Theory of DNA polymerase

Misha Klein

1 Introduction

Inside every living cell there are many different protein machines performing different tasks that are necessary for the cell's survival. Protein machines or biomolecular machines are found in nearly every part of the cell. Ranging from those controlling what molecules/chemicals penetrate the cell through its membrane to machines that transport cargo all the way to machines playing a key role in the cell's nucleus. In this report we focus on the protein machine DNA polymerase (DNAP) which conducts DNA replication. What makes DNAP so special is its extremely high fidelity and production rate. A single DNA polymerase makes an error every 10^7-10^9 base pairs only, even though it reads/synthesizes about 1000 nucleotides every second.
In this report a model will be set up that tries to explain the high fidelity and other features of DNAP dynamics that are seen in typical experiments. This section will provide the reader with the Biology needed to understand the theory. In addition, an introduction into typical experiments performed on DNAP will be given. The used model is based on the continuous time random walk theory of which an introduction will be given in the next section. After that the model that applies to the polymerase will be introduced and the most important mathematical results will be derived. This report shows two major results. One concerns the dynamics of DNAP during replication, excluding any errors. The mathematical model will be compared to simulated data from a Monte Carlo simulation using the Gillespie algorithm. The comparison is done through the method of maximum likelihood fitting. Both the fitting method and simulation algorithm will be discussed. Finally, the performance of the imperfect DNAP will be assessed based on the derived mathematical model.
Main motivation for this research is to reach more understanding of DNAP, and similar protein machines, on a fundamental level. Another motivation is from a medical point of view, since accurate DNA replication is vital for us humans and incorporated mutations are said to be responsible for many types of deceases such as cancer.

1.1 Biology of DNA polymerase

DNA polymerase is a complex formed from multiple proteins all working together to perform a specific task: It copies, replicates, DNA. DNA consists of two strands, chains, of nucleotides (DNA = deoxyribonucleic acid) that are wrapped around each other to form a helical structure, as discovered by Watson and Crick in 1953. Every link in the chain may consist out of any of the four nucleotides: (A)denine,(T)hymine, (C)ytosine or (G)uanine. The two strands of DNA are complementary to each other, since an A always will bind to a T and a C to a G. This makes that both strands may serve as a template for a new strand, this is exploited during replication. To start replication, the DNA must be partially ripped open first. Specific proteins are able to break the hydrogen bonds keeping the two strands together. A short region (compared to the entire genome) where only single-stranded DNA is present is formed. Next, a special protein(machine) called primase is capable of synthesizing a short ,about 10bp (base pairs) long complementary strand of RNA. This is needed to get DNAP started. The polymerase is capable of catalyzing the synthesis of DNA. Every newly incoming nucleotide is bound to a chain of three phosphor groups. DNAP breaks one of the bonds in this chain, releasing enough energy to catalyze DNA synthesis. The nucleotide is bonded to a single phosphor group after the reaction making the process irreversible. The workings of a polymerase are described by two active sites within the protein complex. One active site where a new nucleotide can be added to the product chain. When the DNA is bound to this active site the DNAP will be said to be in its "polymerase state". Another active site, located in another domain of the complex, plays a special role in this report. When the end of the new strand is bound to this active site, the double stranded DNA is partly ripped open to enable DNAP of cleaving off (excising) the nucleotides it just added. When using this active site, the DNAP is said to be in its "exo-state". Why would a DNAP ever want to remove nucleotides? To correct for a possible error! Whenever an error is placed, an incorrect Watson-Crick base pair is formed (imperfection in the helix). Although this is energetically unfavorable, the extremely high fidelity of DNAP cannot be explained based on energetics alone. A DNAP is capable of proofreading its product while it is

producing it. Just as the name suggests, the DNAP goes over the DNA it just synthesized to see if everything went correct. If it finds an error, it is capable of removing it by switching to its exo-state.
Besides having two conformational states, the model will consist of more states. Replication can start at multiple sites at the same time, such that multiple polymerases can be active at the same time. DNAP is able to bind and unbind and reside in solution. It may occur that the polymerase does not bind correctly to the DNA (or the DNA to its active site). In this case, no new nucleotides will be added nor will their be any excised. The polymerase has paused so to speak. Not only is the paused polymerase inactive, it forms a "roadblock" for any other polymerase. Single molecule experiments are used to understand the movement of DNAP during the replication process.

1.2 Introduction to optical tweezer experiments

Optical tweezers are devices capable of trapping a micrometer sized object by the use of light. This is achieved by aiming a laser at a dielectric bead. When the laser light encounters the bead it will either get reflected or transmitted. Since the bead is dielectric, any transmitted light will get refracted causing light rays to propagate along a different direction upon exiting the bead. Light carriers momentum. A change in the direction of beam propagation causes a change in momentum of the beam. A change in momentum corresponds to an applied force on the light by Newton's second law. Newton's third law now tells us that an opposite force will be exerted on the bead. This can be used to trap the bead at a specific position by making use of the properties of the laser light. A focussed laser beam has a Gaussian intensity profile in the plane perpendicular to its propagation. This means that light intensity is at its maximum on the axis and decreases radially. The force applied on the bead is proportional to the intensity of the light. This will cause the bead to move towards the peak of intensity, the beam waist (the center of a Gaussian beam). In practice, the bead will be trapped slightly downstream from the beam waist due to reflected light transferring momentum to the bead along the direction of propagation of the laser beam.
To study protein machines such as DNA polymerase, two optical traps are used. The two beads are used to clamp DNA suspended between them. The DNA then consists of a single strand attached to the beads on both sides and an incomplete complementary strand attached to a single bead. An active polymerase working on the suspended DNA will add or remove nucleotides from the incomplete strand. When a new nucleotide is added, that strand increases in length, thereby exerting a tension force on one of the beads. The slight displacement of the bead can be tracked. From this, the position of the DNAP as a function of time can be inferred (see figures (25) and (19)). A plot of the position of DNAP versus time will be referred to as a trace. More on these traces will be discussed in section 5.

