Summer Internship Project
Report

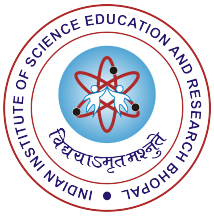
Protein Searches for Target on DNA - A Theoretical Approach

Submitted by

Soumik Nath
BS-MS, Department of Chemistry
IISER Bhopal

Under the guidance of

Dr. Rati Sharma
Department of Chemistry



Department of Chemistry
Indian Institute of Science Education and Research
Bhopal

Summer Internship 2025

Contents

Chapter 1

Objective

During my internship, I explored the problem of "speed-selectivity paradox" in protein-DNA interactions. This can be considered as a puzzle where proteins seem to face conflicting demands of moving quickly while accurately finding their target sites on DNA. My goal was to use and understand the discrete-stochastic model discussed in the research article titled "Speed-Selectivity Paradox in the Protein Search for Targets on DNA: Is It Real or Not?" from Prof. A.B. Kolomeisky's lab and to finally determine the role of target position and DNA length on search time.

To tackle this, I:

1. **Reproduced the paper's key results** using Monte Carlo simulations, mimicking how proteins randomly hop and slide along DNA.
2. **Derived the analytical equations** from the paper to understand the math.
3. **Compared theory with simulations** by plotting both results, showing how closely they match, proving the paradox vanishes when using discrete models instead of oversimplified continuum ones.

Also to build my understanding of stochastic modeling approaches, I first implemented the Gillespie algorithm for the Monte Carlo simulation of Michaelis-Menten enzyme kinetics.

Chapter 2

Gillespie Algorithm

The Gillespie algorithm, also known as the Stochastic Simulation Algorithm (SSA) is a method for simulating chemical reactions, biological systems, etc. while accounting for their inherent randomness. It tracks two stochastic elements:

1. Which reaction occurs next (selected probabilistically based on reaction rates)
2. When it occurs (with exponentially distributed waiting times)

Key Steps:

1. **Initialization:** Initialize the number of molecules in the system, reaction constants, random number generators and any other parameter if needed.
2. **Monte Carlo step:** Generate random numbers to determine the next reaction to occur as well as the time interval. To determine the next reaction event, the following steps are performed:
   (a) Generate two random numbers (r_1 and r_2) uniformly distributed in [0,1].
   (b) Compute the time increment τ until the next reaction:
   $$\tau = \frac{1}{a_0} \ln\left(\frac{1}{r_1}\right)$$
   where $a_0 = \sum a_r$ is the sum of all reaction propensities. Propensity (a_r) is the probability per unit time that a specific reaction occurs, which is basically proportional to the rate of reaction. The next reaction occurs at time $t + \tau$.

(c) Compute which reaction occurs at time $t + \tau$. Find j such that:

$$\sum_{i=1}^{j-1} a_i < r_2 a_0 \leq \sum_{i=1}^{j} a_i$$

Since the above condition is being followed, the j^{th} reaction takes place.

3. **Update:** Increase the time by the randomly generated time in Monte Carlo step. Update the molecule count based on the reaction that occurred.

4. **Iterate:** Go back to step 2 unless the number of reactants is zero or the simulation time has been exceeded.

Michaelis-Menten Kinetics

In order to understand the Gillespie algorithm properly, I wrote a Monte Carlo simulation code for Michaelis-Menten Kinetics in order to assess the algorithm's accuracy and also to get a gist of what it can do.

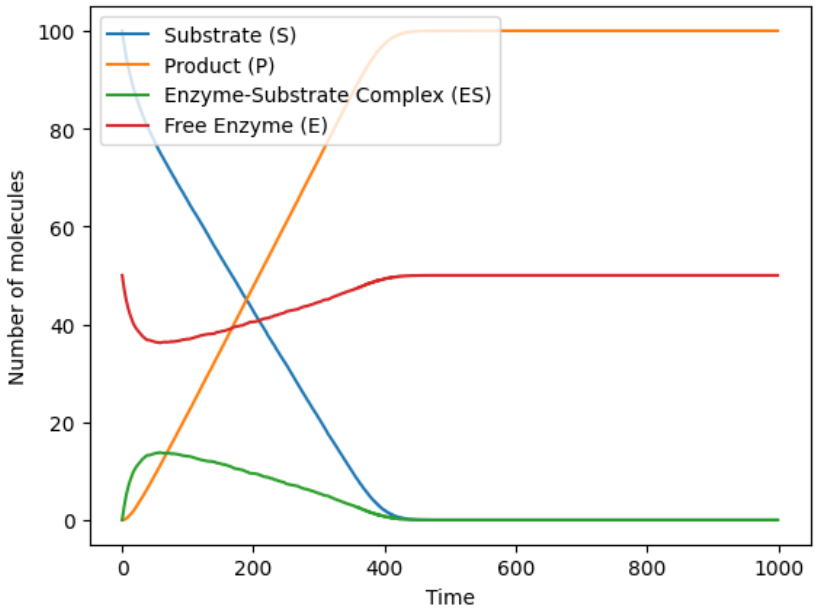


Figure 2.1: *Time evolution of molecular species in the Michaelis-Menten system. Substrate (S) decreases as product (P) accumulates, while the ES complex builds up and then disappears as the reaction proceeds. Free enzyme (E) is initially consumed then regenerated.*

Simulation Steps

- **Initialization**: The system starts with defined initial counts of enzyme (E), substrate (S), and zero concentrations of the enzyme-substrate complex (ES) and product (P).
- **Dynamics**: The system evolves through three primary reactions:
    1. **Binding**: Enzyme (E) and substrate (S) combine to form the complex (ES) with rate constant k_1.
    2. **Dissociation**: The complex (ES) can dissociate back into E and S with rate constant k_2.
    3. **Catalysis**: The complex (ES) converts to enzyme (E) and product (P) with rate constant k_3.
- **Termination**: The simulation stops when the substrate is fully depleted.

Validation and Comparison

- The simulation outputs (plotted trajectories of S, P, ES, and E) align with the expected behavior of Michaelis-Menten kinetics, confirming the validity of the stochastic approach.

The simulation begins with 50 enzyme molecules (E) and 100 substrate molecules (S), with no initial ES complex or product (P). As shown in Figure (2.1), S decreases over time while P accumulates, following classic Michaelis-Menten kinetics. The ES complex forms rapidly as E binds S (rate $k_1 = 0.001$), then gradually dissociates either back to $E + S$ ($k_2 = 0.1$) or forward to $E + P$ ($k_3 = 0.1$). Free enzyme is initially consumed but later regenerated as the reaction progresses, until all substrate is depleted. The stochastic simulation (1000 runs) matches the expected results.

