OXFORD JOURNAL
OF STUDENT
SCHOLARSHIP

Effects of Edge Deletion on First-Return Time Distributions of Simple Random Walks

Vincent Fortajada
vfortajada@gmail.com

ABSTRACT

In modern times, many mathematicians are concerned with the dynamics of various graph families such as Random Regular Graphs and Erdős-Rényi-type graphs. We are particularly concerned with the question: how does the random elimination of edges influence the return-time distributions of a simple random walk for 5 random starting positions on a graph? Throughout this paper, we aim to address this question using various simulations and performing the analysis of simple Markov Chains on nine distinct families of structured graphs, including Random Regular Graphs, Erdős-Rényi-type graphs, and the Complete Graph. Within this paper, we also run into the issue of insufficient edge density resulting in infinite return times for smaller graph structures post-perturbation, which had the impact of decreasing results stability for higher noise level. Ultimately, we concluded that random edge noise causes statistically significant shifts in the return time distributions of simple random walks, and displayed the impact of edge density on return times arising from disconnection failures in the context of random edge perturbations and return times.

Keywords: Random Walk, Return Time Distribution, Edge Perturbation, Random Regular Graphs, Erdős–Rényi Graphs, Edge Density

INTRODUCTION

A group of aliens lands in an unfamiliar part of Earth. Driven by curiosity, they begin exploring the neighborhood, but without a reliable GPS, they choose a random road at every intersection. Since their resources to survive are limited, they want to estimate how long it will take to return to their starting point. They encounter a human who is also new to the city but knows some basic facts about the road network. However, the human is running late and can only share a little information. What key questions about the city's road network structure should the aliens ask to best estimate their return time?

Graph theory is a field of mathematics that involves the study of graphs or networks (consisting of edges and vertices). This theory has various applications across the real world, and examples of these are in navigation to solve problems such as the aliens'. An important concept in graph theory is the simple random walk. A simple random walk is a process that occurs when a "walker" moves from an initial vertex to a neighboring vertex with either a weighted probability or an equal chance based on the degree of its current vertex, forming a sequence of vertices traveled. Random walks have many measurable quantities of interest, such as return times, which are the number of steps it takes a "walker" to return to the initial vertex.

Return time distributions are heavily dictated by graph structure. This fact is discussed in several research studies. Unfortunately, these interesting papers exclusively focus on static/unperturbed graphs. Separately, a stream of research has examined how graphs respond to structural change more broadly, including targeted attacks and dynamically evolving structures. These two research directions rarely intersect: the effect of controlled, graduated structural perturbation on the complete first-return time distribution, rather than on these summary metrics alone, remains largely unexamined. This study bridges these two streams by examining how the return time distribution, not solely its aggregate summary statistics, responds to controlled, graduated noise perturbation across multiple graph types, extending our understanding of graph perturbation into the distributional detail that has so far been studied from static, unperturbed graphs. Specifically, this paper helps bridge the gap between these two streams of research by answering the question: how does random edge noise influence the stability of return-time distributions of a simple random walk for 5 random starting positions on small and large graphs?

PRELIMINARIES

In what follows, we will only cover random walks in which each possible step from an origin state has an equal and unweighted probability, and graphs in this experiment are finite, simple, and undirected; however, they may not be necessarily connected. A disconnected graph could result in a random walk being trapped in a disconnected piece of the graph, and as a result, return time approaches infinity. If this happens a substantial number of times per simulation, distributions may fail to meet the criteria necessary to perform statistical analysis. All theoretical results relying on stationary distributions are conditional on graph connectivity, which ensures the random walk is irreducible.

A graph is a mathematical structure involving a group of *vertices* (nodes) connected to each other by *edges*. The two structures that may form in graphs that are of great importance when researching return times are the *cycle* and the *connected graph*. A connected graph structure is a structure in which there exists a path between every pair of vertices, and it is possible to reach every vertex of a graph directly or indirectly. A cycle is a structure of a graph in which there exists a closed path where the starting and ending vertex are the same, and every other vertex is distinct. Graphs can have many shapes depending on the arrangement of their vertices; however, working with them can be simplified by focusing on their vertices and edges while performing mathematical analysis.

The vertices of a graph have important measurable quantities. One such quantity, which we shall use frequently throughout this paper, is the *degree* of a vertex. The degree of a vertex is the number of edges adjacent to that vertex.

MARKOV CHAINS

Following Grinstead and Snell (2006), we define a *Markov Chain* to be a sequence of states in which movement between states is known as a *step*. For such a sequence to be defined as a Markov Chain, it must satisfy the *Markov Property*, which is the property that if a chain is currently in state s_i and moves to state s_j, the transition probability between the two states is $p_{ij,}$ which does not depend on which states the chain was in before the current state.

Grinstead and Snell (2006) also state that transition probabilities between the states of a Markov Chain can be organized into matrices known as *transition matrices*. Working with these matrices, they derive the following theorem: Let P be the transition matrix of a Markov Chain. The *ij*th entry $p^{(n)}_{ij}$ of matrix P^n gives the probability that the Markov Chain, starting in state s_i, will be in state s_j after n steps. This theorem shows that by sorting transition probabilities into matrices, we can perform standard matrix operations to analyze Markov Chains as a collective group of probabilities rather than individually.
When working with random walk Markov Chains, it is best to analyze a Markov Chain's long-term behavior. Further analyzing transition matrices, Grinstead and Snell (2006) derive the following conclusion: We now consider the long-term behavior of a Markov chain when it starts in a state chosen by a probability distribution on the set of states. As a result of this finding, we see that this set of states is known as a *probability vector*. This vector can be used to represent the rows of a transition matrix, and thus the original state of a Markov Chain. If u is a probability vector which represents the initial state of a Markov chain, then we think of the *i*th component of u as representing the probability that the chain starts in state s_i. The probability vector allows us to derive the following property of transition matrices, which will later be used in our analysis of random walks:

$$u^{(n)} = uP^n$$

where u is the probability vector u, P is the transition matrix P of a Markov Chain, and *n* is the total number of steps taken from the initial state of the Markov Chain.

We can use this theorem to find the probability that the chain is in state s_i after n steps, as said probability is the ith entry of this vector.

RANDOM WALKS

As defined by Lovász (1993), a *random walk* is a time-reversible Markov Chain in which, given a vertex on a graph, a neighbor vertex is selected at random and moved to, forming a sequence of points along a graph known as a random walk. Using this definition, we can perform mathematical analysis on random walks similar to a Markov Chain. We first shift our focus to the probability of moving to a specific state of a random walk. According to Lovász (1993), we can form a matrix M as a means of viewing the transition probabilities of a random walk, in which he defines each probability within M as an output of probability distribution $P_t(i)$ where:

$$P_{ij}(i) = \frac{1}{d(i)}$$

if ij is an edge, or 0 otherwise.

Now that we have established random walks as time-reversible Markov Chains, we can perform the linear algebra analysis that can be performed on Markov Chains on random walks. To proceed, let us define the properties of M:

1) M is a stochastic matrix
2) If we allow D to be a diagonal matrix such that $(D)_{ii} = \frac{1}{d(i)}$, and A to be the adjacency matrix of a given graph G, then $M = DA$
3) If G is d-regular, then $M = \frac{1}{d}A$

Using these properties, we can rewrite the transition matrix of a random walk to be a product of two matrices, which will be useful for the analysis of the stable distribution. We can define the probability (row) vector of the probability distribution of our random walk as:

$$p_n(i) = P(X_n = i)$$

where X_n is the state of the random walk at time n, and i is a vertex of G

After performing analysis using this new probability vector, we can substitute $p_n(i)$ into our previous row vector analysis equation to get:

$$p_n = p_0 M^n$$

where M is the transition matrix M, n is the number of steps taken from the starting vertex, and p_0 is the initial distribution of the random walk at its starting vertex

THE STATIONARY DISTRIBUTION

When working with random walks, we are likely to deal with probability distributions when working to find return times on different graphs. In our research, we primarily deal with long-term outcomes, and thus, it is important to find a probability distribution that describes the behavior of a random walk over a long period of time at the equilibrium state of the system.

To find this distribution, we must state the following theorem attributed from (Eriksson, 2023) known as the *Perron-Frobenius Theorem*:
Let $A \geq 0$ be an irreducible matrix. Then,

a) $\rho(A) > 0$
b) $\rho(A) \in \sigma(A)$ and $\rho(A)$ is simple
c) there exists a unique right eigenvector p of A, which can be taken positive, such that $Ap = \rho(A)p$ and $||p||_1 = p_1 + \ldots + p_n = 1$
d) there exists a unique left eigenvector q^T of A, which can be taken positive, such that $q^T A = \rho(A)q^T$ and $q^T p = p_1 q_1 + \ldots + p_n q_n = 1$

If we allow G to be a connected graph, then Perron-Frobenius applies to M, implying the existence of a unique probability distribution π, which is a positive left eigenvector of M associated to its dominant eigenvalue $\lambda = 1$. Using our previous linear algebra analysis:

a) $\pi = \pi M$
b) $\pi(i) > 0$ for all $i \in V$
c) $\sum_{i \in V} \pi(i) = 1$

If we draw the random walk from the initial vertex from π, then:

$$p_n = \pi P^k = \pi$$

Thus, this random walk is a stationary stochastic process.

Now that we know our random walk is a stationary walk, we can derive a formula for its distribution. (Lovász, 1993) states that for a graph similar to G, "A one-line calculation shows that for every graph G, the distribution $\pi(i) = \frac{d(v)}{2m}$ is stationary." This statement applies to our graph G, and thus the *stationary distribution* for our random walk is $\pi(i) = \frac{d(v)}{2m}$. In particular, if G is d-regular, then $\pi(i) = \frac{d}{2m} = \frac{1}{n}$, and π is the uniform distribution.

Time-reversibility is the property that in a random walk, we take steps forwards as often as backward. (Lovász, 1993) states that mathematically, this means:

$$\pi(i)p_{ij} = \pi(j)p_{ji}$$

Due to π being a stationary distribution, it must satisfy the *Detailed Balance Condition*, which implies that our random walk is time-reversible. Moreover, using previous analysis, we can see that $\pi(i)p_{ij} = \frac{1}{2m}$ for all $ij \in E$. We know that a random walk moves along every edge in a given direction with this

frequency; because our random walk is also time-reversible, if the random walk passes through an edge, then the expected number of steps before it traverses the same edge again in the same direction is 2m.

This same property holds for vertices; thus, the expected number of steps before a random walk returns to its initial vertex is:

$$t = \frac{2m}{d(i)} = \frac{1}{\pi(i)}$$

This result is known as Kac's Formula (Lovász, 1993).

The number of steps it takes for a random walk to return to its starting vertex is defined as the return time of a random walk. Return times are still extensively researched in mathematics, especially return times on perturbed graphs, and the following section will cover the known research on this topic.

LITERATURE REVIEW

THEORETICAL FRAMEWORK

Return time distributions are studied in research when it comes to studying return times. In particular, we are interested in the notion of a first-return time, which is the return time of the first time a random walker takes to return to the initial vertex. To begin, we look at the analytic characterizations of first-return time distributions involving random regular graphs. Prior work on random regular graphs shows that first-return distributions decay exponentially due to the structural uniformity of the network (Tishby et al., 2021). Crucially, this implies that a decrease in structural uniformity will distort this clean exponential behavior, typically introducing heavy-tailed distributions rather than uniform geometric decay (Tishby et al., 2021). Tishby analytically approximates infinite RRGs as local Bethe lattices or tree structures, which are easier to work with to find first-return distributions. More specifically, Tishby argues that because every vertex shares the same degree c, transition probabilities are equivalent at every step, thus preventing any structural traps that may hinder the random walk process from returning to the original vertex.

In addition to RRGs, mathematicians have also researched first-return distributions on Erdős-Rényi networks. Disordered graph structures fundamentally change random walk behavior in comparison to RRGs, which have a more uniform graph structure. In sparse Erdős-Rényi graphs, variations in connectivity create structural irregularities that can trap random walkers, resulting in significantly heavier tails in the return-time distribution (Evnin & Horinouchi, 2024). This effect occurs because a sparse Erdős-Rényi network has complex, non-uniform structures, such as condensed clusters or barely connected groups, that contain irregular connections. Any random walker could stumble into one of these structures and find themselves "trapped" in a loop topologically. Traps significantly increase return times, stretching a first-return distribution's tail into slow power-law or stretched exponential decays. This decay contrasts heavily with the behavior present in the first-return distributions of random walks on RRGs (Tishby et al., 2021). Because an RRG is largely uniform in structure, it lacks the complex, non-uniform structures that create local traps, contrasting with the unpredictable nature of Erdős-Rényi networks in both walk behavior and first-return distributions.

Beyond uniform and randomly sparse graph structures, graphs featuring highly heterogeneous structures require an entirely vertex-specific approach to transport dynamics. Research by Masuda and Konno (2004) demonstrates that the first-return time distribution of simple random walks on generalized random graphs is strictly dictated by the local degree of the initial vertex, rendering global network averages inaccurate. Variance across starting positions is mathematically captured through recursion formulas and degree-generating functions evaluated under a memoryless approximation. On generalized configurations, the graph shape is dominated by hub vertices and highly isolated, peripheral leaf nodes. When a random walk starts on a high-degree hub, the local transition probabilities scale such that the walker is likely to execute a rapid, two-step return pathway back to the initial vertex. Similarly, walks beginning on a low-degree peripheral node are structurally forced away into the graph, increasing return times such that the distribution's tail is stretched into heavy-tailed figures. Ultimately, this contrasts sharply with behavior involving RRGs and Erdős-Rényi networks due to the dependence on the initial vertex's degree of generalized configurations.

It is vital to note that localized structures and short cycles strongly shape return behavior, while large-scale global architecture matters less than expected in sparse random graph settings (Tishby et al., 2021; Hörmann et al., 2024). This primarily occurs because random walks are initially blind to large network dimensions, meaning early trajectories are dictated by the immediate topological neighborhood. Even when using localized tree approximations to simplify the graph's shape, small, dense local loops will inevitably disrupt these clean branches and trigger early returns. This local dominance contrasts sharply with uniform RRGs, while conversely setting the stage for the behavior of walks on generalized random graphs.

Noise sensitivity is another topic of interest that mathematicians have studied, and tools such as spectral quantities are used to study it extensively. Beyond pure structural analysis, advanced spectral matrix frameworks allow researchers to isolate topological dynamics from structural noise. The works of Stephan and Massoulié (2020), on non-backtracking spectra, as well as Wei (2022), on signless Laplacian embeddings, highlight that transition matrices of simple random walks can be bounded by predictable eigenvalue limits within highly irregular or weighted nonuniform random graphs. Spectral methods protect random walk models from being analytically overwhelmed by local topological anomalies. By using a non-backtracking matrix, researchers prevent a random walk from immediately backtracking over its last step, which suppresses the dominant, noisy eigenvalues generated by localized high-degree hubs and forces global path structures to the forefront (Stephan & Massoulié, 2020). At the same time, mapping the graph using a signless Laplacian embedding keeps information about localized cycles and density variations that standard, signed Laplacians often erase through mathematical cancellation (Wei, 2022). Together, these constraints force the walk's long-term transition probabilities to follow strict geometric boundaries based entirely on local network density and short-loop configurations.

