Optimization of Transcription Factor Search Times: A Comparative Analysis of Classical and Facilitated Diffusion Models

Hasif Ahmed[1]

[1]*Physics Department, Lawrence University, 711 E Boldt Way, Appleton, WI 54911*

22nd November, 2023

Transcription factors (TFs) are pivotal in gene regulation, necessitating efficient search strategies to locate specific DNA sequences. This thesis investigates the dynamics of TF search times by contrasting classical 1D and 3D diffusion models with facilitated diffusion mechanisms. Classical models predict impractically long search times, with 1D diffusion scaling quadratically with DNA length and 3D diffusion providing optimistic estimates when not accounting for nonspecific binding. Our facilitated diffusion model introduces a synergy between 1D sliding along the DNA and 3D excursions, accelerating the search process significantly. We demonstrate that this model reduces the target attachment time to the order of tens of seconds, aligning with empirical data. Additionally, we extend our analysis to 2D diffusion on a square lattice, revealing a linear scaling of mean first passage time (MFPT) with lattice size under increased simulation numbers, suggesting efficient search kinetics applicable to various biological systems. These findings offer critical insights into the molecular mechanisms underlying cellular function and have the potential to inform the development of artificial biosystems and targeted therapies. The numerical simulations underpinning our theoretical framework are substantiated by a robust computational approach, elucidated within the text.

Introduction

Transcription factors (TFs) are essential in gene regulation, necessitating rapid location of their target DNA segments within cells. The search time for TFs is estimated using 1D and 3D diffusion models.

For 1D classical diffusion, the search time t_{search} is given by:

$$t_{\text{search}} = \frac{L^2}{D_{\text{1D}}}, \tag{1}$$

where L is the total length of the DNA and D_{1D} is the 1D diffusion coefficient. In bacteria, with $L = 10^6$ nm and $D_{\text{1D}} = 10^4 - 10^5$ nm^2/s, this results in a search time of thousands of hours. These values are taken from [1] and [2].

In contrast, the 3D diffusion model estimates the search time as:

$$t_{\text{search}} = \frac{V}{D_{\text{3D}} r}, \tag{2}$$

where V is the volume restricting the TF and its target, D_{3D} is the 3D diffusion coefficient, and r is the typical spatial size of the target- also known as 1 base-pair length. For typical bacterial values ($V = 10^9$ nm^3, $D_{\text{3D}} = 10^7$ nm^2/s, $r = 0.34$ nm), the search time is in the order of hundreds of seconds. However, this estimate is likely optimistic as the protein may spend time bound to non-target sites. [3]

In the model of facilitated diffusion- we show that there exists a coupling between 1D random walk and 3D excursions that results in a speed-up for the search process of the DNA that brings the target attachment time to the order of tens of seconds. We demonstrate that the 3D excursion, in fact, gets rid off 1D random walks that can lead to very long, if not infinite search times. For our purpose, we assume that the target sites are perfectly absorbant of the TF proteins.

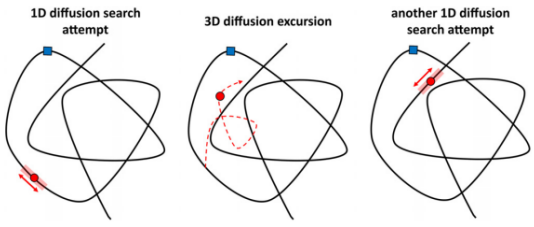

Figure 1: Cartoon illustrating the facilitated diffusion search mechanism. The TF (circle) performs 1D diffusion along the DNA (solid line). It then falls off, performs 3D diffusion, and lands somewhere else along the DNA (3D diffusion excursion)—which could get it closer to its target (square) or further away from it. This process repeats itself until the TF finds the target. [3].

Facilitated Diffusion in 1D

We model the DNA strand as a one-dimensional line segment $AB \in \mathcal{R}$, reflecting the simplest spatial structure for our simulation. At each timestep, the TF protein may move

either right or left along this segment or undergo a 3D excursion, with the latter represented by a fall-off probability γ.