Chapter 3

Introduction to Protein-DNA Target Search Processes

Protein-DNA interactions are crucial for many biological processes. These interactions often begin when a protein locates and binds to a specific DNA sequence, triggering a series of biochemical reactions that regulate cellular functions. Over the past few decades, scientists have studied this process known as protein search and binding using both experiments and theoretical models. While we've learned a lot, the exact dynamics remain unclear, and debates continue.

One of the confounding findings arises from initial experiments with the Lac repressor protein, which interacts with its target DNA sequence considerably quicker (approximately 100-1000 times quicker) than expected from conventional 3D diffusion theory. In an attempt to explain this, researchers developed alternative theories, with ***facilitated diffusion*** being the most accepted. According to this theory, during the search process, the protein slides along the DNA, undergoing 1D Brownian motion in addition to 3D diffusion into the bulk solution.

To keep the model simple, the study we referred to treated the protein as a single molecule. By focusing on this simplified version, the researchers were able to better understand how proteins quickly find their target sites on DNA.

To begin, let's discuss the earlier model, its limitations, and how these shortcomings ultimately led to the development of the current model. The time taken to find the target can be estimated using the following equation based on the classical works of Berg, Winter, and von Hippel.

$$\tau_s = \frac{L}{\lambda}(\tau_{1D} + \tau_{3D}) \tag{3.1}$$

with $\tau_{1D} = \frac{\lambda^2}{2D_1}$ and $\tau_{3D} = \frac{x^2}{2D_3}$, where L is the total contour length of DNA, λ is the average length of DNA that the protein molecule scans during each search cycle , x is the average distance traveled by the protein in the solution before binding to DNA, D_1 is a diffusion constant to move along the DNA, and D_3 is the protein's bulk diffusion constant. Although the above theoretical method has proven to be useful in terms of explaining the dynamics to a great extent, there is an increasing number of experimental and theoretical studies that challenge existing views.

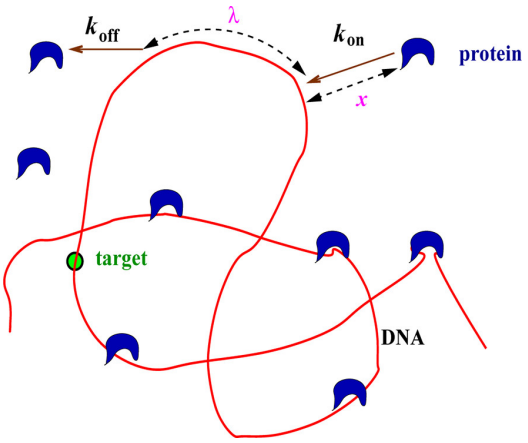


Figure 3.1: *This illustration shows how proteins search for specific targets on DNA. A protein diffuses through solution (traveling an average distance x) before binding to the DNA (with rate k_{on}). It then scans along the DNA (average distance λ) before unbinding (rate k_{off}) and repeating the process. The search succeeds when the protein encounters and binds to its target sequence.*

One of the most surprising and puzzling findings in protein-DNA interactions is the "**speed-selectivity paradox**". Since DNA is made of varying sequences, proteins bind to it with different strengths depending on the genetic code. This variation in binding energy can be measured by a parameter called σ. Interestingly, these sequence-dependent energy fluctuations actually slow down how fast proteins move along DNA. The reason? The protein isn't sliding smoothly. It's essentially stumbling through a constantly changing energy landscape, like walking on uneven ground. This randomness in binding forces makes diffusion much less efficient.

$$D_1 \simeq \exp\left[-\left(\frac{\sigma}{k_{\mathrm{B}}T}\right)^2\right] \tag{3.2}$$

The 'speed-selectivity paradox' arises because proteins need two conflicting traits:

1. **Fast sliding** (requiring weak, nonspecific DNA binding, with energy fluctuations $\sigma < 1 - 2k_BT$).

2. **Stable binding** (requiring strong, specific attachment, with $\sigma > 5k_BT$).

To resolve this, a two-state model was proposed. which discussed the existence of the protein molecule bound to DNA to exist in two different conformational states:

- **Search mode**: Protein slides quickly along DNA.

- **Recognition mode**: Protein pauses to check for target sequences, binding tightly.

Problems with the model:

- The recognition state is oddly assumed to have higher free energy than the search state, even though it binds more strongly (which should lower energy).

- Experiments show switching rates ($\simeq 1s^{-1}$) are far slower than needed ($\simeq 1000s^{-1}$) to match observed search speeds.

The speed-selectivity paradox arises from a key flaw in current protein-search theories: they rely on continuum models (smooth approximations) to describe what's actually a discrete process (proteins binding/unbinding and hopping between DNA sites). One could easily see this by analyzing eq (3.1) in the limit of very small diffusion constant on DNA, $D_1 \to 0$, which is the case for many experimental systems. This model predicts infinite search times which makes no sense, since proteins can still find targets via 3D diffusion (just slower). This error occurs because continuum models only work when the scanning range $\lambda >> a$.

In order to solve the above discussed problems, Kolomeisky and group developed a *discrete-state stochastic model* that accurately describes how proteins search DNA at the single-molecule level. The model explains fast search times seen in experiments and matches Monte Carlo simulations.

Chapter 4

Theoretical Methods

4.1 Discrete-state stochastic model

Since continuum models give misleading predictions for protein-DNA interactions, a discrete-state stochastic model (Figure 4.1) has been used to better capture the search process. Here's how it works:

1. **Protein States:**
    - The protein can either be **free in solution** (state 0) or **bound to any site** on the DNA (sites $i = 1, 2, ..., L$).
    - One of these sites ($i = m$) is the **target** the protein needs to find.
2. **Movement Rules:**
    - **On DNA:** The protein hops left or right at rate u (like a random walk).
    - **Unbinding:** It can detach from DNA at rate k_{off}.
    - **Rebinding:** From solution, it reattaches to any DNA site with equal probability (total rate k_{on}).

To describe the target search dynamics, use backward master equations to study the temporal evolution of the first-passage probabilities. First-passage probability, in the context of stochastic processes, refers to the probability that a randomly evolving system will reach a specific state or region (the target) for the first time at a given time.