This dual spectral approach gives researchers a necessary mathematical upgrade when moving away from perfectly symmetrical graph models. On a uniform RRG, standard adjacency matrices suffer from no

irregular degree fluctuations, meaning standard spectral bounds remain naturally tight (Tishby et al., 2021). However, as we move to sparse Erdős-Rényi regimes or generalized random graphs, local structures and extreme structural hierarchies normally fill the standard spectrum with chaotic, localized eigenvalues (Evnin & Horinouchi, 2024; Masuda & Konno, 2004). Frameworks like the dual spectral approach discussed isolate that local noise, making a stable mathematical baseline that bridges uniform and heavily disordered graph structures.

EMPIRICAL RESEARCH

Mathematical theory as outlined by numerous researchers was backed convincingly by data collected from computer simulations. For instance, analytical distributions such as probability distributions and mean first-return times on RRGs were found to be in excellent agreement with results obtained from computer simulations. In addition, analysis of the tail behavior of the first-return times also agreed substantially with the theoretical form of the first-return distribution tail (Tishby et. al., 2021). Validation studies have also found strong agreement between statistical analysis of return time approximation and simulation results. For instance, in a study done by Hörmann (2024), they tested their approximation on a 6-regular random graph with 2^15 (~32,768) nodes and found excellent agreement between the approximation and simulation results, with the return-time distribution's tail showing a straight line on a semi-log plot, indicating clean exponential decay, matching what Tishby's theoretical framework predicts for regular structures. In addition, they also generated Poisson (Erdős-Rényi-type) random graphs with 100,000 nodes and mean degree 6, and compared the theoretical prediction for the distribution of tail-decay rates across nodes against the empirically observed distribution, again finding close agreement, and specifically noting this held not just for the average behavior but for the full distribution across nodes of differing degree (Hörmann, 2024). Simulation's role in validating return-time theory is not limited to confirming known results, but extends to establishing the reliability of results that cannot be derived exactly and that must come from computer simulations. Tishby et al. (2021) use simulation to find an exact analytical solution for RRGs, which shows that simulated data can reproduce theoretically derived distributions with high accuracy. Hörmann et. al. (2024) extends this justification further: their return-time formula is an approximation rather than an exact result. Altogether, these sources justify computer-generated data as an accurate and reliable research method that both confirms theoretical predictions and evaluates approximate ones where exact derivation is unable to be done.

Structural perturbation is the action of purposefully editing a graph by deleting edges. This might seem like unnecessarily meddling with a controlled variable at first; however, perturbation is a legitimate, established experimental lever for studying random walk sensitivity. For instance, graph perturbations are frequently used throughout a study by Chung in which several “edit-based graph evolving processes” (Chung & Robertson, 2025), such as simple and compound editing processes, are investigated and detailed by spectral analysis done on transition matrices. In addition, a study done by Kawamura demonstrates that the assumption that random walks are often assumed to be inherently robust to structural change due to their probabilistic nature does not hold under adversarial perturbation: targeted edge-rewiring attacks based on node centrality, clustering, or starting position significantly degrade random walk performance across five distinct walk algorithms and five graph structures (Kawamura,

2024). Furthermore, because the effect holds across five different walk algorithms and five different graph structures, Kawamura shows that this sensitivity isn't a fluke of one specific setup, and that new outcome measures such as return-time distributions can be studied with similar methods with confidence due to the generalizability of graph perturbation algorithms in research. This pattern extends across the broader literature on random walk resilience. Research done by Bala Priya C similarly evaluates walk performance under targeted node and edge removal using mixing time and coverage time, again bypassing the return-time distribution directly. Their findings nonetheless offer additional motivation for a multi-graph-type design, showing that different topologies exhibit markedly different resilience. Small-world networks retained 54.7% of their original robustness score (measure of how well a network maintains its connectivity and functionality post node removal) under 15% node removal, compared to more substantial degradation in scale-free networks. (Bala Prinya, 2025). Ultimately, Chung and Kawamura establish graph perturbation as both an analytically rigorous and empirically validated method, while Bala Priya C shows that its effects vary meaningfully across topologies rather than following a single trend. Together, these findings indicate that perturbation is not just a valid experimental lever but one whose effects are inherently structure-dependent—a relationship this study investigates directly by tracking how perturbation reshapes the first-return time distribution differently across graph types.

Vertex-level structural role has also been shown to influence random walk behavior, though through different mechanisms across the literature. Jin (2019) demonstrates that allowing each vertex a distinct restart probability, rather than a uniform one, significantly improves the performance of walk-based ranking and link prediction, inferring that a vertex's structural role shapes how it should be treated within the walk process itself. Thapar (2025) offers a complementary finding: the complete first-return time distribution, computed per vertex, is a mathematically grounded structural descriptor which is more informative than the eigenvalue spectrum and capable of identifying structural roles and enabling accurate graph alignment. Most importantly, this finding gives direct support to our study's core idea that the full return-time distribution, rather than a singular summary statistic, is a meaningful unit to analyze simple random walks.

Across these bodies of work, a consistent limitation emerges: none directly examine how the full return-time distribution responds to random, distributed structural perturbation on a graph. The validation literature considers only fixed, unperturbed graphs, despite the fact that even a single edge can eliminate a bottleneck and significantly alter the distribution. The perturbation literature, meanwhile, either treats the editing process itself as a purely mathematical object (Chung & Robertson, 2025) or applies targeted, rather than random, perturbation while measuring quantities such as hitting, cover, or mixing time rather than the distribution itself (Kawamura et al., 2024; Bala Prinya, 2025). Finally, work on vertex-level structural roles either applies the full return-time distribution only to static, unperturbed graphs (Thapar, 2025) or modifies the walk's rules rather than the graph it operates on (Jin, 2019).

Modifying a random walk's transition rules, independent of the graph it operates on, is also known to substantially affect its statistical behavior. Sauerwald and Zanetti (2019) show that random walks on dynamically changing graphs can, under conditions of stable stationarity, exhibit mixing and hitting time

behavior comparable to static graphs, which is a stability that does not hold when perturbation instead changes node degrees, as in the present study. Resetting mechanisms, by contrast, have been shown to more directly reshape core walk properties: González (2021) derives how stochastic resetting to multiple nodes alters stationary occupation probability and first-passage times, while Michelitsch (2024) establishes that certain resetting protocols can render a walk non-ergodic entirely, with mean first-passage time depending intricately on resetting rate and node proportions. Together, this work demonstrates that walk-rule modification ultimately results in an independent and powerful influence on walk statistics, distinct from structural perturbation. This motivates our study's decision to hold walk rules fixed throughout, isolating structural noise as the sole manipulated variable and avoiding the confound of simultaneously varying both structure and dynamics.

GAP

The literature on return-time distributions of random walks splits into two streams that rarely intersect. One stream derives or approximates first-return time distributions on fixed, unperturbed graphs, setting up the baseline that these distributions depend on degree variation, cycle structure, and local neighborhood connectivity (Tishby et al., 2021; Evnin & Horinouchi, 2024; Hörmann et al., 2024; Masuda & Konno, 2004). This framework has since been extended to non-backtracking walks on networks with arbitrary degree distributions, further reinforcing degree heterogeneity as a core determinant of return-time behavior (Lev-Ari et al., 2025), but still not bridging the gap in understanding the influence of random edge noise on return-time distributions. The other stream examines how graphs and walks respond to changes such as targeted attacks, edge rewiring, stochastic resetting, or continuously evolving structure, but it evaluates this through outcome measures such as mixing time, hitting time, cover time, or stationary occupation probability, rather than the complete return-time distribution itself (Kawamura et al., 2024; Bala Prinya, 2025; Chung & Robertson, 2025; Sauerwald & Zanetti, 2019). Within this stream, vertex-dependent resetting protocols have been shown to yield exact, tunable occupation probabilities and passage times at each vertex (Ye & Chen, 2022), which shows that per-vertex analysis reveals effects that would be obscure in a graph-wide average. However, this sensitivity, too, is measured through passage-time statistics rather than the full return time distribution. A similar pattern appears even outside the random walk literature. In the Szegedy quantum walk model, removing a single edge from a graph produces a measurable, scaling shift in walk behavior (Giordano & Martin-Delgado, 2025), echoing, in a different context, the broader theme that structural perturbation measurably reshapes walk dynamics.

Because of the gaps and limitations present throughout current literature, no existing work examines how full first-return time distributions respond to randomly distributed structural perturbation per starting vertex on small and large finite graphs under graduated noise dilution, the scenario in which many real-world networks change. Our study is of value because it bridges the literature on structural sensitivity already established for static graphs to the literature on perturbation methodology already established for aggregate walk metrics, which offers a new characterization of how return time distributions may destabilize under controlled, random edge noise.

METHODS

Our methods are based on mathematical modeling and computational simulations. In what follows, we describe the latter. The environment simulation is Google Colab, and the programming language is Python.

To start, we code a few example graphs. These graphs are selected to cover a large spectrum of topological and metric properties: we encode planar and non-planar graphs, as well as forests and graphs with a high cycle density. We then plot these graphs; the result of this operation is Figure 1. We will also be comparing results with RRGs (d=8, n=100), Erdős-Rényi-type Graphs (n=100, p=0.7), and the Complete Graph.

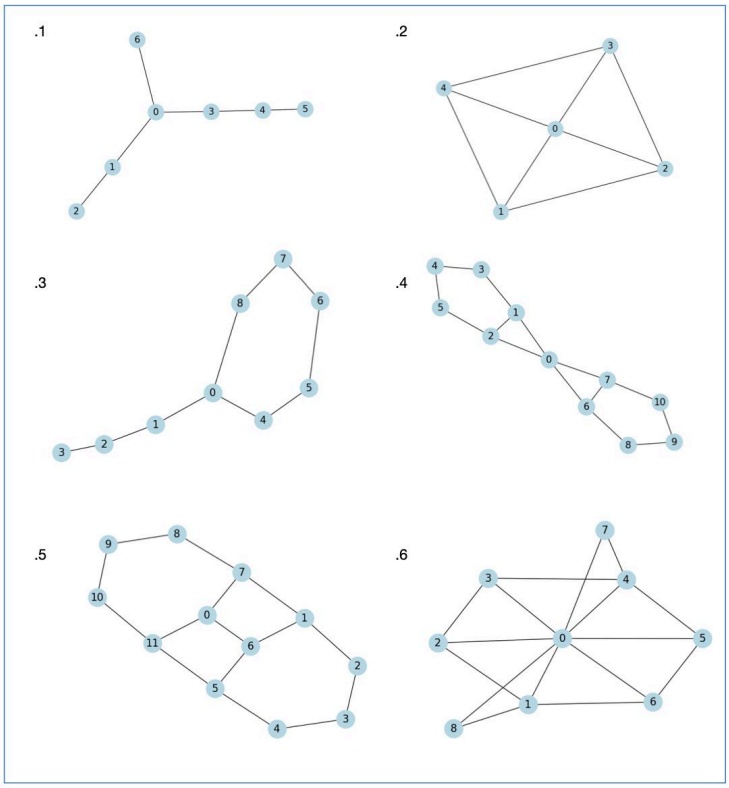

Figure 4.1: Representation of the graphs considered as examples.

1 - Tree Graph
2 - Cross Box Graph
3 - Umbrella Graph
4 - Lemniscate Graph
5 - Colliding Cycles Graph

6 - Winged Hexagon Graph

Below is the code for these graphs:

```
"Tree Graph": {
    "graph_dict": {"0": ["1", "3", "6"],
                   "1": ["0", "2"],
                   "2": ["1"],
                   "3": ["0", "4"],
                   "4": ["3", "5"],
                   "5": ["4"],
                   "6": ["0"]}
},
"Umbrella Graph": {
    "graph_dict": {"0": ["1", "4", "8"],
                   "1": ["0", "2"],
                   "2": ["1", "3"],
                   "3": ["2"],
                   "4": ["0", "5"],
                   "5": ["4", "6"],
                   "6": ["5", "7"],
                   "7": ["6", "8"],
                   "8": ["0", "7"]}
},
"Winged Hexagon Graph": {
    "graph_dict": {"0": ["1", "2", "3", "4", "5", "6", "7", "8"],
                   "1": ["0", "2", "6", "8"],
                   "2": ["0", "1", "3"],
                   "3": ["0", "2", "4"],
                   "4": ["0", "3", "5", "7"],
                   "5": ["0", "4", "6"],
                   "6": ["0", "1", "5"],
                   "7": ["0", "4"],
                   "8": ["0", "1"]}
},
```

```
"Colliding Cycles Graph": {
    "graph_dict": {"0": ["6", "7", "11"],
                   "1": ["2", "6", "7"],
                   "2": ["1", "3"],
                   "3": ["2", "4"],
                   "4": ["3", "5"],
                   "5": ["4", "6", "11"],
                   "6": ["0", "1", "5"],
                   "7": ["0", "1", "8"],
                   "8": ["7", "9"],
                   "9": ["8", "10"],
                   "10": ["9", "11"],
                   "11": ["0", "5", "10"]}
},
"Lemniscate Graph": {
    "graph_dict": {"0": ["1", "2", "6", "7"],
                   "1": ["0", "2", "3"],
                   "2": ["0", "1", "5"],
                   "3": ["1", "4"],
                   "4": ["3", "5"],
                   "5": ["2", "4"],
                   "6": ["0", "7", "8"],
                   "7": ["0", "6", "10"],
                   "8": ["6", "9"],
                   "9": ["8", "10"],
                   "10": ["9", "7"]}
},
"Cross Box Graph": {
    "graph_dict": {"0": ["1", "2", "3", "4"],
                   "1": ["0", "2", "4"],
                   "2": ["0", "1", "3"],
                   "3": ["0", "2", "4"],
                   "4": ["0", "1", "3"]}
}
```

Figure 4.2: Representation of the graphs as replicable code structure.

We then implement a function that, given a graph and a vertex, computes the return time of a simple random walk to the specified vertex. The function simply reproduces a simple and unbiased random walk for 1000 iterations, and returns the average time before the random walk arrived at the initial position. If the initial node is detected to be detached from the rest of the graph, the data will return as disconnected, so we allowed room for 50 perturbation replicates to avoid this from happening, settling on random seed 42. The individual sampled return times are then plotted to generate a distribution, where the average return time is marked. We generate a distribution for 5 randomly picked nodes.

We then implement a noise operator function that takes as input a graph and removes randomly selected edges at three different noise levels (0.0, 0.05, 0.10, 0.15, 0.20) by cycling through the entries of said graph's adjacency matrix and randomly removing edges based on noise level, performing the independent Bernouli deletion of every edge. After applying this function to our initial graphs, we compute again the return times of the updated graphs, i.e., those with the randomly removed edges, for the different starting vertices. We gather the results obtained by running these simulations of random walks, and plot the distributions of the return times for the graphs before and after the removal of the edges. To investigate the stability of return time after the small perturbation, we statistically compare the return time distribution of the graphs with randomly removed edges to the return time distributions of the original

graphs. In particular, we will compare both the distributions themselves and the standard deviations associated with them.