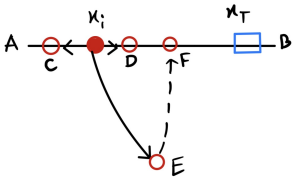

Figure 2: At each point in our simulation, the TF particle at x_i can either go right (D), left (C) or fall off for a 3D excursion (E) till it arrives back at a random position in the strand (F)

Based on our schematic model, our numerical simulation picks up a random position of our DNA (AB) where the length of the DNA strand is 5×10^5 nm, the timestep $\delta t = 10^{-5}$ seconds and the 3D excursion duration we take to be $t_{3D} = 10^{-4}$ seconds following physical measures as mentioned in [3]. We take $\gamma = 0.08$, which is a reasonable fall off probability for our simulation scale. We have a target position x_T, which our TF particle starting from a random initial position x_i wants to reach. At every step, our particle has $\frac{1-\gamma}{2}$ probability of going either right or left by counstruction of the model. Another simplification we made to our model is : when our TF particle reaches L- the right most corner of the strand, it moves left from that position. Similarly, when our particle is at the left most position A(0) in our strand, it moves to the right.
Our simulation stops when the target attachment occurs and throughout the process till attachment, we have a timestep counter. By the end of a simulation, our timestep counter gives us the total time it took for our TF to reach the target we assigned. To understand the simulation with more depth, please refer to the codes we provided in the Appendix.

1 Mathematical Descriptions for 1D Facilitated Diffusion

Here, we derive the mathematical prediction for our bacteria's transcription search completion.

For the 1D facilitated diffusion model, [3] proposes that the probability of hitting a target from target x is described by:

$$p(x) = \exp\left(-\sqrt{\frac{\Gamma}{D}}|x|\right) \tag{3}$$

where $\Gamma = \gamma/\delta t$ and D is the diffusion coefficient.

This result is crucial to derive the theoretical mean first passage time (MFPT) prediction which we will call $T(x)$. The Mean First Passage Time (MFPT) is a critical metric in our analysis, representing the expected time duration for a transcription factor (TF) protein to first reach a specific target site on the DNA strand, under the governed dynamics of diffusion and binding interactions, computed over numerous search attempts.

For a target at distance x, we want to compute $\tilde{T}(x)$ where it follows the following recursion relation:

$$\tilde{T}(x) = \delta t p(x) + (1-\gamma)\frac{\tilde{T}(x+\delta x) + \tilde{T}(x-\delta x)}{2} \tag{4}$$

Here, $\tilde{T}(x)$ includes average time of hitting the target over all trajectories that eventually get to target. So here, when advancing a time step δt, the time contributed to $T(x)$ is the product of δt and $p(x)$. We want to derive the continuum limit of Eq. 4.

Subtracting $T(x)$ from both sides of Eq. 4 and multiplying both sides by $\frac{2}{(\delta x)^2}$ we get :

$$\frac{2\delta t p(x)}{(\delta x)^2} + \frac{\tilde{T}(x+\delta x) + \tilde{T}(x-\delta x) - 2T(x)}{\delta x)^2} - \frac{2\gamma t}{2\gamma t}\frac{\gamma(\tilde{T}(x+\delta x) + \tilde{T}(x-\delta x))}{(\delta x)^2} = 0 \tag{5}$$

Taking the continuum limit $\delta x \to 0, \delta t \to 0$ and $\gamma \to 0$ and using the definition of D and Γ, we get a linear differential equation:

$$D\frac{d^2T(x)}{dx^2} - \Gamma T(x) + p(x) = 0 \tag{6}$$

The boundary conditions applied are $T(0) = T(x \to \infty) = 0$. The initial condition stems from the characterization of $T(x)$ in the context of an average first passage time problem. Moreover, $T(x)$ is exclusively derived from trajectories that successfully reach the target. As x tends towards infinity, the occurrence of such successful trajectories diminishes significantly. This reduction is attributed to the fact that, as illustrated in Eq. 3, the probability of a successful trajectory initiating from a position x reduces exponentially with $|x|$. [3]

Eq. 6 is solvable using variation of parameters as recommended in [3]. We were also able to arrive at the same result by guessing and checking certain solutions. The solution to Eq. 6 is:

$$T(x) = \frac{|x|}{2\sqrt{\Gamma D}}\exp\left(-\sqrt{\frac{\Gamma}{D}}|x|\right) \tag{7}$$