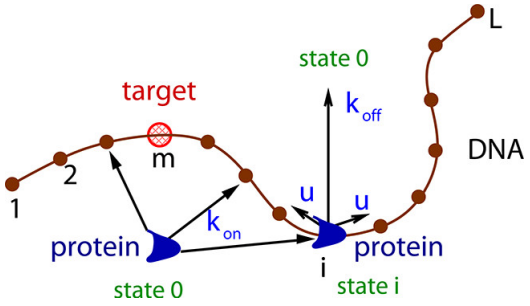


Figure 4.1: *This diagram illustrates the discrete-state model of how proteins search for targets on DNA. Imagine the DNA as a chain with $L-1$ (hopping between sites at rate u) or detach completely (rate k_{off}). When free in solution, it can rebind to any DNA site equally (total binding rate k_{on}). The search successfully ends when the protein lands on and recognizes its target site at position m.*

4.2 Backward Master Equation

In Backward Master Equation (BME), the terminal condition (target) is known and we work to determine how the outcome depends on the initial state. By focusing only on the paths that eventually reach the target, BME avoids unnecessary computations on irrelevant states, significantly improving efficiency.

In this section, we'll walk through the math step-by-step to understand how proteins randomly hop along DNA until they find their target. We'll end with the equation (4.7) which will be used later on.

4.2.1 Derivation of Backward Master Equation

- Protein moves to the right and to the left with rate u. It can also dissociate from its own site with rate k_{off}.
- We define $F_n(t)$ as the chance of the protein finding its target at site m by time t, starting from position n.
- We can write the following equation for this probability:

$$F_n(t+\tau) = \frac{u}{2u+k_{off}}[F_{n+1}(t) + F_{n-1}(t)] + \frac{k_{off}}{2u+k_{off}}F_0(t) \qquad (4.1)$$

- This is a result of the balance of probabilities. With probability $\frac{u}{2u+k_{off}}$, the protein hops right to $n+1$ and continues its search for the target from there (now with probability F_{n+1}).
- With equal probability $\frac{u}{2u+k_{off}}$, it hops left to $n-1$ (with probability to find the target now being F_{n-1}).
- With probability $\frac{k_{off}}{2u+k_{off}}$, it detaches completely and must restart from solution (with probability F_0).
- τ is the average time the protein spends at a particular site before it moves on to the next site.

$$\tau = \frac{1}{2u + k_{off}} \tag{4.2}$$

Multiply both sides of equation (4.1) by $(2u + k_{off})$

$$(2u + k_{off})F_n(t + \tau) = u[F_{n+1}(t) + F_{n-1}(t)] + k_{off}F_0(t) \tag{4.3}$$

Assuming that times are long and $\tau << t$, we can taylor expand the left side of equation (4.3). The taylor expansion has been done considering that we are deriving a first-order backward master equation.

$$(2u + k_{off})(F_n(t) + \tau \cdot \frac{dF_n(t)}{dt}) = u[F_{n+1}(t) + F_{n-1}(t)] + k_{off}F_0(t) \tag{4.4}$$

$$(2u + k_{off}) \cdot \tau \cdot \frac{dF_n(t)}{dt}) = u[F_{n+1}(t) + F_{n-1}(t)] + k_{off}F_0(t) - (2u + k_{off})F_n(t) \tag{4.5}$$

Substituting the value of τ from equation (4.2)

$$(2u+k_{off})\cdot\frac{1}{2u + k_{off}}\cdot\frac{dF_n(t)}{dt}) = u[F_{n+1}(t)+F_{n-1}(t)]+k_{off}F_0(t)-(2u+k_{off})F_n(t) \tag{4.6}$$

$$\boxed{\frac{dF_n(t)}{dt} = u[F_{n+1}(t) + F_{n-1}(t)] + k_{off}F_0(t) - (2u + k_{off})F_n(t)} \tag{4.7}$$

Finally, we get equation (4.7), which we wanted to derive. Similarly, rest of the backward master equations can be derived.

4.3 Calculating the Target Search Time

The equation (4.7) derived in the previous section tells us the temporal evolution of the first-passage probabilities to reach the target on site m at time t for the first time if at $t = 0$ the protein was at the state $n(n = 0, 1, ..., L)$. The equation was derived for sites $2 \leq n \leq L-1$, while for sites at the DNA ends ($n = 1$ and $n = L$) we have

$$\frac{dF_1(t)}{dt} = uF_2(t) + k_{off}F_0(t) - (u + k_{off})F_1(t) \tag{4.8}$$

$$\frac{dF_L(t)}{dt} = uF_{L-1}(t) + k_{off}F_0(t) - (u + k_{off})F_L(t) \tag{4.9}$$

In equations (4.8) and (4.9), we're looking at the ends of the DNA chain. Here, the protein can only jump in one direction, i.e. it can't hop both left and right like it could in equation (4.7).
The backward master equation is different if the protein molecule starts from the solution, $n = 0$,

$$\frac{dF_0(t)}{dt} = \frac{k_{on}}{L}\sum_{n=1}^{L} F_n(t) - k_{on}F_0(t) \tag{4.10}$$

In equation (4.10) we used the fact that the rate to bind to any given site on DNA is $\frac{k_{on}}{L}$, and the total rate of binding to DNA is equal to k_{on}. In addition, initial conditions require that $F_m(t) = \delta(t)$ and $F_{n\neq m}(t = 0) = 0$. These equations can be analyzed by introducing Laplace transformations of first-passage probability functions, $\widetilde{F_n(s)} \equiv \int_0^\infty e^{-st}F_n(t)\,dt$.

$$\mathcal{L}[\frac{dF_n(t)}{dt}] = \int_0^\infty e^{-st}\frac{dF_n(t)}{dt}\,dt \tag{4.11}$$

Using the integration by parts formula
Let $u = e^{-st}$ and $dv = \frac{dF_n(t)}{dt}dt$. Then:
$du = -se^{-st}dt$, $v = F_n(t)$.
Applying integration by parts:

$$\int u\,dv = uv - \int v\,du,$$

we get:

$$\mathcal{L}\left[\frac{dF_n(t)}{dt}\right] = e^{-st}F_n(t)\big|_0^\infty + s\int_0^\infty e^{-st}F_n(t)\,dt$$

At $t = \infty$: $e^{-st} \to 0$ (if $s > 0$, which is true for Laplace transforms). $F_n(\infty)$ is finite (since $F_n(t)$ is a probability and must be bounded). Thus, $e^{-st}F_n(t)\big|_{t=\infty} = 0$.

At $t = 0$: $e^{-st} = 1$. $F_n(0) = 0$ for all $n \neq m$ (initial condition: the target is not yet found).