It is important to note that different graph perturbations are performed on many different runs of the baseline graphs, with many walks simulated to gather data. The reason as to why we do this is due to the formula we derived earlier for the theoretical return time of a simple random walker, which links degree to return times. All mean return times for the unperturbed graphs will be calculated to verify the data and debug the code using the return time formula derived in the preliminaries within stable margins of 1 standard deviation from the mean.

We assess whether the return-time distribution at each node changes under perturbation through the Anderson-Darling test. A significant result indicates that the perturbed distribution diverges meaningfully from the original, with particular sensitivity to shifts in the distribution's tail. A non-significant result indicates no detectable divergence between the original and perturbed distributions. We compute this test for each vertex, comparing its original return-time distribution to its perturbed counterpart at each noise level, allowing us to determine whether structural perturbation produces a statistically significant shift in return-time behavior at that node. If the results show p-values of less than 0.001, they will be rounded to a hard floor of 0.001, as anything less is statistically significant due to such p-values being less than any reasonable significance level. It should be noted that many tests are performed with no multiple-comparison adjustment.

We find the magnitude of change in return-time behavior through the Wasserstein distance between the original and perturbed distributions. Unlike the previous tests, the Wasserstein distance is not a significance test but an effect-size measure: a larger value indicates a greater overall shift in return-time behavior, expressed directly in return-time units, while a smaller value indicates the distribution remained close to its original shape and location. We compute this measure for each graph at each noise level, allowing us to rank graphs by the amount their return-time behavior was altered by perturbation.

In addition to the Wasserstein distance calculations, we will also calculate confidence intervals to obtain a range of possible shifts of the return time distributions post-perturbation. We do this by bootstrap, in which intervals for Wasserstein and Mean distance calculations were estimated using 500 resamples Resampling was performed with replacement on 1000 random walk return time samples for each fixed, perturbed graph, instead of generating new graph realizations.

It is important to note that the parity structure per graph dictates which return times are possible. Bipartite graphs, such as the Tree and Umbrella graphs, only generate return times which are even. Non-bipartite graphs, containing odd cycles, allow for both even and odd return times. This distinction affects our comparison of return time distributions across the different structures listed.

It should be noted that the generated RRG and Erdős-Rényi-type Graph are non-bipartite with high probability. This is justified through the odd cycle nature of RRGs and the calculated expected triangle

count (55463.1) of an Erdős-Rényi-type Graph with characteristics (n=100,p=0.7), guaranteeing odd cycles with high probability.

Walks per Vertex	Perturbation Replicates	Maximum Step Cap	Random Seed
1000	A single noisy graph simulation is simulated, with room for 50 retries to try to guarantee no disconnections to the initial vertex	10000	Seed=42

Figure 4.3: Replicability Parameters

RESULTS

CROSS BOX GRAPH

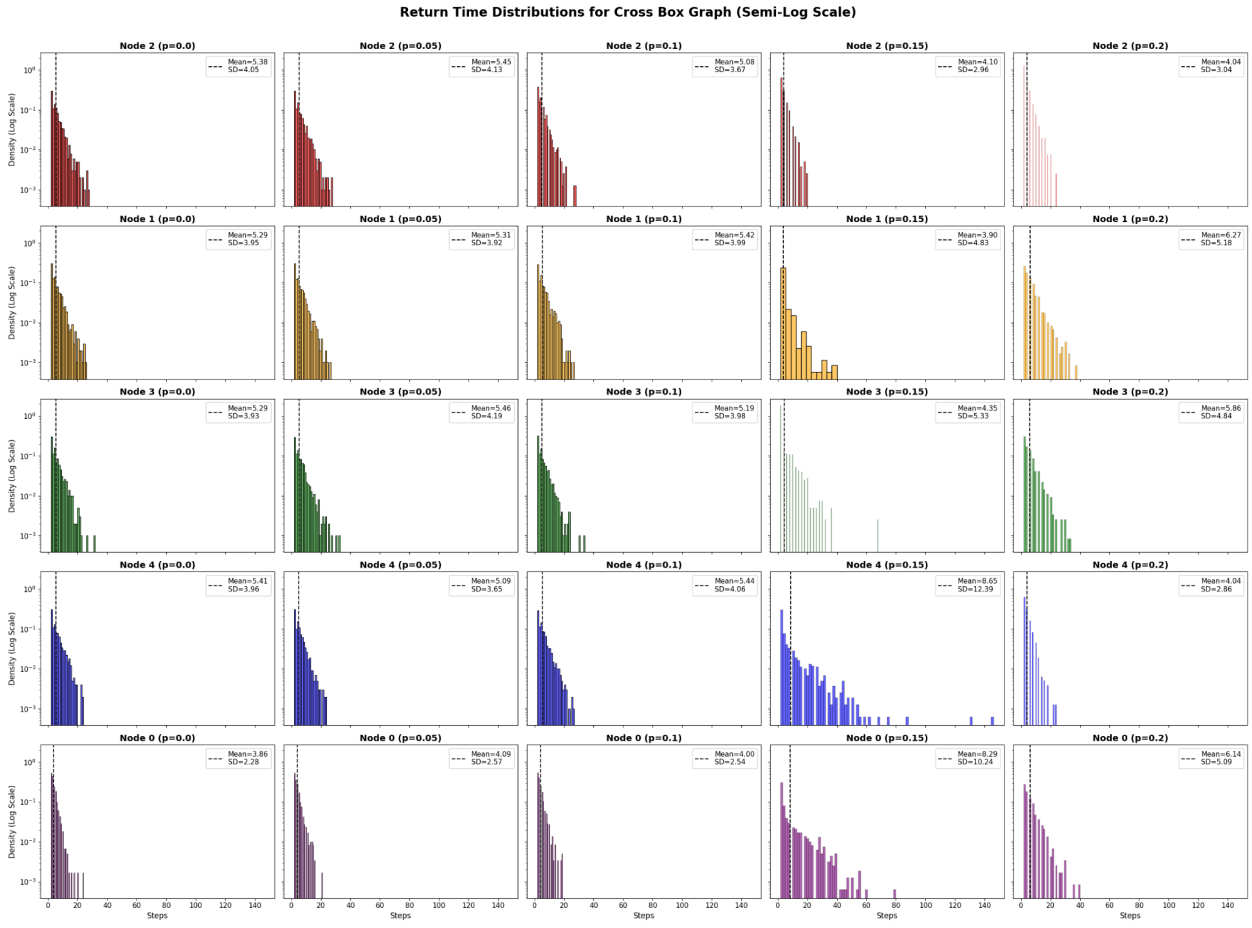

Figure 5.1.1: Return time distributions for the Cross Box Graph.

Figure 5.1.2: Table of Mean RT, and Std Dev for the Cross Box Graph:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
2 (Theoretical=5.33)	Mean=5.38 SD=4.05	Mean=5.45 SD=4.13	Mean=5.08 SD=3.67	Mean=4.10 SD=2.96	Mean=4.04 SD=3.04
1 (Theoretical=5.33)	Mean=5.29 SD=3.95	Mean=5.31 SD=3.92	Mean=5.42 SD=3.99	Mean=3.90 SD=4.83	Mean=6.27 SD=5.18
3 (Theoretical=5.33)	Mean=5.29 SD=3.93	Mean=5.46 SD=4.19	Mean=5.19 SD=3.98	Mean=4.35 SD=5.33	Mean=5.86 SD=4.84
4 (Theoretical=5.33)	Mean=5.41 SD=3.96	Mean=5.44 SD=4.06	Mean=5.44 SD=3.06	Mean=8.65 SD=12.39	Mean=4.04 SD=2.86
0 (Theoretical=4.00)	Mean=3.86 SD=2.28	Mean=4.00 SD=2.54	Mean=4.00 SD=2.54	Mean=8.29 SD=10.24	Mean=6.14 SD=5.09

Data involving the initial Mean RT (Average return time across all simulated random walks) matches the calculated theoretical return time within the stable margins.

TREE GRAPH

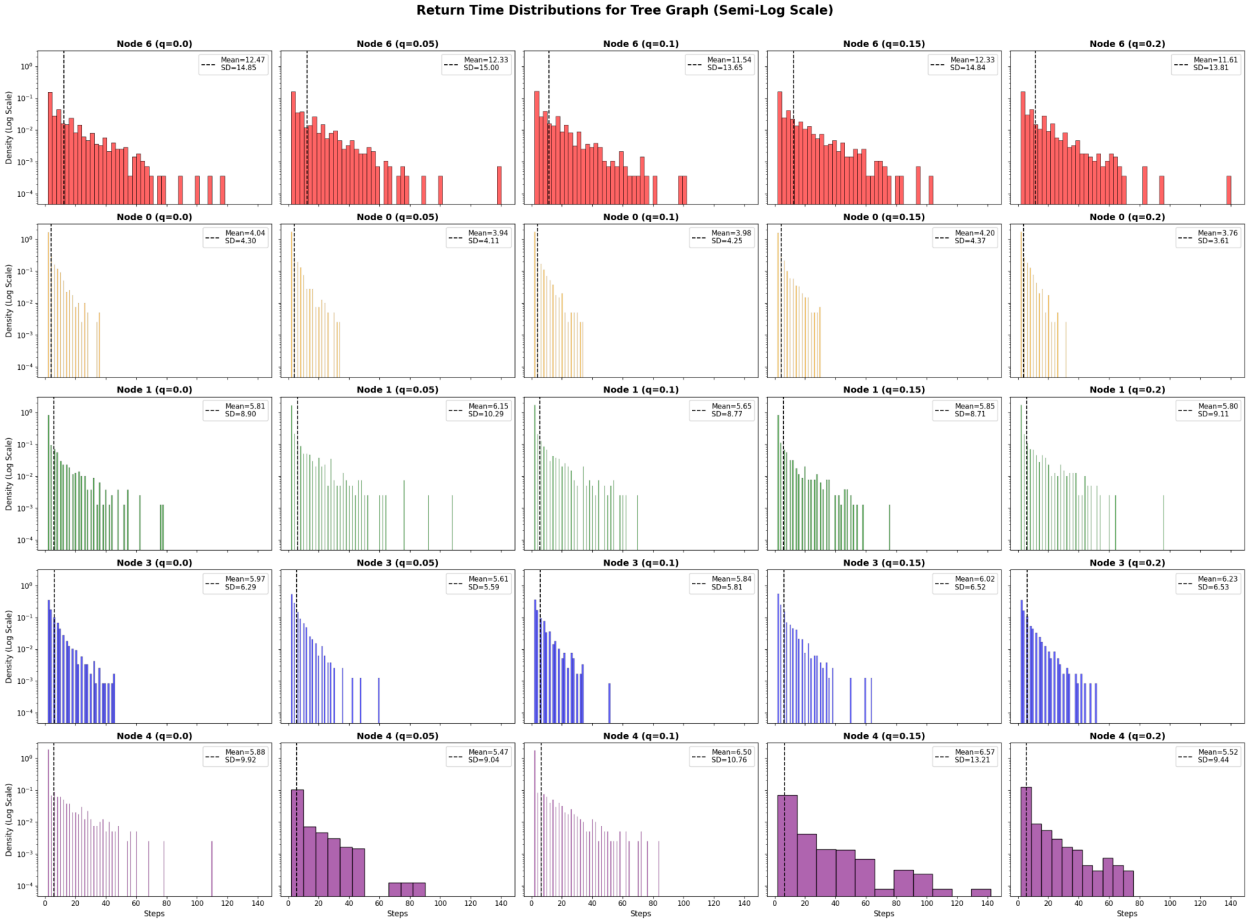

Figure 5.2.1: Return time distributions for the Tree Graph.

Figure 5.2.2: Table of Mean RT, and Std Dev for the Tree Graph:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
6 (Theoretical=12.00)	Mean=12.47 SD=14.85	Mean=12.33 SD=15.00	Mean=11.54 SD=13.65	Mean=12.33 SD=14.84	Mean=11.61 SD=13.81
0 (Theoretical=4.00)	Mean=4.04 SD=4.30	Mean=3.94 SD=4.11	Mean=3.98 SD=4.25	Mean=4.20 SD=4.37	Mean=3.76 SD=3.61
1 (Theoretical=6.00)	Mean=5.81 SD=8.90	Mean=6.15 SD=10.29	Mean=5.65 SD=8.77	Mean=5.85 SD=8.71	Mean=5.80 SD=9.11
3 (Theoretical=6.00)	Mean=5.97 SD=6.29	Mean=5.61 SD=5.59	Mean=5.84 SD=5.81	Mean=6.02 SD=6.52	Mean=6.23 SD=6.53
4 (Theoretical=6.00)	Mean=5.88 SD=9.92	Mean=5.47 SD=9.04	Mean=6.50 SD=10.76	Mean=6.57 SD=13.21	Mean=5.52 SD=9.44

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

UMBRELLA GRAPH

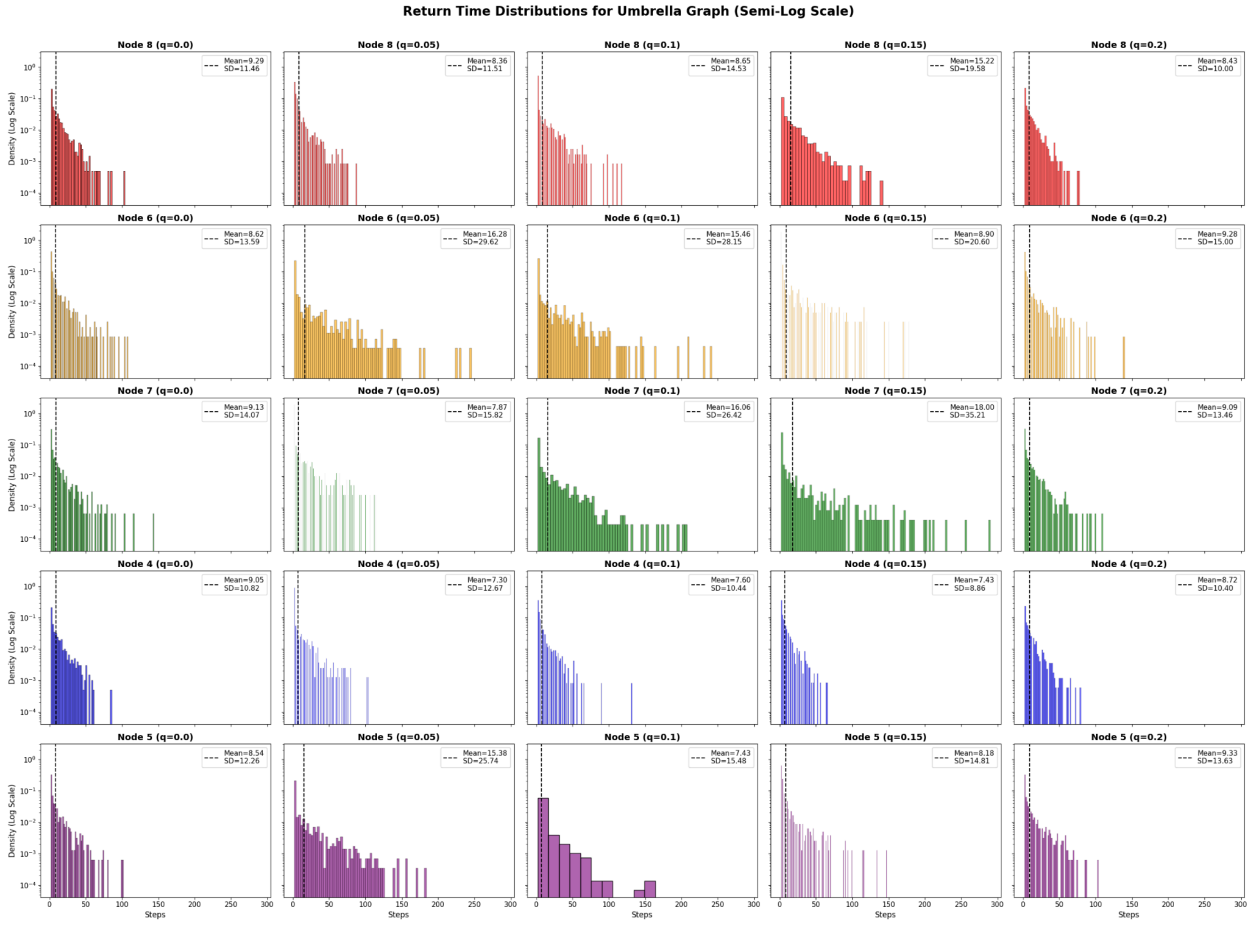

Figure 5.3.1: Return time distributions for the Umbrella Graph.