We here use the concept of conditional probability to account for the fact that we want to consider successful trajectories. So we divide Eq. 7 by the probability of success defined by Eq. 3 and we arrive at:

$$\hat{T}(x) = \frac{|x|}{2\sqrt{\Gamma D}} \tag{8}$$

This result tells us that for successful trajectories, the time to reach a target at distance $|x|$ grows linearly. To arrive at the result for MFPT, we need to compute mean of all the binding positions on our strand. This procedure is explained well in [3]. We use their results of mean first passage time:

$$< T >= \frac{L}{\sqrt{D\Gamma}} + L\sqrt{\frac{\Gamma}{D}} < t_{3D} > \tag{9}$$

In the next section, In the next section we verify the relations Eq. 8 and 9 using our numerical simulation scheme.

2 Comparison of our Simulation with Theory

We want to verify the theoretical predictions of Eq. 8 and 9 using our numerical scheme. To verify the predictions of Eq. 8, we first set a target position at $x_T = L$ for our particle which has initial position $x_i = 0$ in our simulation. We run our simulation for various lengths (L) to find how it affects the time it takes to get at target.

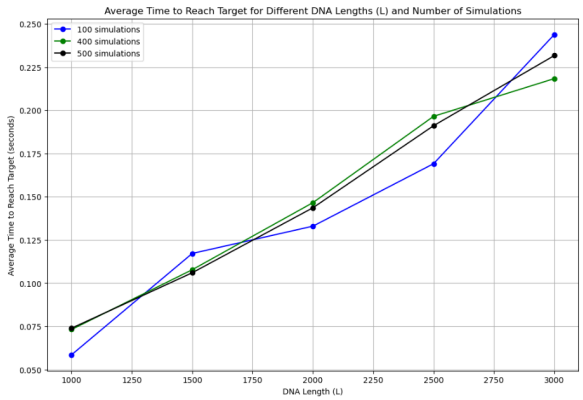

Figure 3: Monte Carlo Simulations for simulated $\hat{T}$

The simulation algorithm extends beyond a singular execution, encompassing a multitude of simulations to achieve a statistically significant sample. This aggregation of multiple runs, each yielding the time taken to reach a pre-defined target position, facilitates the computation of an average metric. Such a methodology is quintessential to Monte Carlo simulations, wherein the convergence of outcomes across numerous iterations provides an empirical yet robust approximation of system behavior.

Fig. 3 shows how for larger number of simulations, $\hat{T}$ plot converges to demonstrating a linear trend with respect to different DNA lengths. This was predicted by Eq. 8.

To derive the mean first passage time, we plot the mean data of multiple initial positions which are randomized in our simulation. Then we employ similar Monte-Carlo simulation for plotting the MFPT of the process.

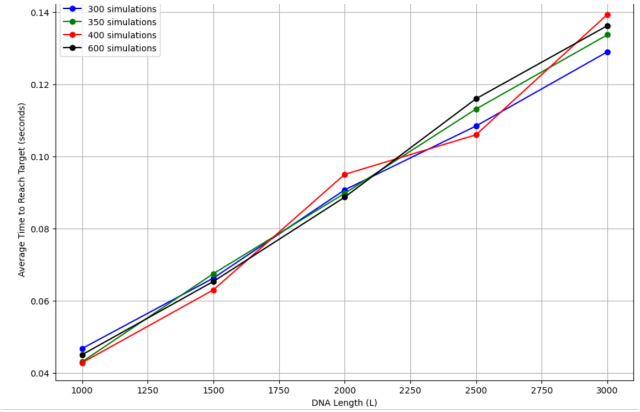

Figure 4: Monte Carlo Simulations for simulated $< T >$

It is imperative to acknowledge that the simulations were conducted over DNA segments ranging from 1000 to 3000 nm in length. While it would be ideal to replicate these simulations for DNA strands of actual biological lengths, which are typically on the order of 5×10^5 nm, such extensive computations were not feasible within the constraints of this project. Nonetheless, the observed linear trends in the simulated data provide insightful approximations regarding the dynamics of the facilitated diffusion process, offering a scalable representation applicable to DNA strands of physiological dimensions.