Thus:

$$e^{-st}F_n(t)\big|_0^\infty = 0 - F_n(0) = 0$$

Substituting back:

$$\mathcal{L}\left[\frac{dF_n(t)}{dt}\right] = s\int_0^\infty e^{-st}F_n(t)\,dt = s\widetilde{F_n(s)}$$

Then, backward master equations (Eqs. 4.7, 4.8, 4.9, and 4.10) can be written as a set of simpler algebraic expressions.

$$(s + 2u + k_{off})\widetilde{F_n(s)} = u[\widetilde{F_{n+1}(s)} + \widetilde{F_{n-1}(s)}] + k_{off}\widetilde{F_0(s)} \tag{4.12}$$

$$(s + u + k_{off})\widetilde{F_1(s)} = u\widetilde{F_2(s)} + k_{off}\widetilde{F_0(s)} \tag{4.13}$$

$$(s + u + k_{off})\widetilde{F_L(s)} = u\widetilde{F_{L-1}(s)} + k_{off}\widetilde{F_0(s)} \tag{4.14}$$

$$(s + k_{on})\widetilde{F_0(s)} = \frac{k_{on}}{L}\sum_{n=1}^{L}\widetilde{F_n(s)} \tag{4.15}$$

These equations are solved by assuming that the general form of the solution is $\widetilde{F_n(s)} = Ay^n + B$, and using boundary and initial conditions, it yields

The Laplace-transformed equation (from Eq. 4.12) is:

$$(s + 2u + k_{off})\widetilde{F_n(s)} = u[\widetilde{F_{n+1}(s)} + \widetilde{F_{n-1}(s)}] + k_{off}\widetilde{F_0(s)}$$

First, consider only the homogeneous part

$$(s + 2u + k_{off})\widetilde{F_n(s)} = u[\widetilde{F_{n+1}(s)} + \widetilde{F_{n-1}(s)}]$$

Assume a trial solution of the form $\widetilde{F_n(s)} = y^n$. Substituting:

$$(s + 2u + k_{\text{off}})y^n = u(y^{n+1} + y^{n-1})$$

Divide by y^{n-1}:

$$(s + 2u + k_{\text{off}})y = u(y^2 + 1)$$

Rearrange into the characteristic equation:

$$uy^2 - (s + 2u + k_{\text{off}})y + u = 0$$

The quadratic equation has roots:

$$\boxed{y = \frac{(s + 2u + k_{\text{off}}) \pm \sqrt{(s + 2u + k_{\text{off}})^2 - 4u^2}}{2u}}$$

For $s > 0$, the discriminant is positive, yielding two distinct real roots: $y_1 = \frac{(s+2u+k_{\text{off}})-\sqrt{\Delta}}{2u}$, $y_2 = \frac{(s+2u+k_{\text{off}})+\sqrt{\Delta}}{2u}$.

The general homogeneous solution is:

$$\widetilde{F_n(s)} = C_1 y^n + C_2 y^{-n}$$

The full equation includes a factor of $k_{\text{off}}\widetilde{F_0(s)}$. To account for this, assume a constant B.

Full solution:

$$\widetilde{F_n(s)} = A(y^n + y^{-n}) + B$$

At $n = m$, $\widetilde{F_m(s)} = 1$. So

$$A(y^m + y^{-m}) + B = 1$$

$$A = \frac{1 - B}{y^m + y^{-m}}$$

This leads to the final forms (Eqs. 4.16 and 4.17):
For $1 \leq n \leq m$:

$$\widetilde{F_n(s)} = \frac{(1 - B)(y^n + y^{-n})}{y^m + y^{-m}} + B \tag{4.16}$$

For $m \leq n \leq L$:

$$\widetilde{F_n(s)} = \frac{(1-B)(y^{1+L-n} + y^{n-L-1})}{y^{1+L-m} + y^{m-L-1}} + B \tag{4.17}$$

Substituting the general solution of $\widetilde{F_n(s)}$ in Eq. (4.12):

$$(s + 2u + k_{off})(Ay^n + B) = u[Ay^{n+1} + B + Ay^{n-1} + B] + k_{off}\widetilde{F_0(s)}$$

Now consider only the constant terms(i.e. terms independent of n):

$$(s + 2u + k_{off})B = 2uB + k_{off}\widetilde{F_0(s)}$$

This leads to the final form of B:

$$\boxed{B = \frac{k_{off}\widetilde{F_0(s)}}{k_{off} + s}} \tag{4.18}$$

The sum of first passage probabilities over all DNA sites:

$$\begin{aligned}\sum_{n=1}^{L} \widetilde{F_n(s)} = &\sum_{n=1}^{m-1} \left[\frac{(1-B)(y^n + y^{-n})}{y^m + y^{-m}} + B\right] + 1 \\ &+ \sum_{n=m+1}^{L} \left[\frac{(1-B)(y^{1+L-n} + y^{n-L-1})}{y^{1+L-m} + y^{m-L-1}} + B\right]\end{aligned} \tag{4.19}$$

1. **For** $1 \leq n < m$**:**

$$\sum_{n=1}^{m-1} y^n = y^1 + y^2 + \cdots + y^{(m-1)} = \frac{y(1 - y^{(m-1)})}{1-y} = \frac{y - y^m}{1-y}$$

$$\sum_{n=1}^{m-1} y^{-n} = y^{-1} + y^{-2} + \cdots + y^{-(m-1)} = y^{-1}\frac{1 - y^{-(m-1)}}{1 - y^{-1}} = \frac{1 - y^{1-m}}{y-1}$$

$$\sum_{n=1}^{m-1} \left(y^n + y^{-n}\right) = \frac{y - y^m}{1-y} + \frac{1 - y^{1-m}}{y-1}$$

$$\sum_{n=1}^{m-1} \left[\frac{y^n + y^{-n}}{y^m + y^{-m}}\right] = \frac{1}{y^m + y^{-m}} \left[\frac{y - y^m}{1-y} + \frac{1 - y^{1-m}}{y-1}\right] \tag{4.20}$$

2. **For $m < n \leq L$:** Substitute $n = m + k$ and express the sum in terms of k:

$$\begin{aligned}
\sum_{n=m+1}^{L} y^{1+L-n} &= \sum_{k=1}^{L-m} y^{1+L-m-k} \\
&= y^{L-m+1} \sum_{k=1}^{L-m} y^{-k} \\
&= y^{L-m+1} \left(y^{-1} + y^{-2} + \cdots + y^{-(L-m)}\right) \\
&= y^{L-m+1} \cdot \frac{y^{-1}(1 - y^{-(L-m)})}{1 - y^{-1}} \\
&= \frac{y^{L-m+1} - y}{y - 1}
\end{aligned}$$

$$\begin{aligned}
\sum_{n=m+1}^{L} y^{n-L-1} &= \sum_{k=1}^{L-m} y^{m+k-L-1} \\
&= y^{m-L-1} \sum_{k=1}^{L-m} y^{k} \\
&= y^{m-L-1} \left(y + y^2 + \cdots + y^{L-m}\right) \\
&= y^{m-L-1} \cdot \frac{y(1 - y^{L-m})}{1 - y} \\
&= \frac{y^{m-L}(1 - y^{L-m})}{1 - y} \\
&= \frac{y^{m-L} - 1}{1 - y}
\end{aligned}$$

$$\sum_{n=m+1}^{L} \left[\frac{y^{1+L-n} + y^{n-L-1}}{y^{1+L-m} + y^{m-L-1}}\right] = \frac{1}{y^{1+L-m} + y^{m-L-1}} \left[\frac{y^{L-m+1} - y}{y - 1} + \frac{y^{m-L} - 1}{1 - y}\right] \tag{4.21}$$