Figure 5.3.2: Table of Mean RT, and Std Dev for the Umbrella Graph:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
8 (Theoretical=9.00)	Mean=9.29 SD=11.46	Mean=8.36 SD=11.51	Mean=8.65 SD=14.53	Mean=15.22 SD=19.58	Mean=8.43 SD=10.00
6 (Theoretical=9.00)	Mean=8.62 SD=13.59	Mean=16.28 SD=29.62	Mean=15.46 SD=28.15	Mean=8.90 SD=20.60	Mean=9.28 SD=15.00
7 (Theoretical=9.00)	Mean=9.13 SD=14.07	Mean=7.87 SD=15.82	Mean=16.06 SD=26.42	Mean=18.00 SD=35.21	Mean=9.09 SD=13.46
4 (Theoretical=9.00)	Mean=9.05 SD=10.82	Mean=7.30 SD=12.67	Mean=7.60 SD=10.44	Mean=7.43 SD=8.86	Mean=8.72 SD=10.40
5 (Theoretical=9.00)	Mean=8.54 SD=12.26	Mean=15.38 SD=25.74	Mean=7.43 SD=15.48	Mean=8.18 SD=14.81	Mean=9.33 SD=13.63

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

WINGED HEXAGON GRAPH

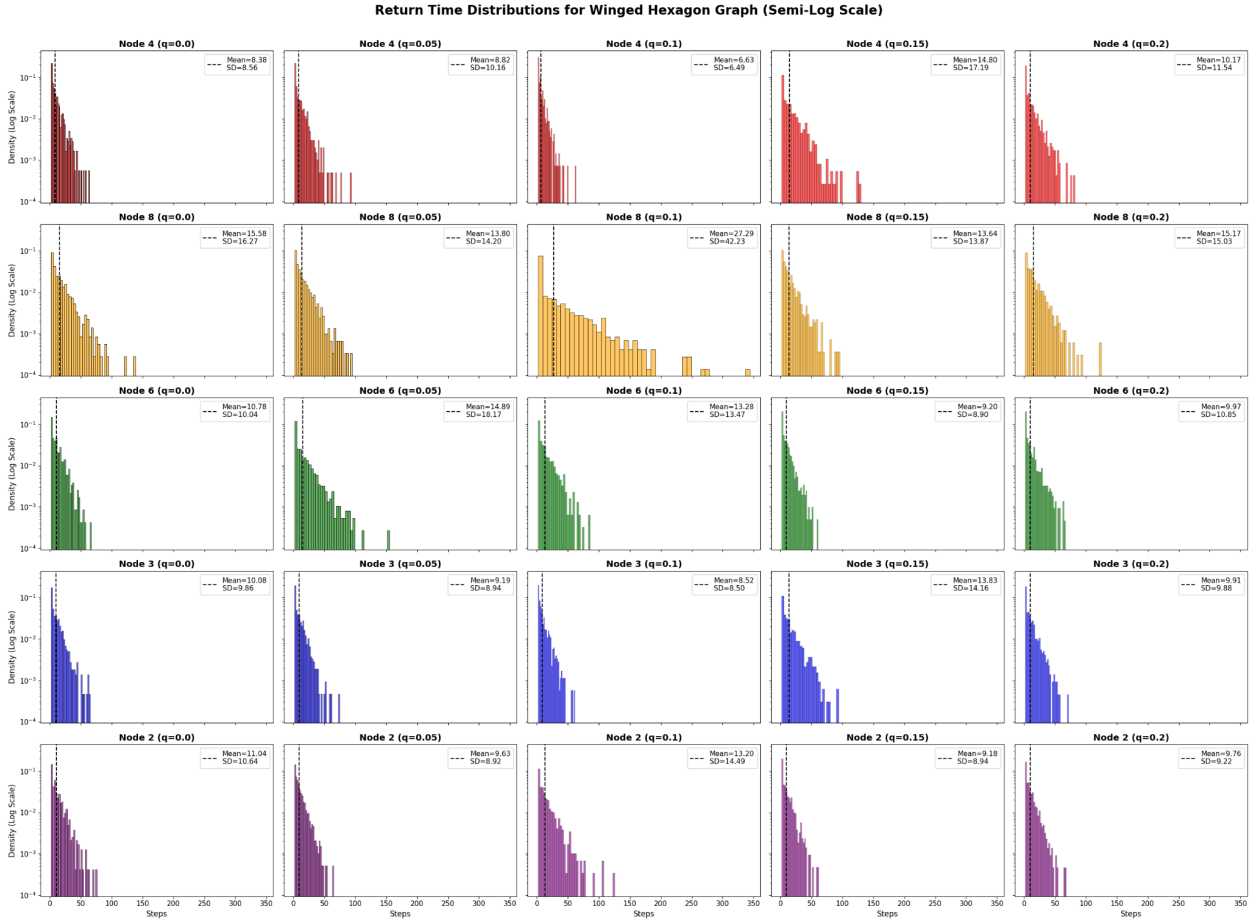

Figure 5.4.1: Return time distributions for the Winged Hexagon Graph.

Figure 5.4.2: Table of Mean RT, and Std Dev for the Winged Hexagon Graph:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
4 (Theoretical=8.00)	Mean=8.38 SD=8.56	Mean=8.82 SD=10.16	Mean=6.63 SD=6.49	Mean=14.80 SD=17.19	Mean=10.17 SD=11.54
8 (Theoretical=16.00)	Mean=15.58 SD=16.27	Mean=13.80 SD=14.20	Mean=27.29 SD=42.23	Mean=13.64 SD=13.87	Mean=15.17 SD=15.03
6 (Theoretical=10.67)	Mean=10.78 SD=10.04	Mean=14.89 SD=18.17	Mean=13.28 SD=13.47	Mean=9.20 SD=8.90	Mean=9.97 SD=10.85
3 (Theoretical=10.67)	Mean=10.08 SD=9.86	Mean=9.19 SD=8.94	Mean=8.52 SD=8.50	Mean=13.83 SD=14.16	Mean=9.91 SD=9.88
2 (Theoretical=10.67)	Mean=11.04 SD=10.64	Mean=9.63 SD=8.92	Mean=13.20 SD=14.49	Mean=9.18 SD=8.94	Mean=9.76 SD=9.22

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

COLLIDING CYCLES GRAPH

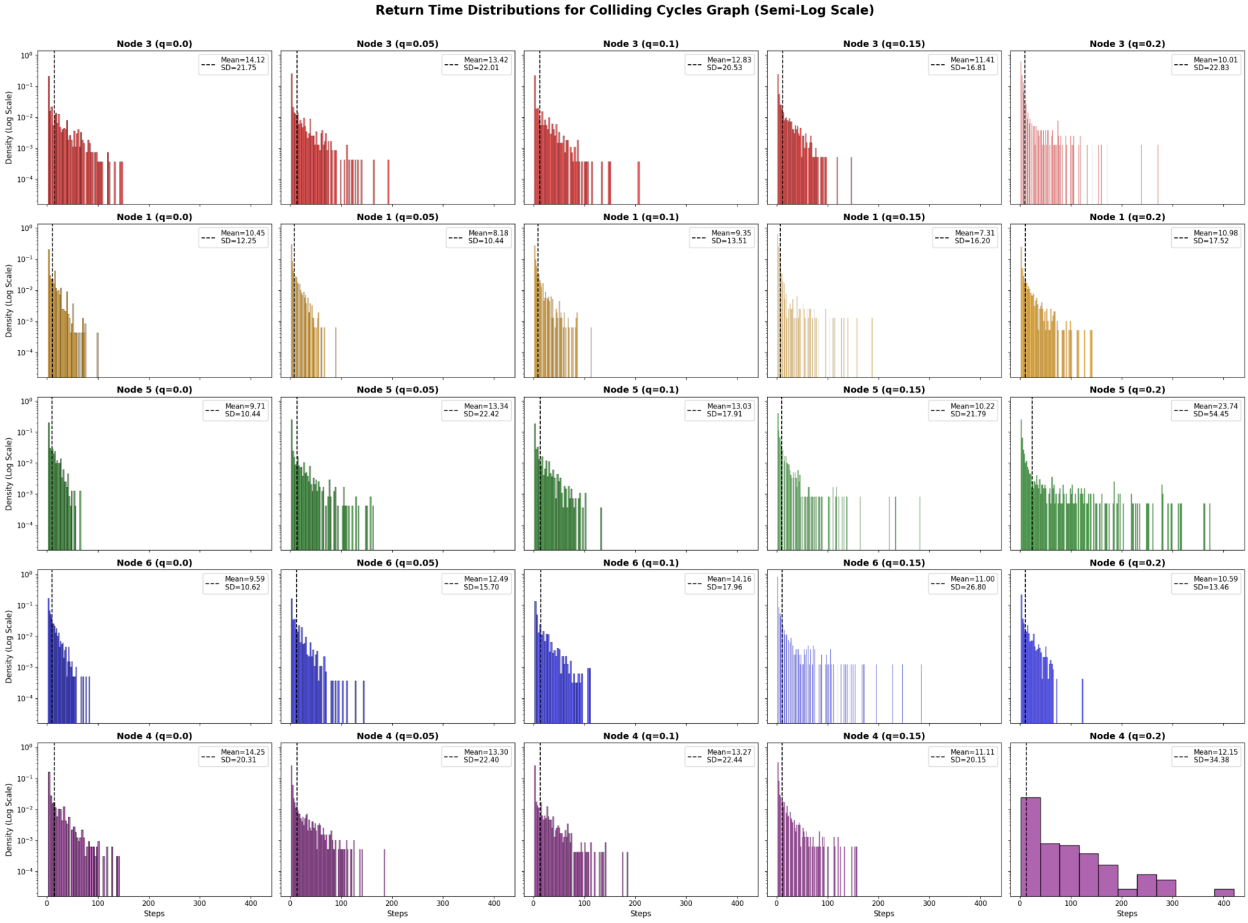

Figure 5.5.1: Return time distributions for the Colliding Cycles Graph.

Figure 5.5.2: Table of Mean RT, and Std Dev for the Colliding Cycles Graph:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
3 (Theoretical=15.00)	Mean=14.12 SD=21.75	Mean=13.42 SD=22.01	Mean=12.83 SD=20.53	Mean=11.41 SD=16.81	Mean=10.01 SD=22.83
1 (Theoretical=10.00)	Mean=10.45 SD=12.25	Mean=8.18 SD=10.44	Mean=9.35 SD=13.51	Mean=7.31 SD=16.20	Mean=10.98 SD=17.52
5 (Theoretical=10.00)	Mean=9.71 SD=10.44	Mean=13.34 SD=22.42	Mean=13.03 SD=17.91	Mean=10.22 SD=21.79	Mean=23.74 SD=54.45
6 (Theoretical=10.00)	Mean=9.59 SD=10.62	Mean=12.49 SD=15.70	Mean=14.16 SD=17.96	Mean=11.00 SD=26.80	Mean=10.59 SD=13.46
4 (Theoretical=15.00)	Mean=14.25 SD=20.31	Mean=13.30 SD=22.40	Mean=13.27 SD=22.44	Mean=11.11 SD=20.15	Mean=12.15 SD=34.38

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

LEMNISCATE GRAPH

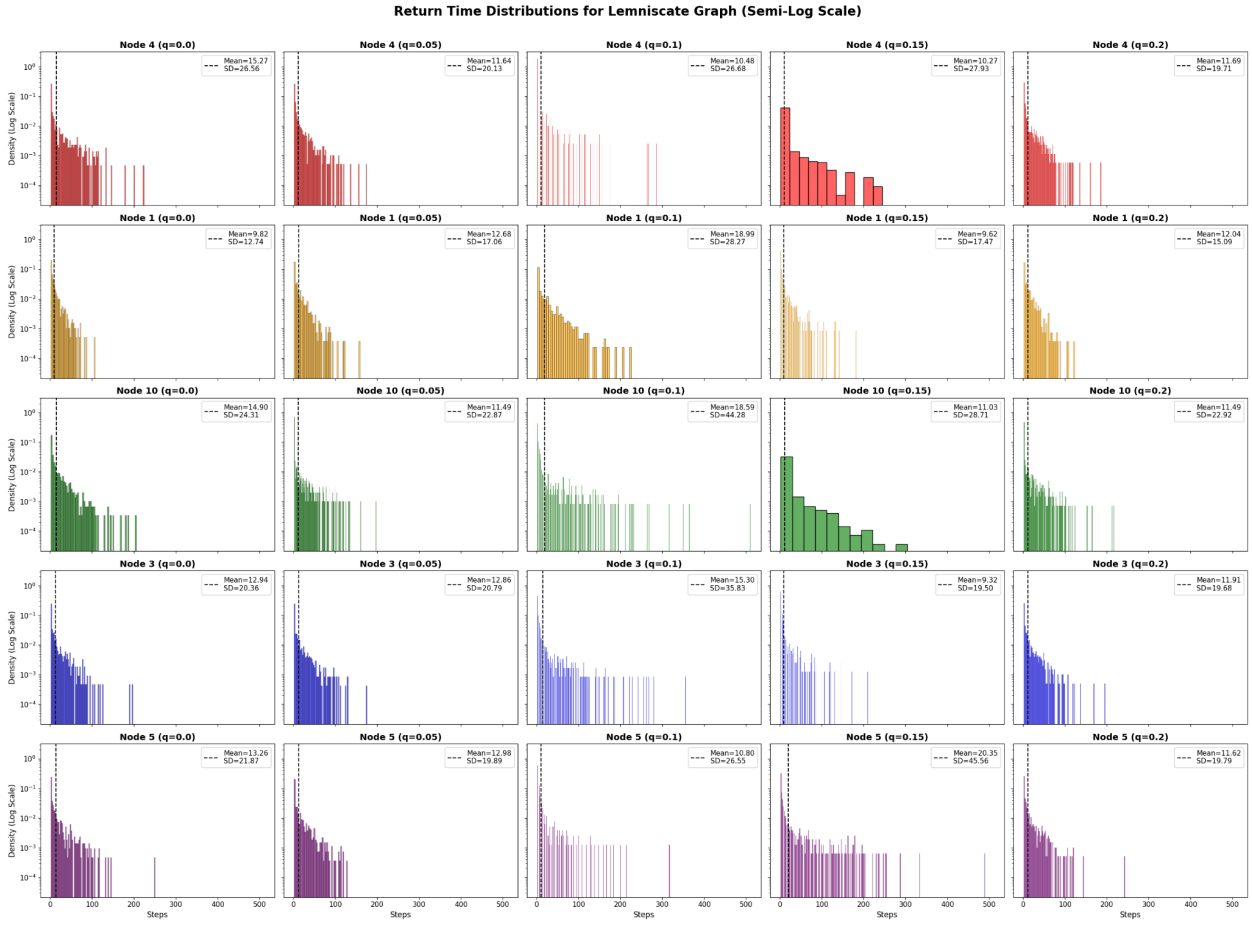

Figure 5.6.1: Return time distributions for the Lemniscate Graph.