For the actual DNA strand measures, [3] claim that by taking $\Gamma = \frac{1}{<t_{3D}>}$ in Eq. 9 and then the TF spends half of its time in 1D and half in 3D with the overall search time in the order of tens of seconds. For the actual length of DNA- 5×10^5 nm, We compute the MFPT to be 14.70 seconds for 500 simulations– which is comparable to [3]'s theoretical prediction.

Facilitated Diffusion in 2D

3 Model Construction and Theory Development

Extending our model for 1-Dimensional strands, we want to explore how facilitated diffusion works in 2D. Now our particle can move in 2 dimensions or decide to detach from the surface for 3D excursion till it reaches the target placed on the surface.

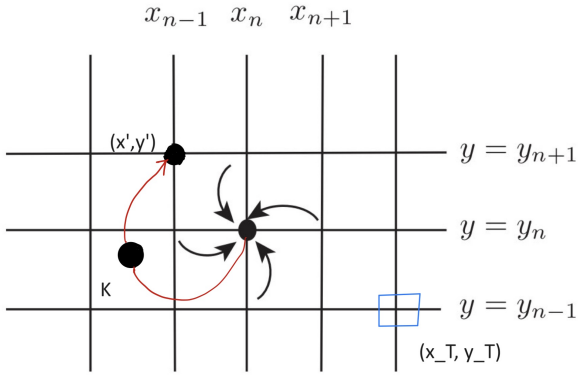

Figure 5: At each point in our simulation, the TF particle at x_i can either go right, left, up or down or fall off for a 3D excursion (K) till it arrives back at a random position in the surface (x', y').

For a particle in this surface, assume the falling off probability is γ as our previous model. Then at each step, the particle has 4 movement directions on the surface- north, south, east and west- parametrized by (x_n, y_n) and the option to fall off with γ. The probabilities should add up to 1. So the recursion relation to determine the probability of an arbitrary particle reaching its target at (x_T, y_T) can be defined by:

$$\tilde{p}(x,y) = (1-\gamma)\left[\frac{\tilde{p}(x,y+\delta y)+\tilde{p}(x,y-\delta y)}{4} + \frac{\tilde{p}(x+\delta x,y)+\tilde{p}(x-\delta x,y)}{4}\right] \quad (10)$$

Subtracting $\tilde{p}(x)$ from both sides of Eq. 10 and multiplying by $\frac{4\delta t}{(\delta x)^2(\delta y)^2}$ we get:

$$\begin{aligned}&\frac{\tilde{p}(x,y+\delta y)+\tilde{p}(x,y-\delta y)-2\tilde{p}(x,y)}{(\delta x)^2(\delta y)^2}\\&+\frac{\tilde{p}(x+\delta x,y)+\tilde{p}(x-\delta x,y)-2\tilde{p}(x,y)}{(\delta x)^2(\delta y)^2}\\&-\frac{\Gamma}{\delta t}\frac{4\delta t}{(\delta x)^2(\delta y)^2}\tilde{p}(x,y)=0\end{aligned}\tag{11}$$

To derive the continuum limit of Eq. 10, we assume $\delta x=\delta y\to 0, \delta t\to 0$ and $\gamma\to 0$ and using the definition of $D=\frac{4\delta t}{(\delta x)^2}$ and $\Gamma=\gamma/\delta t$, we get the differential equation:

$$\nabla^2 p(x,y)=\frac{\Gamma}{D}p(x,y)\tag{12}$$

The operator ∇^2, known as the Laplacian, represents the sum of second-order partial derivatives with respect to the spatial coordinates (x,y). Drawing inspiration from the methodology outlined in [3], we extend our investigation to an infinite domain, characterized by $x,y\in(-\infty,\infty)$, and postulate the target location at the origin $(0,0)$. An iterative approach of guessing and checking leads us to the following expression:

$$p(x,y)=\frac{1}{2}\exp\left(-\sqrt{\frac{\Gamma}{D}}(|x|+|y|)\right)\tag{13}$$

Upon scrutinizing this solution, however, we discern a fundamental inconsistency with physical principles: the derived probability distribution exhibits spherical asymmetry within our two-dimensional context. This implies that two particles, equidistant from the target at $(0,0)$, would nonetheless possess disparate probabilities of reaching it, contingent upon their respective x and y coordinates. Given that the notion of probability should remain invariant under coordinate transformation, we must reject this solution as a viable representation of our system. Hence, we can't pursue the approach to compute the analog of MFPT as outlined in [3].