Use values from Eq. (4.20) and (4.21) and substitute them in Eq. (4.19):

$$\sum_{n=1}^{L} \widetilde{F_n(s)} = 1 + B(L-1) + \frac{(1-B)}{y^m + y^{-m}} \left[\frac{y - y^m}{1-y} + \frac{1 - y^{1-m}}{y-1} \right] + \frac{(1-B)}{y^{1+L-m} + y^{m-L-1}} \left[\frac{y^{L-m+1} - y}{y-1} + \frac{y^{m-L} - 1}{1-y} \right]$$

The above equation can be written in the form $\sum_{n=1}^{L} \widetilde{F_n}(s) = B \cdot L + (1 - B) \cdot S(s)$ where the new auxiliary function S(s) is given by

$$\boxed{S(s) = 1 + \left[\frac{1}{y^m + y^{-m}} \left(\frac{y - y^m}{1-y} + \frac{1 - y^{1-m}}{y-1} \right) \right] + \left[\frac{1}{y^{1+L-m} + y^{m-L-1}} \left(\frac{y^{L-m+1} - y}{y-1} + \frac{y^{m-L} - 1}{1-y} \right) \right]} \tag{4.22}$$

Consider Eq. (4.15):

$$(s + k_{on})\widetilde{F_0(s)} = \frac{k_{on}}{L} \sum_{n=1}^{L} \widetilde{F_n(s)}$$

$$L(s + k_{on})\widetilde{F_0(s)} = k_{on}[B \cdot L + (1 - B) \cdot S(s)]$$

Multiply both sides by $(s + k_{off})$:

$$L(s + k_{off})(s + k_{on})\widetilde{F_0(s)} = k_{on}(s + k_{off})[B \cdot L + (1 - B) \cdot S(s)]$$

$$L(s + k_{off})(s + k_{on})\widetilde{F_0(s)} = k_{on}(s + k_{off})B \cdot L + k_{on}(s + k_{off})(1 - B) \cdot S(s)]$$

Use the value of B from Eq. (4.18) and substitute it in the equation above:

$$L(s+k_{off})(s+k_{on})\widetilde{F_0(s)} = k_{on}k_{off} \cdot L \cdot \widetilde{F_0(s)} + k_{on}(s+k_{off})(s+k_{off} - k_{off}\widetilde{F_0(s)}) \cdot S(s)$$

$$L(s+k_{off})(s+k_{on})\widetilde{F_0(s)} + k_{on}k_{off}S(s)\widetilde{F_0(s)} - k_{on}k_{off}L\widetilde{F_0(s)} = k_{on}(s+k_{off})S(s)$$

$$L(s+k_{off})(s+k_{on})\widetilde{F_0(s)} + k_{on}k_{off}S(s)\widetilde{F_0(s)} - k_{on}k_{off}L\widetilde{F_0(s)} = k_{on}(s+k_{off})S(s)$$

$$\widetilde{F_0(s)}[L(s+k_{off})(s+k_{on}) + k_{on}k_{off}S(s) - k_{on}k_{off}L] = k_{on}(s+k_{off})S(s)$$

$$\widetilde{F_0(s)}[Ls^2 + k_{off}\cdot s\cdot L + k_{on}\cdot s\cdot L + k_{on}k_{off}L + k_{on}k_{off}S(s) - k_{on}k_{off}L] = k_{on}(s+k_{off})S(s)$$

$$\widetilde{F_0(s)}[Ls(k_{off}+k_{on}+s) + k_{on}k_{off}S(s)] = k_{on}(s+k_{off})S(s)$$

$$\boxed{\widetilde{F_0(s)} = \frac{k_{on}(s+k_{off})S(s)}{Ls(k_{off}+k_{on}+s) + k_{on}k_{off}S(s)}} \tag{4.23}$$

Verification at $s = 0$: Let us verify the expression by setting $s = 0$. Substituting $s = 0$ into the equation:

$$\widetilde{F_0(0)} = \frac{k_{on}(0+k_{off})S(0)}{L\cdot 0\cdot(k_{off}+k_{on}+0) + k_{on}k_{off}S(0)} = \frac{k_{on}k_{off}S(0)}{k_{on}k_{off}S(0)} = 1$$

This confirms that $\widetilde{F_0(0)} = 1$ as expected, as taking $(s = 0)$ corresponds to the infinite-time limit $(t = \infty)$ in the time domain. This is because the Laplace transform $\int_0^\infty e^{-st}F(t)dt$ becomes simply $\int_0^\infty F(t)dt$ when $s = 0$.

The average time to find the target starting from the solution, T_0, can be easily found using the following equality

$$T_0 = -\left.\frac{\partial\widetilde{F_0(s)}}{\partial s}\right|_{s=0}$$

$$\begin{aligned}\text{Let, } N(s) &= k_{\text{on}}(s+k_{\text{off}})S(s)\\ D(s) &= Ls(k_{\text{off}}+k_{\text{on}}+s) + k_{\text{on}}k_{\text{off}}S(s)\end{aligned}$$

Thus:

$$\widetilde{F_0(s)} = \frac{N(s)}{D(s)}$$

The derivative of a ratio is:

$$\frac{d}{ds}\left(\frac{N(s)}{D(s)}\right) = \frac{N'(s)D(s) - N(s)D'(s)}{D(s)^2}$$

$$\left.\frac{\partial \widetilde{F_0(s)}}{\partial s}\right|_{s=0} = \left.\frac{N'(0)D(0) - N(0)D'(0)}{D(0)^2}\right|_{s=0} = \left.\frac{N'(0)}{D(0)} - \frac{D'(0)}{D(0)}\right|_{s=0}$$

Numerator $N(s)$:

$$N(s) = k_{\text{on}}(k_{\text{off}} + s)S(s), \quad N(0) = k_{\text{on}}k_{\text{off}}S(0)$$

$$N'(s) = k_{\text{on}}S(s) + k_{\text{on}}(k_{\text{off}} + s)S'(s)$$

At $s = 0$:

$$N'(0) = k_{\text{on}}S(0) + k_{\text{on}}k_{\text{off}}S'(0)$$

Denominator $D(s)$:

$$D(s) = Ls(k_{\text{off}} + k_{\text{on}} + s) + k_{\text{off}}k_{\text{on}}S(s), \quad D(0) = k_{\text{off}}k_{\text{on}}S(0)$$

$$D'(s) = L(k_{\text{off}} + k_{\text{on}} + s) + Ls(1) + k_{\text{off}}k_{\text{on}}S'(s)$$