Figure 5.6.2: Table of Mean RT, and Std Dev for the Lemniscate Graph:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
4 (Theoretical=14.00)	Mean=15.27 SD=26.56	Mean=11.64 SD=20.13	Mean=10.48 SD=26.68	Mean=10.27 SD=27.93	Mean=11.69 SD=19.71
1 (Theoretical=9.33)	Mean=9.82 SD=12.74	Mean=12.68 SD=17.06	Mean=18.99 SD=28.27	Mean=9.62 SD=17.47	Mean=12.04 SD=15.09
10 (Theoretical=14.00)	Mean=14.90 SD=24.31	Mean=11.49 SD=22.87	Mean=18.59 SD=44.28	Mean=11.03 SD=28.71	Mean=11.49 SD=22.92
3 (Theoretical=14.00)	Mean=12.94 SD=20.36	Mean=12.86 SD=20.79	Mean=15.30 SD=35.83	Mean=9.32 SD=19.50	Mean=11.91 SD=19.68
5 (Theoretical=14.00)	Mean=13.26 SD=21.87	Mean=12.98 SD=19.89	Mean=10.80 SD=26.55	Mean=20.35 SD=45.56	Mean=11.62 SD=19.79

COMPLETE GRAPH (K5)

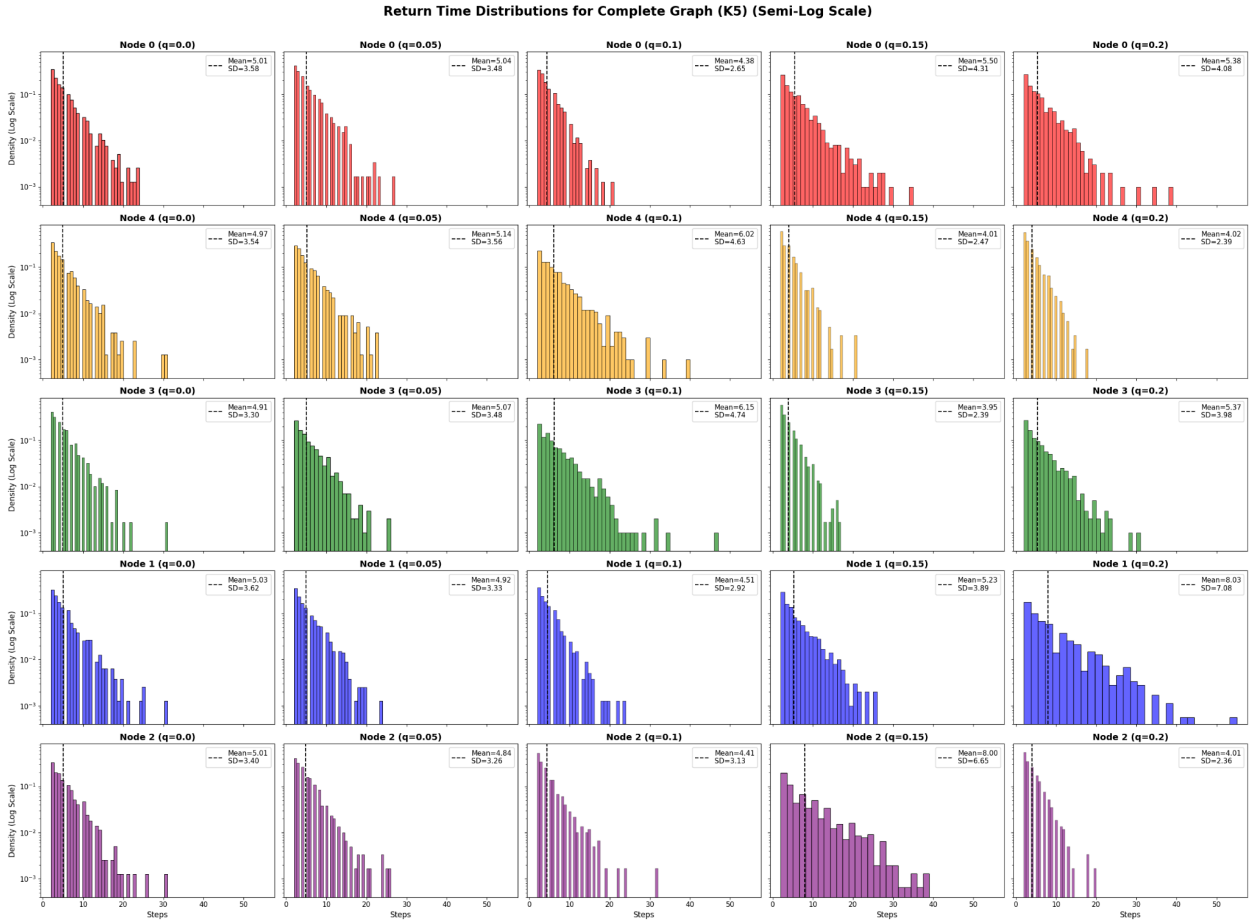

Figure 5.7.1: Return time distributions for the Complete Graph.

Figure 5.7.2: Table of Mean RT, and Std Dev for the Complete Graph (All theoretical return times on K5 are equal to 5):

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
0	Mean=5.01 SD=3.58	Mean=5.04 SD=3.48	Mean=4.38 SD=2.65	Mean=5.50 SD=4.31	Mean=5.38 SD=4.08
4	Mean=4.97 SD=3.54	Mean=5.14 SD=3.56	Mean=6.02 SD=4.63	Mean=4.01 SD=2.47	Mean=4.02 SD=2.39
3	Mean=4.91 SD=3.30	Mean=5.07 SD=3.48	Mean=6.15 SD=4.74	Mean=3.95 SD=2.39	Mean=5.37 SD=3.98
1	Mean=5.03 SD=3.62	Mean=4.92 SD=3.33	Mean=4.51 SD=2.92	Mean=5.23 SD=3.89	Mean=8.03 SD=7.08
2	Mean=5.01 SD=3.40	Mean=4.84 SD=3.26	Mean=4.41 SD=3.13	Mean=8.00 SD=6.65	Mean=4.01 SD=2.36

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

ERDŐS-RÉNYI-TYPE GRAPHS

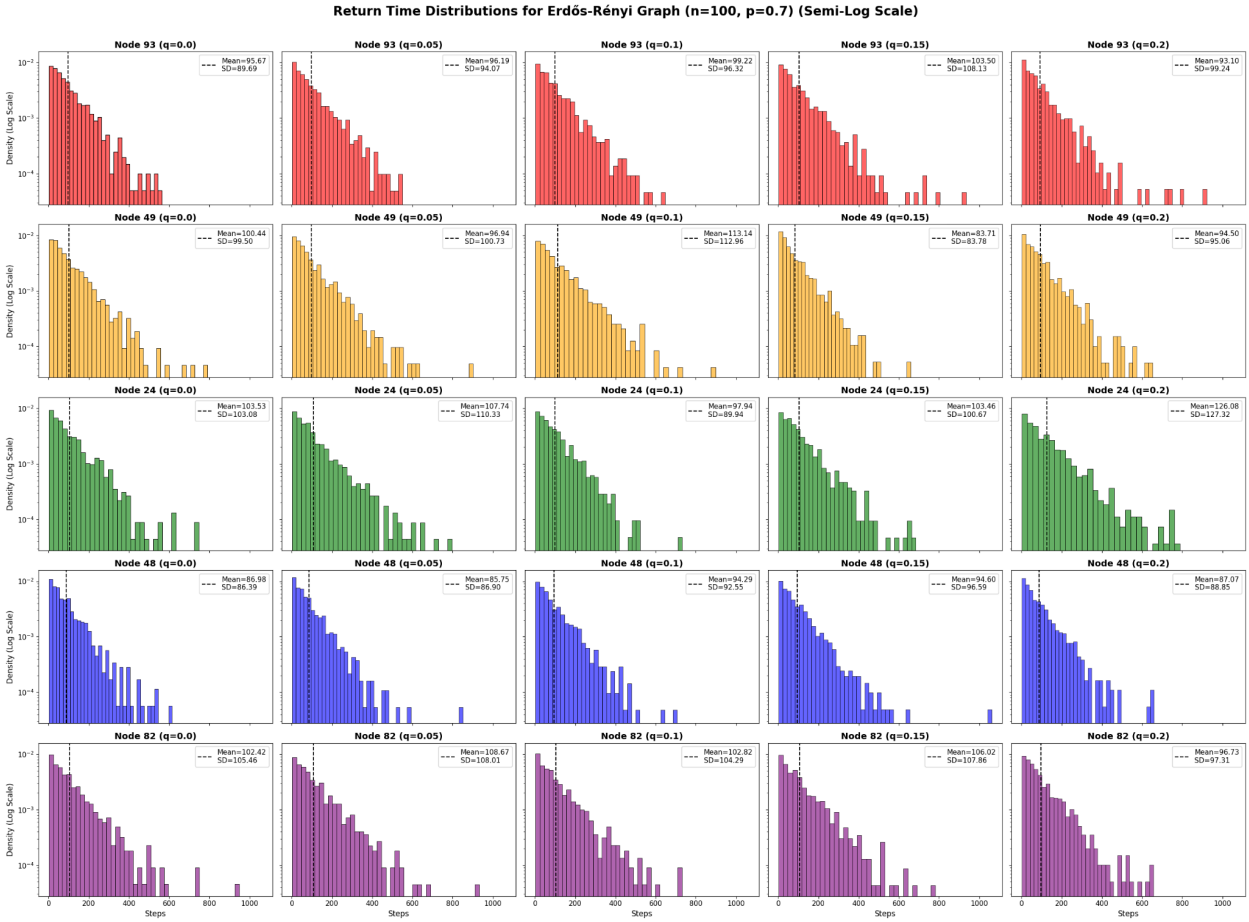

Figure 5.8.1: Return time distributions for the Erdős-Rényi-type Graphs.

Figure 5.8.2: Table of Mean RT, and Std Dev for the Erdős-Rényi-type Graphs:

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
93 (Theoretical=95.48)	Mean=95.67 SD=89.69	Mean=96.19 SD=94.07	Mean=99.22 SD=96.32	Mean=103.50 SD=108.13	Mean=93.10 SD=89.69
49 (Theoretical=98.17)	Mean=100.44 SD=99.50	Mean=96.94 SD=100.73	Mean=113.14 SD=112.96	Mean=83.71 SD=83.78	Mean=94.50 SD=95.06
24 (Theoretical=104.03)	Mean=103.53 SD=103.08	Mean=107.74 SD=110.33	Mean=97.94 SD=89.94	Mean=103.46 SD=100.67	Mean=126.08 SD=127.32
48 (Theoretical=91.71)	Mean=86.98 SD=86.39	Mean=85.75 SD=86.90	Mean=94.29 SD=92.55	Mean=94.60 SD=96.59	Mean=87.07 SD=88.85
82 (Theoretical=102.50)	Mean=102.42 SD=105.46	Mean=108.67 SD=108.01	Mean=102.82 SD=104.29	Mean=106.02 SD=107.86	Mean=96.73 SD=97.31

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

RANDOM REGULAR GRAPHS (RRG)

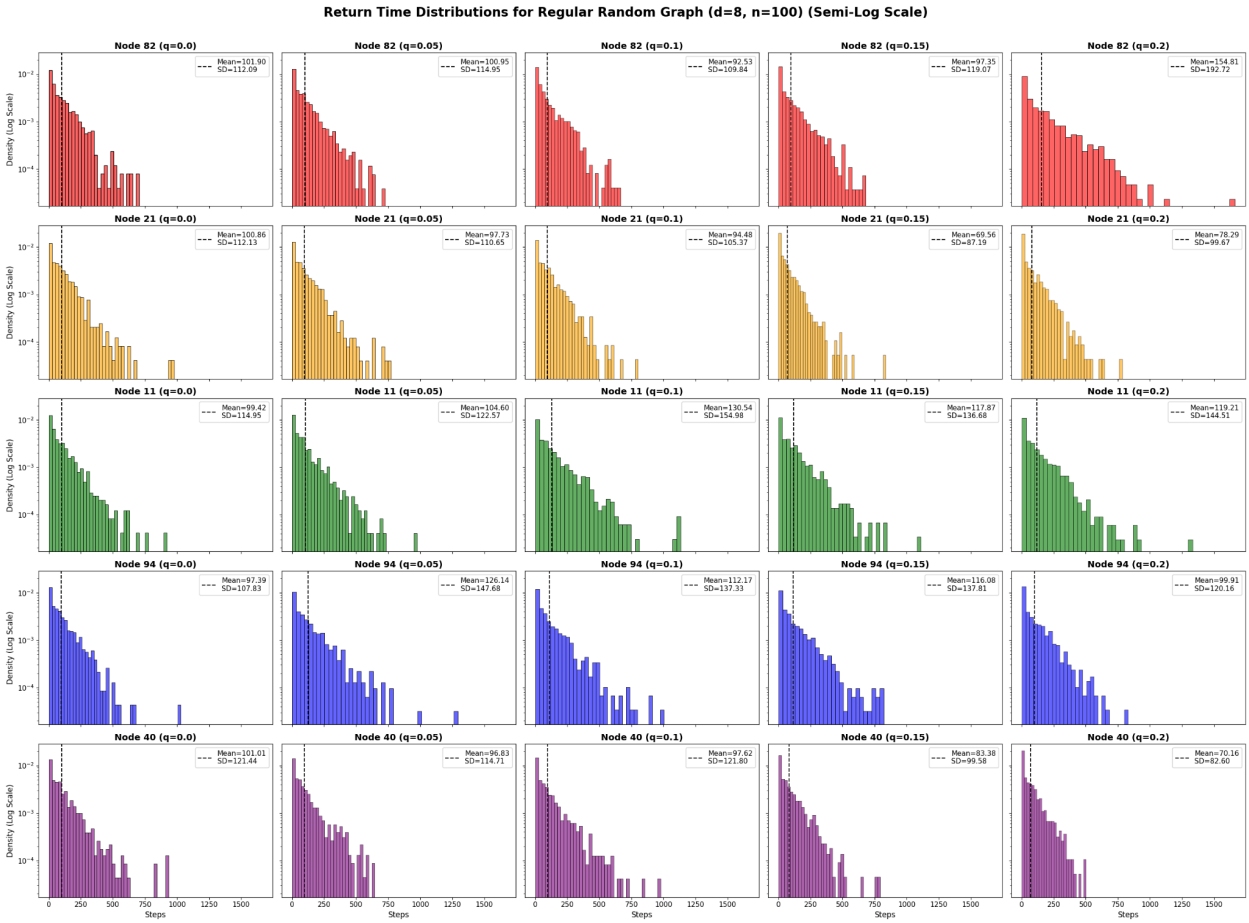

Figure 5.9.1: Return time distributions for the Random Regular Graphs.

