho_0)} \\ \widehat{m_n(\rho_1)} \\ ... \\ \widehat{m_n(\rho_{k-1})} \end{bmatrix}$$

Where

$$A = \begin{bmatrix} \frac{k-1}{k}\widehat{m_1}(\rho_0) & -\frac{1}{k}\widehat{m_1}(\rho_1) & ... & -\frac{1}{k}\widehat{m_1}(\rho_{k-1}) \\ -\frac{1}{k}\widehat{m_1}(\rho_0) & \frac{k-1}{k}\widehat{m_1}(\rho_1) & .... & -\frac{1}{k}\widehat{m_1}(\rho_{k-1}) \\ ... & ... & ... & ... \\ -\frac{1}{k}\widehat{m_1}(\rho_0) & -\frac{1}{k}\widehat{m_1}(\rho_1) & .... & \frac{k-1}{k}\widehat{m_1}(\rho_{k-1}) \end{bmatrix}$$

$$A = \begin{bmatrix} \frac{k-1}{k} & -\frac{1}{k} & ... & -\frac{1}{k} \\ -\frac{1}{k} & \frac{k-1}{k} & .... & -\frac{1}{k} \\ ... & ... & ... & ... \\ -\frac{1}{k} & -\frac{1}{k} & .... & \frac{k-1}{k} \end{bmatrix} \begin{bmatrix} \widehat{p}(\rho_0) & 0 & ... & 0 \\ 0 & \widehat{p}(\rho_1) & .... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & .... & \widehat{p}(\rho_{k-1}) \end{bmatrix}$$

Which means that we have that

$$A^{n-1}\widehat{m_1} = \widehat{m_n}$$

I haven't made any assumptions so far except for symmetric random walks and abelian groups. Right now, we will be working with cycles and then generalize to abelian groups as a whole.

Association Schemes

Let A_k be the distance indicator matrices of k matrices. Let $M_{n_{i,j}} = P($Probability that it takes n steps for first
It is clear that any vertex transitive graph that we have.

$$M_0 = I$$

and

$$M_1 = \frac{1}{d}A^1 - \frac{1}{d}\frac{Trace(A)}{V}I$$

This makes sure that a vertex can't visit itself in one move. We also have.

$$M_2 = \frac{1}{d^2}A^2 - \frac{1}{d}\frac{Trace(A)}{V}M_1 - \frac{1}{d^2}\frac{Trace(A^2)}{V}M_0$$

This has it so we can't visit our selves in 2 moves or 1 move. Generalizing this

$$M_n = \frac{1}{d^n}A^n - \sum_{k=1}^{n}\frac{1}{d^k}\frac{Trace(A^k)}{V}M_{n-k}$$

Basically this says, we are not allowed to visit the node earlier than n moves and then make a loop to that same node to achieve a walk of technically *technically* n moves from i to j. We get this sum.

$$\sum_{k=0}^{n} \frac{1}{d^k} \frac{Trace(A^k)}{V} M_{n-k} = \frac{1}{d^n} A^n$$

As this is a Cauchy product, we get the following

$$\sum_{n=0}^{\infty} \frac{t^n}{d^n} \frac{Trace(A^n)}{V} \sum_{n=0}^{\infty} M_n t^n = \sum_{n=0}^{\infty} \frac{t^n}{d^n} A^n$$

By definition of trace we have.

$$\frac{1}{V} \left(\sum_{n=0}^{\infty} \frac{t^k}{d^k} \sum_{i=0}^{V} \lambda_i^k \right) \sum_{n=0}^{\infty} M_n t^n = \sum_{n=0}^{\infty} \frac{t^n}{d^n} A^n$$

$$\frac{1}{V} \left(\sum_{i=0}^{V} \sum_{n=0}^{\infty} \lambda_i^k \frac{t^k}{d^k} \right) \sum_{n=0}^{\infty} M_n t^n = \sum_{n=0}^{\infty} \frac{t^n}{d^n} A^n$$

$$\frac{1}{V} \left(\sum_{i=0}^{V} \frac{1}{1 - \frac{\lambda_i t}{d}} \right) \sum_{n=0}^{\infty} M_n t^n = (I - \frac{t}{d} A)^{-1}$$

so for $|t| < 1$

$$\sum_{n=0}^{\infty} M_n t^n = \frac{V(I - \frac{t}{d} A)^{-1}}{\sum_{i=0}^{V} \frac{1}{1 - \frac{\lambda_i t}{d}}}$$

This is a very general result for all vertex transitive graphs. Which apply to all Cayley graphs as well.

Acknowledgments

Thank you to Jonathan Pakianthan, Alex Iosevich and Matthew Dannenburg for their continued support. Thank you to my team members at STEM FOR ALL 2024: Yujia Hu, Yiling Zou, Charlie Li. Thank you to my family and friends for supporting me through this journey to be an Honors Math Major.

References

[1] Hatice Kübra Duru and Durmuş Bozkurt. Integer powers of certain complex (2k+1)-diagonal toeplitz matrices. *Selcuk University, Science Faculty Department of Mathematics*, August 2018. Accessed August 24, 2018.

[2] Enzhi Li. A fast method to calculate hitting time distribution for a random walk on connected undirected graph, 2019. arXiv preprint arXiv:1908.09644.

[3] Yuuho Tanaka. On the average hitting times of weighted cayley graphs, 2023.

[4] Christopher Zhang. Formulas for hitting times and cover times for random walks on groups, 2023.