At $s = 0$:

$$D'(0) = L(k_{\text{off}} + k_{\text{on}}) + k_{\text{off}}k_{\text{on}}S'(0)$$

Substitute $N'(0)$ and $D'(0)$:

$$\left.\frac{\partial \widetilde{F_0(s)}}{\partial s}\right|_{s=0} = \frac{k_{\text{on}}S(0) + k_{\text{on}}k_{\text{off}}S'(0)}{k_{\text{off}}k_{\text{on}}S(0)} - \frac{L(k_{\text{off}} + k_{\text{on}}) + k_{\text{off}}k_{\text{on}}S'(0)}{k_{\text{off}}k_{\text{on}}S(0)}$$

Simplify:

$$= \frac{1}{k_{\text{off}}} + \frac{S'(0)}{S(0)} - \frac{L(k_{\text{off}} + k_{\text{on}})}{k_{\text{off}}k_{\text{on}}S(0)} - \frac{S'(0)}{S(0)}$$

The $S'(0)$ terms cancel, leaving:

$$= \frac{1}{k_{\text{off}}} - \frac{L(k_{\text{off}} + k_{\text{on}})}{k_{\text{off}}k_{\text{on}}S(0)}$$

The mean first-passage time is:

$$T_0 = -\frac{\partial \widetilde{F_0(s)}}{\partial s}\bigg|_{s=0} = \frac{L(k_{\text{off}} + k_{\text{on}})}{k_{\text{off}} k_{\text{on}} S(0)} - \frac{1}{k_{\text{off}}}$$

Factor out $\frac{1}{k_{\text{off}}}$:

$$T_0 = \frac{1}{k_{\text{off}}} \left(\frac{L(k_{\text{off}} + k_{\text{on}})}{k_{\text{on}} S(0)} - 1 \right)$$

Simplify:

$$T_0 = \frac{1}{k_{\text{on}}} \frac{L}{S(0)} + \frac{1}{k_{\text{off}}} \left(\frac{L}{S(0)} - 1 \right)$$

$$\boxed{T_0 = \frac{1}{k_{\text{on}}} \frac{L}{S(0)} + \frac{1}{k_{\text{off}}} \frac{L - S(0)}{S(0)}} \tag{4.24}$$

According to Eq. (4.15):

$$(s + k_{on}) \widetilde{F_0(s)} = \frac{k_{on}}{L} \sum_{n=1}^{L} \widetilde{F_n(s)}$$

$$\sum_{n=1}^{L} \widetilde{F_n(s)} = \frac{L(s + k_{\text{on}})}{k_{\text{on}}} \widetilde{F_0(s)}$$

Take the derivative with respect to s:

$$\frac{d}{ds} \sum_{n=1}^{L} \widetilde{F_n(s)} = \frac{L}{k_{\text{on}}} \left(\widetilde{F_0(s)} + (s + k_{\text{on}}) \widetilde{F_0'(s)} \right)$$

Evaluate at $s = 0$:

$$\frac{d}{ds} \sum_{n=1}^{L} \widetilde{F_n(s)} \bigg|_{s=0} = \frac{L}{k_{\text{on}}} \left(\widetilde{F_0(0)} + k_{\text{on}} \widetilde{F_0'(0)} \right)$$

Since $\widetilde{F_0(0)} = 1$ (normalization), and $\widetilde{F_0'(0)} = -T_0$:

$$\frac{d}{ds} \sum_{n=1}^{L} \widetilde{F_n(s)} \bigg|_{s=0} = \frac{L}{k_{\text{on}}} (1 - k_{\text{on}} T_0)$$

First-passage times to reach the target at the site m starting with equal probability on any site on the DNA chain can be computed from

$$T_m \equiv -\frac{1}{L}\,\frac{d}{ds}\sum_{n=1}^{L}\widetilde{F_n(s)}\bigg|_{s=0}$$

From the definition of T_m:

$$T_m = -\frac{1}{L}\left(\frac{L}{k_{\text{on}}}(1-k_{\text{on}}T_0)\right) = T_0 - \frac{1}{k_{\text{on}}}$$

Thus:

$$T_m = T_0 - \frac{1}{k_{\text{on}}}$$

$$\boxed{T_0 = T_m + \frac{1}{k_{\text{on}}}}$$

$$T_0 = \frac{1}{k_{\text{on}}}\frac{L}{S(0)} + \frac{1}{k_{\text{off}}}\frac{L-S(0)}{S(0)}$$

Substitute into T_m:

$$T_m = \frac{1}{k_{\text{off}}}\frac{L-S(0)}{S(0)} + \frac{1}{k_{\text{on}}}\left(\frac{L}{S(0)} - 1\right)$$

$$T_m = \frac{L-S(0)}{S(0)}\left(\frac{1}{k_{\text{off}}} + \frac{1}{k_{\text{on}}}\right)$$

Factor out $\frac{1}{k_{\text{on}}k_{\text{off}}}$:

$$\boxed{T_m = \frac{(k_{\text{on}}+k_{\text{off}})(L-S(0))}{k_{\text{on}}k_{\text{off}}S(0)}} \tag{4.25}$$

Any other dynamic properties of the system can be evaluated in a similar way.

Chapter 5

Results and Discussion

Effect of Target Position on Search Time:

- The search time for a target on a DNA chain varies depending on its position.
- **Symmetry Effect:** When the target is in the middle of the DNA chain (e.g., for $L = 11$), the search is fastest due to symmetry. This is evident in the red curve/symbols in Figure 5.1.

DNA Length Dependence:

- As the DNA length increases (e.g., $L = 101$ and $L = 1001$), the effect of target position on search time becomes less pronounced.
- For longer DNA chains, the search time plateaus, indicating minimal variation unless the target is near the ends.

Biological Implications:

- For realistic DNA lengths ($L = 10^6 - 10^9$), the search time is effectively independent of the target position, provided the target is not at the very ends.
- This suggests that **target position is not a critical factor** in protein search dynamics for most biological systems, simplifying the understanding of how proteins locate their targets.

Methodological Insight:

- The theoretical model (solid curves) aligns well with Monte Carlo simulations (symbols), validating the analytical approach.

- The parameters used ($k_{\text{off}} = 10^5\,\text{s}^{-1}$, $u = k_{\text{on}} = 10^5\,\text{s}^{-1}$) are relevant to lac repressor proteins, making the findings biologically significant.