In our quest to approach this problem from an alternative perspective, we adopt the framework proposed in [4]. This entails resolving an equation of the structure:

$$\frac{dp}{dt}=D\nabla^2 p-\Gamma p,\tag{14}$$

where p is a function of the spatial-temporal coordinates (X,t), with X representing a two-dimensional vector (x,y). The initial condition is established as $p(X,0)=\delta(X-X_0)$. We proceed by integrating $p(X,t)$ over time to derive an analytical form for $p(x,y)$. Then we derive the equation for $T(x,y)$ in analogy to the 1D case. We then condition $T(x,y)$ by our probability of success in order to theoretically predict the MFPT. This analysis, still under development, is intended for inclusion in our forthcoming publication.

4 Numerical Simulation for 2D

In our computational model described in section **3**, we construct a two-dimensional square lattice of size $A=L^2$, where a particle commences its random walk from a randomly assigned starting position. The particle's movement is governed by a probabilistic selection of directions – left, right, up, down, or a 'fall' event. The latter represents a reset to another random location, with a time penalty t_{3d} added to the cumulative time count. For each square lattice with length L, we perform multiple simulations, each running until the particle reaches the predetermined target position and then compute the mean over multiple target positions. The average time to reach the target across these simulations is computed and plotted against the lattice length.

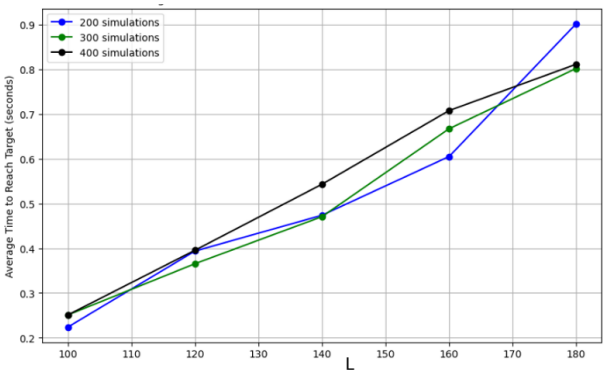

Figure 6: MFPT across various Lattice Sizes for 2D facilitated diffusion

From Fig. 6, it's observed that as the number of simulations increases, the relationship between MFPT and lattice length L and hence lattice size $\sqrt{A}$ starts to show a linear trend rather than a quadratic one. This linear relationship is indicative of an efficient search process, which is less time-consuming and energy-intensive compared to a quadratic relationship where the time would increase at a much faster rate as L increases.The linear trend suggests that for biological systems operating in 2 dimensions, processes such as facilitated diffusion, where the movement of molecules across a membrane is assisted by proteins, can be highly effective.

Discussion

This thesis delves into the intricate mechanisms by which transcription factors (TFs) navigate to their target sites on DNA, juxtaposing classical diffusion models with the more nuanced facilitated diffusion model. Classical diffusion models, while providing foundational insights, fall short in accounting for the rapid search times observed *in vivo*. [2] The quadratic dependency of search time on DNA length posited by 1D models, and the overly optimistic estimates of the 3D models, propelled us to consider facilitated diffusion as a more representative model for TF search dynamics. A crucial aspect of our analysis is the Mean First Passage Time

(MFPT) calculation. Our simulations, adapting the facilitated diffusion framework, yield an MFPT of approximately 14.70 seconds for DNA lengths characteristic of bacterial cells. This finding is consistent with the theoretical predictions in [3]. Such a convergence of simulation and theory underscores the robustness of our model and provides a compelling argument for the efficiency of facilitated diffusion as a search strategy.

The essence of our work lies in the coupling between 1D and 3D diffusion—a nuanced dance between sliding along the DNA and intermittent dissociations leading to 3D excursions. This interplay significantly accelerates the search process, bringing theoretical predictions into harmony with observed biological timescales. The implications of this discovery are profound, offering a mechanistic understanding that transcends the simplistic view of random walks in isolation. It is the orchestration of these movements that encapsulates the efficiency of biological systems, and our simulations have unveiled this orchestration with clarity and precision.