Figure 5.9.2: Table of Mean RT, and Std Dev for the Random Regular Graphs(All theoretical return times on RRG(d=8, n=100) are equal to 100):

Vertex	q=0.00	q=0.05	q=0.10	q=0.15	q=0.20
82	Mean=101.90 SD=112.09	Mean=100.95 SD=114.95	Mean=92.53 SD=109.84	Mean=97.35 SD=119.07	Mean=154.81 SD=192.72
21	Mean=100.86 SD=112.13	Mean=97.73 SD=110.65	Mean=94.48 SD=105.37	Mean=69.56 SD=87.19	Mean=78.29 SD=99.67
11	Mean=99.42 SD=114.95	Mean=104.60 SD=122.57	Mean=130.54 SD=154.98	Mean=117.87 SD=136.68	Mean=119.21 SD=144.51
94	Mean=97.39 SD=107.83	Mean=126.14 SD=147.68	Mean=112.17 SD=137.33	Mean=116.08 SD=137.81	Mean=99.91 SD=120.16
40	Mean=101.01 SD=121.44	Mean=96.83 SD=114.71	Mean=97.62 SD=121.80	Mean=83.38 SD=99.58	Mean=70.16 SD=82.60

Data involving the initial Mean RT matches the calculated theoretical return time within the stable margins.

*Note: Due to the properties of RRGs, every vertex behaves identically in terms of its degree, so return time distributions behave the same regardless of the selected initial vertex, and using 5 randomly selected starting vertices is merely for replication purposes.

ANALYSIS

ANDERSON-DARLING K-SAMPLE TEST (NORMALIZED)

The null hypothesis (H_0) is that the original and perturbed return-time distributions come from the same underlying distribution. The alternative (H_1) is that they differ. The Anderson-Darling test is appropriate given the expected heavy-tailed nature of return-time distributions under structural perturbation. We are performing this test on data collected from thousands of walks from samples of a population of randomly perturbed graphs. Significance was tested at $\alpha = 0.05$.

*Note, p will be defined in this section as the p-value for the AD Test, not to be confused with the edge probability of the generated Erdős-Rényi-type graphs

Figure 6.1.1: Starting Vertex 1 Test Table:

Graph	P-Value + Normalized k-Sample Test Statistic (q=0.05)	P-Value + Normalized k-Sample Test Statistic (q=0.10)	P-Value + Normalized k-Sample Test Statistic (q=0.15)	P-Value + Normalized k-Sample Test Statistic (q=0.20)
Complete Graph (K5)	p>0.250 k=-0.619	p<0.001 k=6.796	p=0.038 k=0.159	p=0.083 k=1.443
Regular Random Graphs (RRGs)	p=0.077 k=1.515	p=0.001 k=6.187	p<0.001 k=16.668	p<0.001 k=31.655
Erdős-Rényi-type Graphs	p=0.224 k=0.419	p>0.250 k=-0.457	p=0.224 k=0.417	p=0.085 k=1.417
Tree Graph	p>0.250 k=-1.093	p>0.250 k=0.026	p>0.250 k=-1.039	p>0.250 k=-0.452
Umbrella Graph	p=0.003 k=5.316	p<0.001 k=98.795	p<0.001 k=39.317	p>0.250 k=0.253
Winged Hexagon Graph	p=0.046 k=2.068	p<0.001 k=11.448	p<0.001 k=49.009	p=0.002 k=5.760
Colliding Cycles Graph	p>0.250 k=-0.761	p>0.250 k=-0.337	p=0.050 k=1.973	p<0.001 k=21.061
Lemniscate Graph	p=0.010 k=3.727	p<0.001 k=131.605	p<0.001 k=164.054	p=0.020 k=2.968
Cross Box Graph	p>0.250 k=-1.045	p>0.250 k=0.122	p<0.001 k=74.712	p<0.001 k=86.313

Figure 6.1.2: Starting Vertex 2 Test Table:

Graph	P-Value + Normalized k-Sample Test Statistic (q=0.05)	P-Value + Normalized k-Sample Test Statistic (q=0.10)	P-Value + Normalized k-Sample Test Statistic (q=0.15)	P-Value + Normalized k-Sample Test Statistic (q=0.20)
Complete Graph (K5)	p=0.119 k=1.065	p<0.001 k=18.542	p<0.001 k=29.924	p<0.001 k=27.404
Regular Random Graphs (RRGs)	p>0.250 k=-0.538	p=0.176 k=0.664	p<0.001 k=36.035	p<0.001 k=31.254
Erdős-Rényi-type Graphs	p>0.250 k=0.159	p=0.032 k=2.440	p<0.001 k=10.394	p=0.090 k=1.350
Tree Graph	p>0.250 k=-0.977	p>0.250 k=-0.235	p=0.240 k=0.348	p>0.250 k=-0.030
Umbrella Graph	p<0.001 k=12.952	p<0.001 k=11.259	p<0.001 k=109.953	p>0.250 k=-0.583
Winged Hexagon Graph	p=0.028 k=2.607	p<0.001 k=129.524	p=0.013 k=3.419	p>0.250 k=0.323
Colliding Cycles Graph	p<0.001 k=27.417	p<0.001 k=13.284	p<0.001 k=89.451	p<0.001 k=26.194
Lemniscate Graph	p<0.001 k=7.371	p<0.001 k=31.552	p<0.001 k=42.011	p<0.001 k=6.752
Cross Box Graph	p>0.250 k=-1.250	p>0.250 k=-0.347	p<0.001 k=369.608	p<0.001 k=10.951

Figure 6.1.3: Starting Vertex 3 Test Table:

Graph	P-Value + Normalized k-Sample Test Statistic (q=0.05)	P-Value + Normalized k-Sample Test Statistic (q=0.10)	P-Value + Normalized k-Sample Test Statistic (q=0.15)	P-Value + Normalized k-Sample Test Statistic (q=0.20)
Complete Graph (K5)	p=0.235 k=0.369	p<0.001 k=23.72	p<0.001 k=40.374	p=0.011 k=3.641
Regular Random Graphs (RRGs)	p>0.250 k=-0.538	p<0.001 k=10.533	p=0.003 k=5.174	p<0.001 k=9.393
Erdős-Rényi-type Graphs	p>0.250 k=-0.541	p=0.246 k=0.326	p>0.250 k=0.033	p<0.001 k=8.586
Tree Graph	p>0.250 k=-1.205	p>0.250 k=-0.729	p>0.250 k=-0.884	p>0.250 k=-0.400
Umbrella Graph	p<0.001 k=125.844	p<0.001 k=16.781	p<0.001 k=10.815	p>0.250 k=-1.228
Winged Hexagon Graph	p<0.001 k=40.074	p<0.001 k=7.449	p<0.001 k=10.590	p<0.001 k=14.074
Colliding Cycles Graph	p<0.001 k=25.780	p<0.001 k=9.013	p<0.001 k=35.862	p<0.001 k=46.637
Lemniscate Graph	p<0.001 k=134.044	p<0.001 k=24.539	p<0.001 k=223.525	p<0.001 k=101.053
Cross Box Graph	p>0.250 k=-0.604	p>0.250 k=-0.696	p<0.001 k=286.941	p=0.002 k=5.702

Figure 6.1.4: Starting Vertex 4 Test Table:

Graph	P-Value + Normalized k-Sample Test Statistic (q=0.05)	P-Value + Normalized k-Sample Test Statistic (q=0.10)	P-Value + Normalized k-Sample Test Statistic (q=0.15)	P-Value + Normalized k-Sample Test Statistic (q=0.20)
Complete Graph (K5)	p>0.250 k=-0.375	p=0.004 k=4.855	p=0.101 k=1.235	p<0.001 k=74.307
Regular Random Graphs (RRGs)	p<0.001 k=8.886	p=0.007 k=4.117	p=0.001 k=6.099	p<0.001 k=12.732
Erdős-Rényi-type Graphs	p>0.250 k=-0.871	p=0.129 k=0.978	p=0.153 k=0.808	p>0.250 k=-1.076
Tree Graph	p>0.250 k=-0.742	p>0.250 k=-1.191	p>0.250 k=-0.861	p>0.250 k=-0.678
Umbrella Graph	p<0.001 k=142.633	p<0.001 k=7.152	p=0.003 k=5.060	p>0.250 k=-0.173
Winged Hexagon Graph	p=0.025 k=2.733	p<0.001 k=11.976	p<0.001 k=20.825	p>0.250 k=-0.335
Colliding Cycles Graph	p<0.001 k=7.243	p<0.001 k=16.054	p<0.001 k=178.738	p<0.001 k=7.738
Lemniscate Graph	p=0.004 k=4.927	p<0.001 k=25.819	p<0.001 k=33.035	p<0.001 k=9.750
Cross Box Graph	p=0.114 k=1.103	p>0.250 k=-0.544	p<0.001 k=59.026	p<0.001 k=71.672

Figure 6.1.5: Starting Vertex 5 Test Table:

Graph	P-Value + Normalized k-Sample Test Statistic (q=0.05)	P-Value + Normalized k-Sample Test Statistic (q=0.10)	P-Value + Normalized k-Sample Test Statistic (q=0.15)	P-Value + Normalized k-Sample Test Statistic (q=0.20)
Complete Graph (K5)	p>0.250 k=0.131	p<0.001 k=16.822	p<0.001 k=81.179	p<0.001 k=34.879
Regular Random Graphs (RRGs)	p>0.250 k=-0.235	p=0.030 k=2.508	p<0.001 k=6.950	p<0.001 k=24.583
Erdős-Rényi-type Graphs	p=0.243 k=0.335	p>0.250 k=-1.076	p>0.250 k=-0.796	p>0.250 k=0.017
Tree Graph	p>0.250 k=-0.196	p=0.088 k=1.380	p>0.250 k=-0.849	p>0.250 k=-0.855
Umbrella Graph	p<0.001 k=13.761	p<0.001 k=120.790	p=0.021 k=2.928	p>0.250 k=-0.747
Winged Hexagon Graph	p=0.012 k=3.554	p=0.008 k=4.059	p<0.001 k=10.856	p=0.019 k=3.017
Colliding Cycles Graph	p<0.001 k=23.901	p<0.001 k=22.666	p<0.001 k=30.549	p<0.001 k=256.694
Lemniscate Graph	p=0.003 k=4.994	p<0.001 k=22.804	p<0.001 k=14.837	p<0.001 k=12.515
Cross Box Graph	p=0.142 k=0.884	p>0.250 k=-0.493	p<0.001 k=99.558	p<0.001 k=99.018

WASSERSTEIN DISTANCE

*Note: this is not a significance test, so no conditions or hypotheses will be shown

Figure 6.2.1: Starting Vertex 1 Wasserstein Table:

Graph	Shift and Mean Difference (q=0.05)	Shift and Mean Difference (q=0.10)	Shift and Mean Difference (q=0.15)	Shift and Mean Difference (q=0.20)
Complete Graph (K5)	Shift=0.128 MD=0.032	Shift=0.622 MD=-0.622	Shift=0.494 MD=0.494	Shift=0.383 MD=0.379
Regular Random Graphs (RRGs)	Shift=4.827 MD=-0.951	Shift=10.040 MD=-9.374	Shift=10.085 MD=-4.555	Shift=56.899 MD=52.913
Erdős-Rényi-type Graphs	Shift=3.958 MD=0.530	Shift=4.593 MD=3.555	Shift=9.166 MD=7.840	Shift=6.736 MD=-2.562
Tree Graph	Shift=0.412 MD=-0.136	Shift=0.954 MD=-0.926	Shift=0.356 MD=-0.136	Shift=0.932 MD=-0.864
Umbrella Graph	Shift=1.244 MD=-0.932	Shift=2.410 MD=-0.638	Shift=5.928 MD=5.928	Shift=0.860 MD=-0.860
Winged Hexagon Graph	Shift=0.823 MD=0.437	Shift=1.751 MD=-1.751	Shift=6.419 MD=6.419	Shift=1.906 MD=1.788
Colliding Cycles Graph	Shift=1.040 MD=-0.700	Shift=1.486 MD=-1.294	Shift=2.790 MD=-2.710	Shift=5.218 MD=-4.110
Lemniscate Graph	Shift=3.630 MD=-3.630	Shift=5.730 MD=-4.784	Shift=6.352 MD=-4.994	Shift=3.583 MD=-3.583
Cross Box Graph	Shift=0.135 MD=0.075	Shift=0.293 MD=-0.291	Shift=1.271 MD=-1.271	Shift=1.331 MD=-1.331

Figure 6.2.2: Starting Vertex 2 Wasserstein Table:

Graph	Shift and Mean Difference (q=0.05)	Shift and Mean Difference (q=0.10)	Shift and Mean Difference (q=0.15)	Shift and Mean Difference (q=0.20)
Complete Graph (K5)	Shift=0.207 MD=0.173	Shift=1.048 MD=1.048	Shift=0.953 MD=-0.953	Shift=0.950 MD=-0.950
Regular Random Graphs (RRGs)	Shift=4.503 MD=-3.127	Shift=6.900 MD=-6.380	Shift=31.300 MD=-31.300	Shift=22.569 MD=-22.569
Erdős-Rényi-type Graphs	Shift=4.962 MD=-3.494	Shift=12.936 MD=12.702	Shift=16.729 MD=-16.729	Shift=6.260 MD=-5.936
Tree Graph	Shift=0.130 MD=-0.106	Shift=0.120 MD=-0.064	Shift=0.220 MD=0.160	Shift=0.284 MD=-0.284
Umbrella Graph	Shift=7.654 MD=7.654	Shift=6.836 MD=6.836	Shift=3.324 MD=0.280	Shift=0.674 MD=0.654
Winged Hexagon Graph	Shift=1.808 MD=-1.782	Shift=14.770 MD=11.706	Shift=1.998 MD=-1.940	Shift=0.884 MD=-0.408
Colliding Cycles Graph	Shift=2.266 MD=-2.266	Shift=1.816 MD=-1.096	Shift=5.000 MD=-3.140	Shift=2.640 MD=0.528
Lemniscate Graph	Shift=2.894 MD=2.860	Shift=9.171 MD=9.171	Shift=2.205 MD=-0.207	Shift=2.238 MD=2.222
Cross Box Graph	Shift=0.073 MD=0.019	Shift=0.188 MD=0.132	Shift=1.762 MD=-1.390	Shift=1.002 MD=0.978

Figure 6.2.3: Starting Vertex 3 Wasserstein Table:

Graph	Shift and Mean Difference (q=0.05)	Shift and Mean Difference (q=0.10)	Shift and Mean Difference (q=0.15)	Shift and Mean Difference (q=0.20)
Complete Graph (K5)	Shift=0.226 MD=0.158	Shift=1.237 MD=1.237	Shift=0.959 MD=-0.959	Shift=0.509 MD=0.455
Regular Random Graphs (RRGs)	Shift=5.959 MD=5.183	Shift=31.317 MD=31.129	Shift=18.962 MD=18.458	Shift=22.091 MD=19.791
Erdős-Rényi-type Graphs	Shift=5.127 MD=4.211	Shift=7.588 MD=-5.588	Shift=4.881 MD=-0.065	Shift=22.551 MD=22.547
Tree Graph	Shift=0.398 MD=0.346	Shift=0.236 MD=-0.156	Shift=0.202 MD=0.046	Shift=0.318 MD=-0.010
Umbrella Graph	Shift=2.440 MD=-1.256	Shift=6.928 MD=6.928	Shift=8.868 MD=8.868	Shift=0.370 MD=-0.034
Winged Hexagon Graph	Shift=4.877 MD=4.107	Shift=2.501 MD=2.497	Shift=1.583 MD=-1.583	Shift=1.290 MD=-0.808
Colliding Cycles Graph	Shift=5.042 MD=3.630	Shift=3.612 MD=3.320	Shift=4.168 MD=0.508	Shift=16.240 MD=14.032
Lemniscate Graph	Shift=3.494 MD=-3.418	Shift=7.703 MD=3.687	Shift=5.995 MD=-3.871	Shift=3.497 MD=-3.417
Cross Box Graph	Shift=0.202 MD=0.170	Shift=0.141 MD=-0.093	Shift=1.532 MD=-0.938	Shift=0.676 MD=0.572

Figure 6.2.4: Starting Vertex 4 Wasserstein Table:

Graph	Shift and Mean Difference (q=0.05)	Shift and Mean Difference (q=0.10)	Shift and Mean Difference (q=0.15)	Shift and Mean Difference (q=0.20)
Complete Graph (K5)	Shift=0.187 MD=-0.111	Shift=0.517 MD=-0.517	Shift=0.297 MD=0.201	Shift=3.021 MD=3.003
Regular Random Graphs (RRGs)	Shift=28.802 MD=28.750	Shift=17.464 MD=14.778	Shift=20.887 MD=18.685	Shift=11.911 MD=2.521
Erdős-Rényi-type Graphs	Shift=2.817 MD=-1.235	Shift=7.546 MD=7.302	Shift=7.637 MD=7.613	Shift=3.118 MD=0.090
Tree Graph	Shift=0.400 MD=-0.364	Shift=0.198 MD=-0.130	Shift=0.268 MD=0.048	Shift=0.290 MD=0.262
Umbrella Graph	Shift=2.646 MD=-1.754	Shift=1.648 MD=-1.448	Shift=1.628 MD=-1.620	Shift=0.606 MD=-0.334
Winged Hexagon Graph	Shift=0.912 MD=-0.892	Shift=1.561 MD=-1.561	Shift=3.755 MD=3.745	Shift=0.422 MD=-0.174
Colliding Cycles Graph	Shift=2.898 MD=2.898	Shift=4.568 MD=4.568	Shift=6.448 MD=1.412	Shift=1.740 MD=1.000
Lemniscate Graph	Shift=0.782 MD=-0.076	Shift=5.664 MD=2.360	Shift=4.124 MD=-3.616	Shift=1.155 MD=-1.029
Cross Box Graph	Shift=0.340 MD=-0.320	Shift=0.159 MD=0.031	Shift=3.823 MD=3.239	Shift=1.373 MD=-1.373

Figure 6.2.5: Starting Vertex 5 Wasserstein Table:

Graph	Shift and Mean Difference (q=0.05)	Shift and Mean Difference (q=0.10)	Shift and Mean Difference (q=0.15)	Shift and Mean Difference (q=0.20)
Complete Graph (K5)	Shift=0.224 MD=-0.174	Shift=0.604 MD=-0.598	Shift=2.988 MD=2.988	Shift=1.000 MD=-1.000
Regular Random Graphs (RRGs)	Shift=7.505 MD=-4.177	Shift=6.722 MD=-3.388	Shift=17.637 MD=-17.637	Shift=30.852 MD=-30.852
Erdős-Rényi-type Graphs	Shift=6.749 MD=6.247	Shift=2.735 MD=0.395	Shift=4.504 MD=3.600	Shift=6.369 MD=5.693
Tree Graph	Shift=0.434 MD=-0.406	Shift=0.674 MD=0.618	Shift=0.848 MD=0.688	Shift=0.480 MD=-0.360
Umbrella Graph	Shift=6.856 MD=6.840	Shift=2.424 MD=-1.112	Shift=1.598 MD=-0.362	Shift=0.822 MD=0.786
Winged Hexagon Graph	Shift=1.410 MD=-1.410	Shift=2.213 MD=2.157	Shift=1.858 MD=-1.858	Shift=1.278 MD=-1.278
Colliding Cycles Graph	Shift=2.472 MD=-0.948	Shift=2.066 MD=-0.978	Shift=3.578 MD=-3.146	Shift=8.508 MD=-2.100
Lemniscate Graph	Shift=1.178 MD=-0.274	Shift=4.442 MD=-2.456	Shift=8.450 MD=7.090	Shift=1.637 MD=-1.635
Cross Box Graph	Shift=0.252 MD=0.228	Shift=0.175 MD=0.147	Shift=4.762 MD=4.430	Shift=2.306 MD=2.282

CONFIDENCE INTERVALS (C=95%)

Figure 6.3.1: Starting Vertex 1 Intervals Table:

Graph	Shift and Mean Difference Interval (q=0.05)	Shift and Mean Difference Interval (q=0.10)	Shift and Mean Difference Interval (q=0.15)	Shift and Mean Difference Interval (q=0.20)
Complete Graph (K5)	Shift: (0.118,0.372) MD: (-0.255,0.317)	Shift: (0.397,0.865) MD: (-0.865,-0.348)	Shift: (0.209,0.842) MD: (0.126,0.842)	Shift: (0.209,0.706) MD: (0.006,0.697)
Regular Random Graphs (RRGs)	Shift: (4.692,12.965) MD: (-10.659,9.306)	Shift: (6.349,18.604) MD: (-18.584,0.148)	Shift: (7.305,17.037) MD: (-14.640,5.230)	Shift: (45.669,69.915) MD: (39.841,66.964)
Erdős-Rényi-type Graphs	Shift: (3.614,10.598) MD: (-7.481,7.505)	Shift: (3.362,12.261) MD: (-4.634,11.917)	Shift: (4.904,17.480) MD: (-0.577,16.566)	Shift: (4.646,13.967) MD: (-10.776,5.375)
Tree Graph	Shift: (0.486,1.607) MD: (-1.459,1.054)	Shift: (0.528,2.224) MD: (-2.215,0.364)	Shift: (0.433,1.499) MD: (-1.278,1.124)	Shift: (0.518,2.137) MD: (-2.091,0.392)
Umbrella Graph	Shift: (0.934,2.102) MD: (-1.985,0.059)	Shift: (1.830,3.112) MD: (-1.729,0.482)	Shift: (4.549,7.249) MD: (4.549,7.249)	Shift: (0.369,1.816) MD: (-1.816,0.018)
Winged Hexagon Graph	Shift: (0.480,1.469) MD: (-0.427,1.317)	Shift: (1.192,2.389) MD: (-2.382,-1.141)	Shift: (5.299,7.718) MD: (5.277,7.718)	Shift: (1.253,2.717) MD: (0.961,2.626)
Colliding Cycles Graph	Shift: (0.688,2.886) MD: (-2.587,0.990)	Shift: (0.831,3.183) MD: (-2.991,0.474)	Shift: (1.504,4.442) MD: (-4.399,-1.103)	Shift: (4.070,6.687) MD: (-6.026,-2.247)

Lemniscate Graph	Shift: (2.062,5.620) MD: (-5.620,-1.713)	Shift: (4.311,7.781) MD: (-7.131,-2.434)	Shift: (4.975,8.191) MD: (-7.285,-2.742)	Shift: (1.932,5.691) MD: (-5.676,-1.532)
Cross Box Graph	Shift: (0.129,0.471) MD: (-0.298,0.457)	Shift: (0.147,0.619) MD: (-0.618,0.058)	Shift: (0.966,1.571) MD: (-1.571,-0.966)	Shift: (0.998,1.638) MD: (-1.638,-0.998)

Figure 6.3.2: Starting Vertex 2 Intervals Table:

Graph	Shift and Mean Difference Interval (q=0.05)	Shift and Mean Difference Interval (q=0.10)	Shift and Mean Difference Interval (q=0.15)	Shift and Mean Difference Interval (q=0.20)
Complete Graph (K5)	Shift: (0.133,0.534) MD: (-0.165,0.508)	Shift: (0.712,1.416) MD: (0.697,1.416)	Shift: (0.717,1.206) MD: (-1.206,-0.715)	Shift: (0.690,1.226) MD: (-1.226,-0.690)
Regular Random Graphs (RRGs)	Shift: (3.946,12.685) MD: (-12.150,5.752)	Shift: (4.539,15.553) MD: (-15.553,4.214)	Shift: (22.573,40.080) MD: (-40.067,-22.488)	Shift: (14.757,32.714) MD: (-32.714,-13.324)
Erdős-Rényi-type Graphs	Shift: (4.040,12.720) MD: (-11.748,5.560)	Shift: (5.830,20.891) MD: (3.446,20.319)	Shift: (9.101,24.246) MD: (-24.246,-8.748)	Shift: (4.124,14.830) MD: (-14.517,2.025)
Tree Graph	Shift: (0.122,0.455) MD: (-0.439,0.250)	Shift: (0.122,0.438) MD: (-0.412,0.307)	Shift: (0.145,0.565) MD: (-0.202,0.513)	Shift: (0.128,0.616) MD: (-0.616,0.117)
Umbrella Graph	Shift: (5.986,9.892) MD: (5.901,9.892)	Shift: (5.263,8.814) MD: (5.186,8.810)	Shift: (2.533,4.360) MD: (-1.110,1.670)	Shift: (0.476,1.973) MD: (-0.630,1.965)
Winged Hexagon Graph	Shift: (0.848,3.063)	Shift: (12.545,16.999)	Shift: (1.027,3.343)	Shift: (0.634,2.015)

	MD: (-3.060,-0.522)	MD: (8.883,14.400)	MD: (-3.283,-0.624)	MD: (-1.823,1.029)
Colliding Cycles Graph	Shift: (1.399,3.266) MD: (-3.266,-1.310)	Shift: (1.333,2.596) MD: (-2.260,-0.068)	Shift: (4.264,5.770) MD: (-4.300,-1.855)	Shift: (1.958,3.450) MD: (-0.800,1.761)
Lemniscate Graph	Shift: (1.637,4.064) MD: (1.484,4.055)	Shift: (7.391,11.102) MD: (7.382,11.102)	Shift: (1.618,3.041) MD: (-1.521,1.193)	Shift: (1.116,3.373) MD: (0.889,3.361)
Cross Box Graph	Shift: (0.105,0.424) MD: (-0.299,0.373)	Shift: (0.150,0.507) MD: (-0.217,0.483)	Shift: (1.533,2.027) MD: (-1.770,-1.033)	Shift: (0.672,1.433) MD: (0.609,1.410)

Figure 6.3.3: Starting Vertex 3 Intervals Table:

Graph	Shift and Mean Difference Interval (q=0.05)	Shift and Mean Difference Interval (q=0.10)	Shift and Mean Difference Interval (q=0.15)	Shift and Mean Difference Interval (q=0.20)
Complete Graph (K5)	Shift: (0.134,0.479) MD: (-0.146,-0.690)	Shift: (0.890,1.600) MD: (0.884,1.600)	Shift: (0.703,1.210) MD: (-1.210,-0.703)	Shift: (0.281,0.825) MD: (0.097,0.807)
Regular Random Graphs (RRGs)	Shift: (4.084,15.431) MD: (-5.240,15.004)	Shift: (20.605,42.815) MD: (19.490,42.703)	Shift: (9.195,28.442) MD: (7.278,27.974)	Shift: (13.300,32.464) MD: (8.221,30.813)
Erdős-Rényi-type Graphs	Shift: (4.545,13.312) MD: (-5.068,12.785)	Shift: (4.735,14.553) MD: (-13.817,2.766)	Shift: (4.505,11.544) MD: (-9.124,7.767)	Shift: (13.791,33.569) MD: (13.324,33.507)
Tree Graph	Shift: (0.285,1.240) MD: (-0.543,1.207)	Shift: (0.257,1.015) MD: (-0.963,0.588)	Shift: (0.233,0.896) MD: (-0.737,0.770)	Shift: (0.280,0.901) MD: (-0.771,0.717)
Umbrella Graph	Shift: (2.035,3.226) MD: (-2.568,-0.022)	Shift: (5.158,8.889) MD: (5.076,8.889)	Shift: (6.829,11.309) MD: (6.516,11.307)	Shift: (0.426,1.451) MD: (-1.350,1.051)
Winged Hexagon Graph	Shift: (3.847,5.993) MD: (2.865,5.363)	Shift: (1.571,3.456) MD: (1.465,3.456)	Shift: (0.821,2.510) MD: (-2.506,-0.706)	Shift: (0.838,1.970) MD: (-1.614,0.127)
Colliding Cycles Graph	Shift: (3.959,6.204) MD:	Shift: (2.698,4.670) MD:	Shift: (3.209,5.160) MD:	Shift: (13.373,19.262) MD:

	(2.084,5.129)	(2.035,4.545)	(-0.860,1.971)	(10.736,17.357)
Lemniscate Graph	Shift: (2.616,5.424) MD: (-5.420,-1.453)	Shift: (5.663,10.144) MD: (0.540,6.869)	Shift: (5.023,7.498) MD: (-6.243,-1.502)	Shift: (2.440,5.342) MD: (-5.318,-1.371)
Cross Box Graph	Shift: (0.161,0.502) MD: (-0.185,0.489)	Shift: (0.123,0.520) MD: (-0.494,0.278)	Shift: (1.326,1.812) MD: (-1.347,-0.522)	Shift: (0.428,0.959) MD: (0.177,0.890)

Figure 6.3.4: Starting Vertex 4 Intervals Table:

Graph	Shift and Mean Difference Interval (q=0.05)	Shift and Mean Difference Interval (q=0.10)	Shift and Mean Difference Interval (q=0.15)	Shift and Mean Difference Interval (q=0.20)
Complete Graph (K5)	Shift: (0.140,0.442) MD: (-0.430,0.180)	Shift: (0.261,0.793) MD: (-0.791,-0.247)	Shift: (0.177,0.609) MD: (-0.140,0.542)	Shift: (2.599,3.544) MD: (2.550,3.526)
Regular Random Graphs (RRGs)	Shift: (19.392,40.144) MD: (18.169,40.067)	Shift: (10.127,27.221) MD: (3.901,25.448)	Shift: (12.952,30.043) MD: (7.620,28.953)	Shift: (7.988,19.951) MD: (-6.949,14.497)
Erdős-Rényi-type Graphs	Shift: (3.283,9.353) MD: (-8.008,6.312)	Shift: (4.022,16.306) MD: (-0.644,15.986)	Shift: (3.867,15.288) MD: (-0.657,15.206)	Shift: (3.340,9.051) MD: (-7.078,7.750)
Tree Graph	Shift: (0.219,0.917) MD: (-0.880,0.098)	Shift: (0.209,0.697) MD: (-0.661,0.396)	Shift: (0.251,0.676) MD: (-0.481,0.588)	Shift: (0.202,0.839) MD: (-0.281,0.821)
Umbrella Graph	Shift: (2.148,3.381) MD: (-2.731,-0.710)	Shift: (0.990,2.421) MD: (-2.292,-0.498)	Shift: (0.974,2.485) MD: (-2.451,-0.890)	Shift: (0.412,1.393) MD: (-1.218,0.678)
Winged Hexagon Graph	Shift: (0.426,1.742) MD: (-1.715,-0.034)	Shift: (0.869,2.352) MD: (-2.352,-0.815)	Shift: (2.771,4.820) MD: (2.714,4.815)	Shift: (0.366,1.138) MD: (-1.069,0.612)

Colliding Cycles Graph	Shift: (1.680,3.945) MD: (1.660,3.939)	Shift: (3.330,5.795) MD: (3.288,5.792)	Shift: (5.223,7.844) MD: (-0.165,3.292)	Shift: (1.195,2.525) MD: (0.043,2.017)
Lemniscate Graph	Shift: (0.785,2.261) MD: (-1.957,1.456)	Shift: (4.295,7.474) MD: (-0.129,4.944)	Shift: (3.049,5.447) MD: (-5.195,-1.699)	Shift: (0.762,2.818) MD: (-2.676,0.815)
Cross Box Graph	Shift: (0.179,0.659) MD: (-0.625,0.030)	Shift: (0.133,0.444) MD: (-0.304,0.391)	Shift: (3.227,4.523) MD: (2.532,4.089)	Shift: (1.071,1.688) MD: (-1.686,-1.071)

Figure 6.3.5: Starting Vertex 5 Intervals Table:

Graph	Shift and Mean Difference Interval (q=0.05)	Shift and Mean Difference Interval (q=0.10)	Shift and Mean Difference Interval (q=0.15)	Shift and Mean Difference Interval (q=0.20)
Complete Graph (K5)	Shift: (0.132,0.461) MD: (-0.420,0.120)	Shift: (0.396,0.878) MD: (-0.866,-0.288)	Shift: (2.564,3.456) MD: (2.558,3.456)	Shift: (0.757,1.250) MD: (-1.250,-0.757)
Regular Random Graphs (RRGs)	Shift: (5.727,16.431) MD: (-15.163,5.762)	Shift: (4.929,15.240) MD: (-13.751,7.362)	Shift: (9.108,27.887) MD: (-27.887,-7.883)	Shift: (21.135,40.377) MD: (-40.377,-21.135)
Erdős-Rényi-type Graphs	Shift: (4.207,16.130) MD: (-3.492,15.013)	Shift: (3.654,12.054) MD: (-9.790,9.844)	Shift: (3.928,13.942) MD: (-6.470,13.246)	Shift: (4.216,15.060) MD: (-14.688,3.413)
Tree Graph	Shift: (0.332,1.203) MD: (-1.203,0.413)	Shift: (0.330,1.564) MD: (-0.334,1.547)	Shift: (0.532,1.742) MD: (-0.313,1.665)	Shift: (0.297,1.242) MD: (-1.179,0.474)
Umbrella Graph	Shift: (5.168,8.694) MD: (4.997,8.665)	Shift: (1.912,3.201) MD: (-2.276,0.167)	Shift: (1.076,2.388) MD: (-1.499,0.765)	Shift: (0.488,1.957) MD: (-0.340,1.929)
Winged Hexagon Graph	Shift: (0.728,2.277)	Shift: (1.346,3.326)	Shift: (1.047,2.702)	Shift: (0.549,2.124)

	MD: (-2.277,-0.619)	MD: (1.192,3.326)	MD: (-2.702,-0.922)	MD: (-2.124,-0.356)
Colliding Cycles Graph	Shift: (1.846,3.688) MD: (-2.797,0.893)	Shift: (1.506,3.481) MD: (-2.865,0.857)	Shift: (2.603,5.240) MD: (-4.900,-1.384)	Shift: (7.080,10.297) MD: (-4.297,0.519)
Lemniscate Graph	Shift: (0.873,2.430) MD: (-2.125,1.631)	Shift: (3.333,6.091) MD: (-4.702,-0.238)	Shift: (6.291,11.244) MD: (4.061,10.485)	Shift: (0.993,3.553) MD: (-3.400,0.037)
Cross Box Graph	Shift: (0.134,0.470) MD: (-0.031,0.457)	Shift: (0.095,0.388) MD: (-0.065,0.375)	Shift: (4.127,5.397) MD: (3.768,5.106)	Shift: (2.012,2.651) MD: (1.970,2.641)

DISCUSSION

Our p-values for the Anderson-Darling Tests showed varied results for significance, however, evident patterns emerge. For instance, increasing the noise probability across the graphs generally increases the frequency of statistically significant differences, though it should be noted this is not strictly monotonic behavior. K5 showed significant differences at lower levels and more consistently than the large Erdős–Rényi graph, while the RRGs showed increasingly consistent significance at higher noise levels. It is also important to note that significance varied across the starting vertices within the same graph types and noise levels, which suggests that return-time distribution sensitivity to noise-controlled perturbations is influenced by the starting vertex. Notably, some graph families become less significant at the highest noise level.

In addition, our results for the Wasserstein distance and mean difference yielded a variety of patterns as well. It is important to note that while the pattern that the magnitude of shifts increases with graph size, it is not an indicator of significant results, and cannot be used to isolate individual effects of perturbations on the return time distributions. Shifts in magnitudes however, did not consistently increase with noise probability, implying a nonlinear, non-monotonic relationship between the influence of random noise operators on graphs and return times. Mean differences were both positive and negative across graph families, starting vertices, and noise level, which gives more evidence for the non-monotonic behavior described. Finally, shift magnitudes varied considerably depending on starting vertex regardless of graph family and noise level, which adds evidence that the starting position of random walks may influence return-time distribution sensitivity to noise-controlled perturbations.

Our confidence intervals provide further patterns and supporting evidence. For instance, confidence intervals for shift were consistently above zero across the graph families, which suggests positive

distributional distances across every reported condition. Confidence intervals for mean difference vary considerably across different starting vertices, which further implies that the direction of the mean is influenced by the starting vertex of the simple random walk. It is also important to note that these intervals do not change monotonically as noise increases, further suggesting that the effects of noise-controlled perturbations on return time distributions is non-monotonic.

It is noteworthy that the Anderson-Darling Test results can also be verified by the confidence intervals, since a result with changing signs on the bounds of intervals provides evidence for non-statistically significant results, and our confidence level of C=95% matches with our significance level of α=0.05. The AD tests match with these intervals 79.3% of the time, though this descriptive statistic should not be taken as proof of validation. Note that this number may be affected by unusual data produced from the issues cited previously with higher noise levels on the smaller graphs.

LIMITATIONS

While the results of this experiment are helpful in our understanding of noise operators and their effects on return time distributions, this study does have significant limitations.

The major limitation of this study is the graph types we used to run tests on. Because these graphs are relatively small, it is likely that perturbations to these graphs may cause disconnections to the target node, which results in the return time approaching infinity, as the walker would never get back to the disconnected return node. Because of this, statistical results at higher noise levels for these graphs become more unstable and unreliable, and outlier results for noise levels q=0.2 and q=0.25 should be taken with a grain of salt. Furthermore, many of our observations come from general patterns and observations, and outliers within the data will need to be investigated further in future studies. Finally, certain observations such as non-monotonicity are untested, and our results will produce both conclusions supported by the results, and candidate findings with great supporting evidence derived from patterns within the results.

A secondary limitation that arises from this study is that it is purely statistical. Statistical results allow for trends in data to be revealed and analyzed, which could lead mathematicians toward a direction of interest when researching random walks. However, because our data is sourced from a computer simulation, it is not purely analytical, and trends discussed in the Analysis section are purely suggestive of the impact of noise operators on the return time distributions of small graphs, and are by no means mathematically proven to be true. Until such results are absolutely proven to be true, any trend that results from statistical analysis of computer simulations of this scenario is merely a trend and not foundational theory**.**

CONCLUSION

Ultimately, we conclude that random edge noise on graphs significantly shifts their return-time distribution, but this shift is not determined by noise level alone. While the first stream of literature established that graph structure alone shapes return-time behavior on static, unperturbed graphs, our

findings extend this by showing that the stability of that behavior under perturbation is itself structure-dependent, rather than governed by noise level alone. Furthermore, our studies extended the work done within the second stream of literature on randomly changed graph structures to full return time distributions, contributing to the gap left by previous research as a result of focus on aggregate data such as mixing times on randomly perturbed graphs rather than return time distributions. More generally, our results support the following conclusions:

1) Noise perturbations influence return time distributions, though this significance is not universal and may be influenced by external factors like the starting vertex of a simple random walk.
2) Distributional shifts do not always correspond to significant changes in mean return time. The Wasserstein distances were always positive, indicating a shift in the distribution, but our mean difference intervals occasionally cross zero, showing a shift in the distribution but a non-significant change in mean return time.

Our results also support the following plausible candidate findings:

1) The influence of random noise operators on return time distributions is non-monotonic, i.e., sometimes the return time is increased, and sometimes it is decreased
2) Not all vertices of a graph are affected in the same way. For instance, we know the return time of a vertex can be calculated using Kac's formula, but because of the random perturbations, random vertices may lose a random number of edges and decrease in degree, changing the Kac's formula calculation and yielding varying impact for different vertices.
3) Return time distributions for smaller graphs are more influenced by random perturbations than larger graphs. This is supported by the pattern of significant results, which shows significant results present on smaller graphs like K5 earlier and more consistently than our larger Erdős–Rényi graphs.
4) Structural differences and the initial vertex influence return time distributions. This is observed through the variation in results, in which identical graph structures showed varying results across different starting positions, and different graph structures showed varying results amongst each other.

Future work should test whether this interaction holds on local graph structures, where the impact of the independent Bernouli deletion of every edge may differ significantly from the graphs examined. Alternatively, future work could investigate the addition of random edge noise rather than the deletion of edges in graphs.

Just as the aliens' best estimate of their return time hinges on knowing the road network's connectivity, our study shows that a single quick fact, degree relative to total edges, can meaningfully predict return time even before any walk begins. But it also shows that such an estimate is only as reliable as the road network itself remains. The moment a road is removed, as our study demonstrates through structural perturbation, the aliens' careful calculation may no longer hold.

All funding towards this paper came from personal funds and was managed by the Lumiere Education Research Program. There were no conflicts of interests during the writing of this paper, this paper was

initially written for Lumiere Education as a part of their Premium Research and Publication Program, and my mentor Gaia Carenini helped with debugging, revisions, and left feedback on drafts.

REFERENCES

Bala Priya, C. (2025). *Analyzing random walk resilience to targeted graph attacks through mixing time and coverage measures*. In *2025 5th Asian Conference on Innovation in Technology (ASIANCON)*. IEEE. https://doi.org/10.1109/ASIANCON66527.2025.11281280

Chung, F., & Robertson, S. J. (2025). Stochastically evolving graphs via edit semigroups. *Proceedings of the National Academy of Sciences, 122*(49), Article e2526595122. https://doi.org/10.1073/pnas.2526595122

Eriksson, K. (2023). *Perron–Frobenius' theory and applications* [Bachelor's thesis, Linköping University]. https://www.diva-portal.org/smash/get/diva2:1768625/FULLTEXT01.pdf

Evnin, O., & Horinouchi, W. (2025). First return times on sparse random graphs. *Journal of Physics A: Mathematical and Theoretical, 58*(7), Article 075002. https://doi.org/10.1088/1751-8121/adb084

González, F. H., Riascos, A. P., & Boyer, D. (2021). Diffusive transport on networks with stochastic resetting to multiple nodes. *Physical Review E, 103*(6), Article 062126. https://doi.org/10.1103/PhysRevE.103.062126

Grinstead, C. M., & Snell, J. L. (1997). *Introduction to probability* (2nd revised ed.). American Mathematical Society. https://math.dartmouth.edu/~prob/prob/prob.pdf

Hörmann, E., Lambiotte, R., & Cantwell, G. T. (2025). Approximation for return time distributions of random walks on sparse networks. *Physical Review E, 111*(6), Article 064306. https://doi.org/10.1103/PhysRevE.111.064306

Kawamura, H., Shiina, S., Aung, H. N., & Ohsaki, H. (2024). Robustness of random walk on a graph against adversary attacks. In *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE. https://doi.org/10.1109/COMPSAC61105.2024.00146

Lev-Ari, D., Tishby, I., Biham, O., Katzav, E., & Krapf, D. (2025). Analytical results for the distribution of first return times of non-backtracking random walks on configuration model networks. *Journal of Physics A: Mathematical and Theoretical, 58*, Article 505002. https://doi.org/10.1088/1751-8121/ae257d

Lovász, L. (1993). Random walks on graphs: A survey. In *Combinatorics, Paul Erdős is Eighty* (Vol. 2, pp. 1–46). Bolyai Society Mathematical Studies. https://www.cs.cmu.edu/~15859n/RelatedWork/random-walks-on-graphs.pdf

Masuda, N., & Konno, N. (2004). Return times of random walk on generalized random graphs. *Physical Review E, 69*(6), Article 066113. https://doi.org/10.1103/PhysRevE.69.066113

Michelitsch, T. M., D'Onofrio, G., Polito, F., & Riascos, A. P. (2025). Random walks with stochastic resetting in complex networks: A discrete-time approach. *Chaos, 35*(1), Article 013119. https://doi.org/10.1063/5.0238517

Sauerwald, T., & Zanetti, L. (2019). Random walks on dynamic graphs: Mixing times, hitting times, and return probabilities. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)* (LIPIcs, Vol. 132, Article 93). Schloss Dagstuhl – Leibniz-Zentrum für Informatik. https://doi.org/10.4230/LIPIcs.ICALP.2019.93

Stephan, L., & Massoulié, L. (2022). Non-backtracking spectra of weighted inhomogeneous random graphs. *Mathematical Statistics and Learning, 5*(3–4), 201–271. https://doi.org/10.4171/MSL/34

Thapar, V., Lambiotte, R., & Cantwell, G. T. (2025). *Embedding networks with the random walk first return time distribution* (arXiv:2512.02694). arXiv. https://arxiv.org/abs/2512.02694

Tishby, I., Biham, O., & Katzav, E. (2021). Analytical results for the distribution of first return times of random walks on random regular graphs. *Journal of Physics A: Mathematical and Theoretical, 54*(32), Article 325001. https://doi.org/10.1088/1751-8121/ac0d34

Ye, Y., & Chen, H. (2022). Random walks on complex networks under node-dependent stochastic resetting. *Journal of Statistical Mechanics: Theory and Experiment, 2022*(5), Article 053201. https://doi.org/10.1088/1742-5468/ac625b