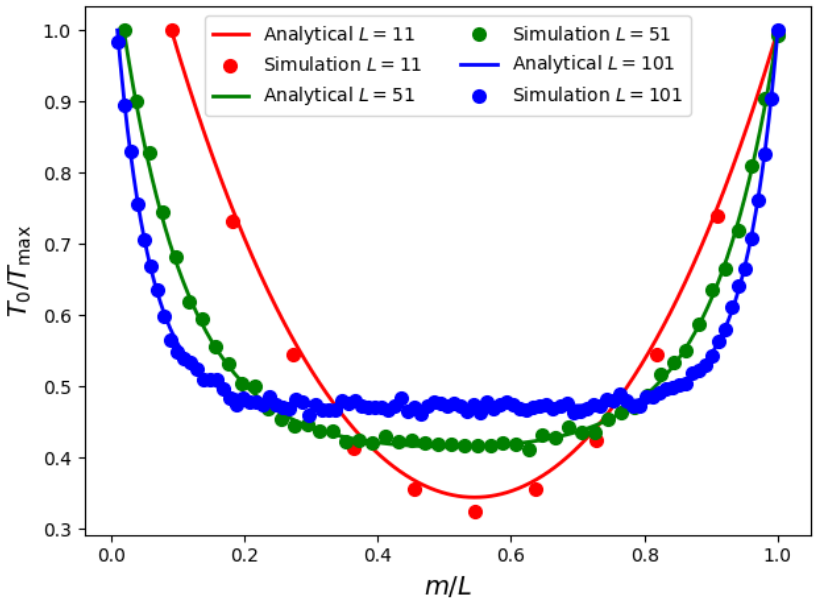


Figure 5.1: *Relative search time as a function of the target position on the DNA chain. Solid curves are analytical results, while symbols are from Monte Carlo simulations. The red curve and red symbols correspond to* $L = 11$*, the green curve and green symbols are for* $L = 51$*, and the blue curve and blue symbols describe* $L = 101$*. The transition rates are* $k_{off} = 10^3 s^{-1}$ *and* $u = k_{on} = 10^5 s^{-1}$*.*

Monte Carlo Simulation Approach

The Monte Carlo simulations were designed to stochastically model the protein search process on DNA and validate the analytical results. The key aspects of the implementation are as follows:

Simulation Steps

- **Initialization**: The protein starts at a random position on the DNA chain (including the bulk solution, represented as position 0). Each simulation starts with the same random settings so the results can be repeated and verified.

- **Dynamics**: The protein undergoes three primary processes:
  1. *Binding*: From the bulk (position 0), the protein binds to a random DNA site with rate k_{on}/L.
  2. *Sliding*: When bound, the protein moves left/right with rate u or dissociates with rate k_{off}. Boundary conditions (positions 1 and L) restrict sliding beyond the chain ends.
  3. *Termination*: The search concludes when the protein reaches the target position m.
- **Time Tracking**: The simulation uses exponential waiting times ($\Delta t = -\ln(\text{rand})/\text{total rate}$) to model stochastic transitions, where `rand` is a uniform random number in [0,1].

Validation and Comparison

- The simulation outputs (symbols in Figure 5.1) align closely with the analytical curves, confirming the theoretical framework's accuracy. Minor deviations arise from finite sampling but diminish with increased trials.
- The modular design allows easy parameter adjustments (e.g., L, k_{off}) to test different biological scenarios while maintaining computational efficiency.

Chapter 6

Future Work

Up until now, the protein search problem is modeled using a 1D discrete-state stochastic model with the DNA as a linear chain. The dynamics involve a protein that can slide (hop left/right) along the DNA at rate u and dissociate into the bulk (cytoplasm) with rate k_{off}, following the backward master equation formalism. The cytoplasmic return to DNA was treated as an immediate, uniform rebinding.

For the new project, we consider a two-dimensional $L \times L$ lattice representing the cell environment:

- The DNA occupies a central horizontal line: all sites $(i, L/2), i = 0, 1, \cdots, L-1$.
- All other sites represent the cytoplasm.
- The protein can move in 1D along DNA, dissociate into the 2D cytoplasm, diffuse randomly in the cytoplasm, and eventually rebind to the DNA at a neighboring site.

In the **2D lattice mode**l:

- When the protein dissociates from the DNA, it enters the cytoplasm at a site adjacent to the DNA.
- In the cytoplasm, the protein performs a random walk (2D diffusion) with hopping rate D_{2D} to neighboring sites.
- If a protein in the cytoplasm reaches a site adjacent to the DNA, it can bind to that DNA site with rate k_{on}.

Let $F_{i,j}(t)$ denote the first-passage probability starting at the cytoplasmic site (i, j), and $F_n(t)$ as the first-passage probability for DNA sites.

New Master Equations:

1. **DNA Sites** $(i, L/2)$
   The equations for DNA sites remain as in the 1D case, except dissociation now passes the protein to an adjacent cytoplasmic site.

$$\begin{aligned}\frac{dF_{i,L/2}(t)}{dt} =& u\left[F_{i+1,L/2}(t) + F_{i-1,L/2}(t)\right] \quad + k_{\text{off}}\left[\frac{1}{N_{\text{cyt}}}\sum_{\substack{(m,n)\in\text{Cyt}\\ \text{Nbr}(i,L/2)}} F_{m,n}(t)\right] \\ & - (2u + k_{\text{off}})F_{i,L/2}(t)\end{aligned}$$

   - Nearest neighbors in the cytoplasm to $(i, L/2)$ constitute the set $\text{CytNbr}(i, L/2)$, typically up to 2 per DNA site (above and below).
   - N_{cyt} is the number of such cytoplasmic neighbors.

2. **Cytoplasmic Sites** $((i, j), j \neq L/2)$
   For any cytoplasmic site, the master equation is defined as follows:

$$\begin{aligned}\frac{dF_{i,j}(t)}{dt} =& D_{2D}\sum_{(m,n)\in\text{Nbr}(i,j)} F_{m,n}(t) \quad + k_{\text{on}}\left[\frac{1}{N_{\text{DNA}}}\sum_{\substack{(p,q)\in\text{DNA}\\ \text{Nbr}(i,j)}} F_{p,q}(t)\right] \\ & - \left[D_{2D} + k_{\text{on}}\right] F_{i,j}(t)\end{aligned}$$

   - D_{2D} is the cytoplasmic diffusion rate.
   - $\text{Nbr}(i, j)$: Set of up to 4 lattice-neighbor sites.
   - n_{nbr}: Number of allowed neighbors for (i, j) (usually 4, but 3 or 2 at edges/corners).
   - $\text{DNANbr}(i, j)$: Set of DNA sites among the neighbors of (i, j), possibly 0, 1, or 2.
   - N_{dna}: Number of adjacent DNA sites to (i, j).