The extension of our model to a 2D lattice further enhances our understanding of TF dynamics in more complex environments. The linear relationship between MFPT and lattice size observed in our 2D simulations suggests an efficient search process, reinforcing the idea that facilitated diffusion remains effective across various spatial configurations. This insight is particularly pertinent in light of the discussions in [5], which delve into the nuances of mean first passage times in different dimensional spaces- particularly surface mediated diffusion in spherical domains.
While our results are promising and align with empirical observations, they also open the door to several avenues for further exploration. The approach to theoretically predict the mean first passage time for 2D facilitated diffusion needs to be further investigated and will be reflected in our updated publication.

The simplified assumptions, such as the instantaneous reattachment of TFs following 3D excursions, might not entirely encapsulate the stochastic nature of molecular interactions in real cellular environments. Furthermore, in the process of TFs attaching with target site, TFs must recognize target sites with 0.34 nm precision, as displacement by 1 base-pair in DNA sequence results in a different site. The sequence-specific energy leads to a rugged energy landscape for protein translocation along DNA. This rugged trajectory influenced by DNA sequencing is further explained in [1]. These environments are characterized by physical barriers and a heterogeneity that could significantly influence the search times. Future models could aim to integrate these complexities, thus providing a more granular understanding of the TF search process.

Additionally, our findings have broader implications beyond the scope of molecular biology. The principles of facilitated diffusion, illustrated through our computational models, could inform the design of efficient search algorithms in synthetic biological systems and nanotechnology.

In summary, this thesis explores the dynamics of transcription factor search behavior through computational simulations based on probabilistic models. While these models provide insights into the mechanisms of facilitated diffusion, the applicability of these findings to real-world biological systems remains to be validated through empirical studies. Our work contributes to the theoretical understanding of molecular search behaviors and underscores the need for further research to integrate these models with experimental data. This study encourages a cautious optimism about the potential for these findings to inform future scientific and technological advancements.

Acknowledgements

I would like to thank Joseph N. Gregg, Kimberly Dickson and Andrew J. Sage- professors from Lawrence University, for their insightful thoughts. I would like to acknowledge GitHub Copilot- a programming AI assistant, for its support with debugging errors in our simulation scheme. I would also like to acknowledge Seeley G. Mudd Library, Lawrence University- for their help with textbook loaning services that were instrumental to the discoveries of our research. Lastly, I would like to acknowledge Douglas S. Martin, Associate Professor in Physics at Lawrence University, for his excellent advisory and constant support throughout the 10 weeks of research that led to this thesis. He is aware of the struggles and barriers faced in my pursuit of thesis completion and I am thankful for every bit of his support.

Appendix

Appendix A: 1D Simulation Code

The following Python code was followed for the 1D simulations presented in this thesis with subtle variations in parameters for different cases:

```
import numpy as np
import matplotlib.pyplot as plt

import numpy as np
import matplotlib.pyplot as plt

# Parameters for the simulations
gamma = 0.08
t_3d = 10**(-4) - 10**(-5)  # Time to return after falling off. This was computed such that total 3d excursion duration is 10**(-4) seconds
target_position = 990  # The target position
L_values = np.arange(1000, 3001, 500)  # DNA lengths from 1000 to 3000 with increments of 500
num_simulations_values = [300, 350, 400, 600]  # Different numbers of simulations to try

# Function to simulate trajectory
def simulate_trajectory(L, gamma, t_3d, target_position):
    position = np.random.randint(0, high=L+1)  # Start at a random position including L
```

```
    time_count = 0  # Initialize time count

    # Simulate until the target is reached
    while position != target_position:
        # Check for border conditions
        if position == 0:
            move = 'right'
        elif position == L:
            move = 'left'
        else:
            move = np.random.choice(['left', 'right', 'fall'], p=[(1-gamma)/2, (1-gamma)/2, gamma])

        # Execute the move
        if move == 'left':
            position -= 1
        elif move == 'right':
            position += 1
        else:  # if move == 'fall'
            position = np.random.randint(0, high=L+1)
            time_count += t_3d  # Add the time penalty for falling off

        time_count += 10**(-5)  # Increment time count for each move or fall

    return time_count

# Simulate for different L values and numbers of simulations, and calculate average times
results = {}

for num_simulations in num_simulations_values:
    average_times = []
    for L in L_values:
        times = [simulate_trajectory(L, gamma, t_3d, target_position) for _ in range(num_simulations)]
        average_times.append(np.mean(times))
    results[num_simulations] = average_times

# Plotting the results
plt.figure(figsize=(12, 8))

colors = ['b', 'g', 'r', 'k']  # Different colors for different simulation numbers

for i, num_simulations in enumerate(num_simulations_values):
    plt.plot(L_values, results[num_simulations], marker='o', color=colors[i], linestyle='-', label=f'{num_simulations} simulations')

plt.title('Average Time to Reach Target for Different DNA Lengths (L) and Number of Simulations')
plt.xlabel('DNA Length (L)')
plt.ylabel('Average Time to Reach Target (seconds)')
plt.legend()
plt.grid(True)
plt.show()
```