3. **Target Site**
   For the target site $(i^*, L/2)$:

$$F_{i^*,L/2}(t) = 1$$

The figure below illustrates the newly proposed 2D lattice model:

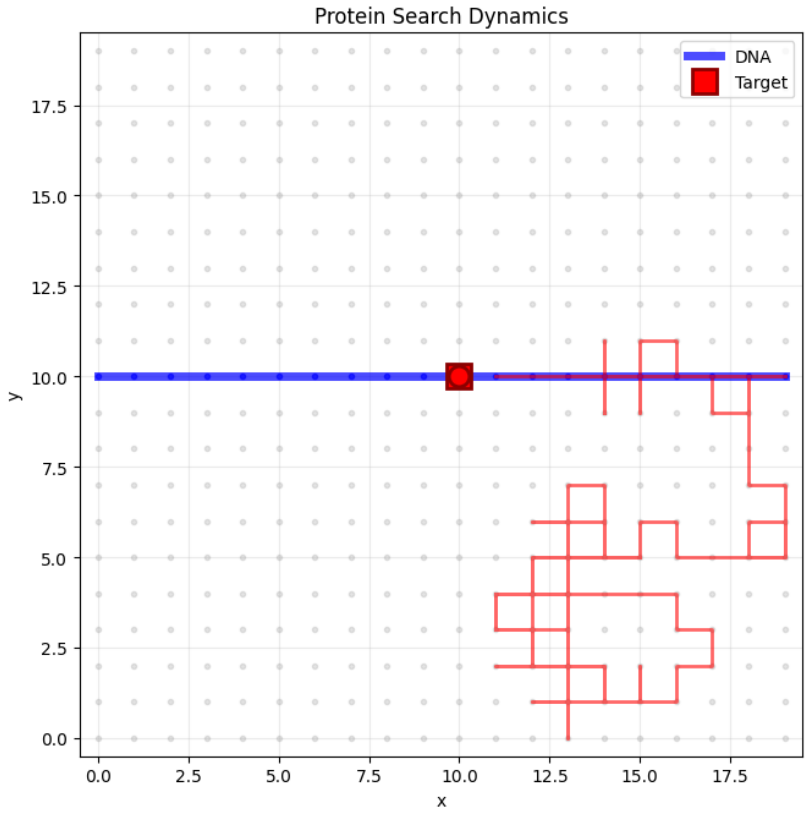


Figure 6.1: *Protein search dynamics showing: (1) 1D sliding along DNA (rate u), (2) dissociation into cytoplasm (rate k_{off}), (3) 2D random walk (rate D_{2D}), and (4) rebinding to DNA (rate k_{on}).*

Chapter 7

Literature Review

1. **Protein search for multiple targets on DNA**
   This study explores how multiple target sites influence search dynamics using a discrete-state stochastic model. While multiple targets generally speed up the search compared to a single target, the acceleration isn't always proportional to their number sometimes, counterintuitively, it slows the process. Key factors like target spacing, protein scanning behavior, and DNA length determine these outcomes. The findings have been supported by Monte Carlo simulations and experimental comparisons.

2. **Discrete-state stochastic kinetic models for target DNA search by proteins: Theory and experimental applications**
   Transcription factors and DNA-modifying enzymes locate their targets through random searches along DNA. Discrete-state stochastic kinetic models have been developed to analyze how search efficiency depends on molecular properties (e.g., protein-DNA interactions) and external factors like crowding. These models bridge microscopic processes (e.g., binding, sliding) with macroscopic observations, while also aiding in interpreting experimental data from techniques such as stopped-flow assays, single-molecule imaging, and NMR.

3. **Active motion can be beneficial for target search with resetting in a thermal environment**
   This study investigates how combining two types of environmental noise - thermal (passive) and telegraphic (active) - affects search efficiency. The research focuses on a self-propelled "run-and-tumble" particle in a thermal bath that undergoes position resets, either maintaining or reversing its propulsion direction. Surprisingly, thermal noise actually

enhances search efficiency by reducing the mean first-passage time. Additionally, velocity reversal during resets can further decrease search time. These findings reveal complex interactions between active and passive noise components in target-search processes.

4. **Facilitated search of proteins on DNA: correlations are important**
   Protein target search on DNA is remarkably efficient despite low protein concentrations and vast genomic complexity. While current theory explains this through a combination of $3D$ diffusion and $1D$ sliding along DNA, some predictions conflict with single-molecule observations. An alternative approach emphasizes the role of correlated motions caused by non-specific protein-DNA interactions. Through Monte Carlo simulations, this study demonstrates that search acceleration occurs only at intermediate non-specific binding strengths and protein concentrations, highlighting the importance of these transient interactions.

5. **Sequence heterogeneity accelerates protein search for targets on DNA**
   This theoretical study examines how DNA sequence properties including symmetry, heterogeneity, and local chemical composition influence the efficiency of protein target search. Using a discrete-state stochastic model with first-passage analysis, the work challenges conventional assumptions by demonstrating that sequence heterogeneity accelerates protein search, while local chemical composition near target sites can further modulate search dynamics.

6. **Theoretical insights into the full description of DNA target search by subdiffusing proteins**
   DNA-binding proteins (DBPs) navigate the crowded cellular environment to locate their target sites on DNA, a process critical for initiating key biological functions. While traditional studies focus on average search times, this work highlights that the most probable search paths can differ dramatically in timescale (by orders of magnitude) from the mean. Cellular conditions like crowding and viscoelasticity often force proteins into anomalous (slowed) diffusion, a factor frequently ignored in theoretical models. Here, the authors derive analytical formulas to predict search-time distributions for subdiffusing proteins, accounting for real-world cellular complexity.

References

[1] Veksler, A., & Kolomeisky, A. B. (2013). Speed-selectivity paradox in the protein search for targets on DNA: Is it real or not? *The Journal of Physical Chemistry B*, **117**(42), 12695–12701.
https://pubs.acs.org/doi/10.1021/jp311466f

[2] Shvets, A. A., Kochugaeva, M. P., & Kolomeisky, A. B. (2018). Mechanisms of protein search for targets on DNA: Theoretical insights. *Molecules*, **23**(9), 2106.
https://doi.org/10.3390/molecules23092106

[3] Kolomeisky, A. B., & Veksler, A. (2012). How to accelerate protein search on DNA: Location and dissociation. *The Journal of Chemical Physics*, **136**(12).
https://doi.org/10.1063/1.3697763

[4] Mondal, K., & Chaudhury, S. (2020). Effect of DNA conformation on the protein search for targets on DNA: A theoretical perspective. *The Journal of Physical Chemistry B*, **124**(17), 3518–3526.
https://doi.org/10.1021/acs.jpcb.0c01996