Appendix B: 2D Simulation Code

The following python code was followed for setting up the 2D lattice and the whole process of plotting MFPT for different Lattice sizes.

```
import numpy as np
import matplotlib.pyplot as plt

# Function to simulate trajectory on a 2D lattice with a specific target
def simulate_trajectory_2d(L, gamma, t_3d, target_position):
    position = (np.random.randint(0, L), np.random.randint(0, L))
    time_count = 0

    while position != target_position:
        move = np.random.choice(['left', 'right', 'up', 'down', 'fall'], p=[(1-gamma)/4]*4 + [gamma])

        if move == 'left':
            position = (max(position[0] - 1, 0), position[1])
        elif move == 'right':
            position = (min(position[0] + 1, L-1), position[1])
        elif move == 'up':
            position = (position[0], max(position[1] - 1, 0))
        elif move == 'down':
            position = (position[0], min(position[1] + 1, L-1))
        elif move == 'fall':
            position = (np.random.randint(0, L), np.random.randint(0, L))
            time_count += t_3d

        time_count += 10**(-5)

    return time_count

# Parameters
gamma = 0.08
t_3d = 10**(-4) - 10**(-5)
target_position = (10, 10)
num_simulations_values = [200, 300, 400]  # Different numbers of simulations to try

# Varying L values
L_values = np.arange(100, 200, 20)  # Lattice sizes from 100 to 150 with increment of 5

# Plotting setup
plt.figure(figsize=(10, 6))
colors = ['b', 'g', 'k']  # Different colors for different simulation numbers

# Run simulations for each number of simulations and each L value
for i, num_simulations in enumerate(num_simulations_values):
    average_times = []
    for L in L_values:
        times = [simulate_trajectory_2d(L, gamma, t_3d, target_position) for _ in range(num_simulations)]
        average_time = np.mean(times)
        average_times.append(average_time)

    # Plot for each number of simulations
    plt.plot(L_values, average_times, marker='o', color=colors[i], label=f'{num_simulations} simulations')

```

```
# Final plot adjustments
plt.xlabel('Lattice Size (L)')
plt.ylabel('Average Time to Reach Target (seconds)')
plt.title('Average Time vs Lattice Size for Different Numbers of Simulations')
plt.legend()
plt.grid(True)
plt.show()
```

References

[1] Leonid Mirny et al. "How a protein searches for its site on DNA: the mechanism of facilitated diffusion". In: *Journal of Physics A: Mathematical and Theoretical* 42.43 (Oct. 2009), p. 434013. ISSN: 1751-8113. DOI: 10.1088/1751-8113/42/43/434013.

[2] M Sheinman et al. "Classes of fast and specific search mechanisms for proteins on DNA". In: *Reports on Progress in Physics* 75.2 (Feb. 2012), p. 026601. ISSN: 0034-4885. DOI: 10.1088/0034-4885/75/2/026601.

[3] Ori Hachmo and Ariel Amir. "Conditional probability as found in nature: Facilitated diffusion". In: *American Journal of Physics* 91.8 (2023). ISSN: 0002-9505. DOI: 10.1119/5.0123866.

[4] A.E. Lindsay. *Modelling Diffusion and Capture*. Tech. rep. University of Notre Dame, June 2017.

[5] O. Bénichou et al. "Mean First-Passage Time of Surface-Mediated Diffusion in Spherical Domains". In: *Journal of Statistical Physics* 142.4 (Feb. 2011), pp. 657–685. ISSN: 0022-4715. DOI: 10.1007/s10955-011-0138-6.