2 Continuous time random walk theory

This section provides an introduction to the necessary mathematical theory needed to understand the models used in subsequent sections. All of the models will involve diagrams of nodes and arrows indicating connections between them. A convenient way of approaching these problems will be to view them as little "board games". Walking over the board is done by hoping from one node to another, one at the time and only in a direction along an arrow. This section will mainly focus on the "rules of the game".

2.1 First passage problems

When one considers the movement of a body on an interval x∈[a,b], then a simple question one may ask is: *"What is the time at which the particle passes the boundary at a or b for the first time?"* This time is called the first passage time. One may also ask: *"What is the probability that the first passage time at boundary a is $t=t_a$?"* This will be referred to as the first passage probability. Throughout this report we will be working with such first passage probabilities. More precisely, we let $\Psi(t)\mathrm{d}t$ denote the probability that the first passage time lies within [t,t+dt]. Hence, $\Psi(t)$ is the "first passage probability density". We will be working with discretized space, whilst time will still be treated as a continuous variable. As an example, let us consider the following "board game":

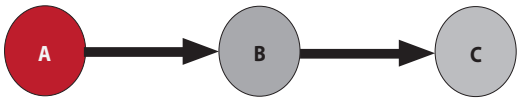


Figure 1: *Example 1*

Say, that we start at node A. As mentioned, in our next move we can only reach a node that neighbors A and for which there is an arrow pointing from A to the designated node. In this case, we have little choice but walking to node B. We may ask ourselves what the first passage probability (density) at node B is at a time t. We denote this by $\phi_{A-B}(t)$. Note that we make the assumption that no time is elapsed between leaving node A and arrival at one of the neighboring nodes (here B). The time we are recording, is strictly the time until a transition takes place. The transitions themselves are instantaneous. In other words, it is as if we are engaged in a game of "speed chess " in which we record the times it takes us to decide which move we are going to make. Next, we may ask ourselves what the probability is that it will take us a time t to arrive at node C. Since node A is not directly connected to node C, all possible paths that bring us to node C must have first brought us to node B at some earlier time $\tau < t$. Hence:

$$\Psi_{A-C} = \int_0^\infty \phi_{A-B}(\tau)\phi_{B-C}(t-\tau)\mathrm{d}\tau \tag{1}$$

will equal the total first passage probability for a player starting out at node A and wanting to end at node C. Throughout this report, a ϕ will denote a first passage when only considering paths that include taking a single step on the board, whilst a Ψ will be used for composite paths. The integral above reflects that we must sum over all possible ways of ending up at C via node A ("more possibilities,more likely"). In this case, that entails summing over all possible times at which we arrived at the intermediate node B. Note, that the above integral equals the convolution of $\phi_{A-B}(t)$ with $\phi_{B-C}(t)$. This is one of the main reasons for us not to work with the temporal probability density, but rather with its Laplace transform:

$$\Psi(s) = \mathcal{L}\{\Psi(t)\} = \int_0^\infty \Psi(t)e^{-st}\mathrm{d}t \tag{2}$$

Recall that the Laplace transform of a convolution of two functions (signals) in time equals the product of the individual Laplace transforms in s-space.

$$\begin{aligned}
\Psi_{A-C}(s) &= \mathcal{L}\{\int_0^\infty \phi_{A-B}(\tau)\phi_{B-C}(t-\tau)\mathrm{d}\tau\} \\
&= \int_0^\infty \int_0^\infty \phi_{A-B}(\tau)\phi_{B-C}(t-\tau)\mathrm{d}\tau e^{-st}\mathrm{d}t \\
&= \int_0^\infty \int_0^\infty \phi_{A-B}(\tau)\phi_{B-C}(t-\tau)e^{-st}\mathrm{d}t\mathrm{d}\tau \\
&\equiv \int_0^\infty \int_0^\infty \phi_{A-B}(\tau)\phi_{B-C}(u)e^{-s(u+\tau)}\mathrm{d}u\mathrm{d}\tau \\
&= \int_0^\infty \phi_{A-B}(\tau)e^{-s\tau}\mathrm{d}\tau \int_0^\infty \phi_{B-C}(u)e^{-su}\mathrm{d}u \\
&= \phi_{A-B}(s)\phi_{B-C}(s)
\end{aligned} \tag{3}$$

In the fourth line the variable substitution u=t-τ is made. The Laplace transform is a linear operator. This property will be put into practice in the next example:

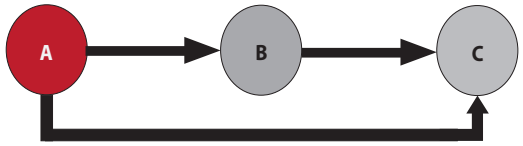


Figure 2: *Example 2*

If for this situation we would want to know Ψ_{A-C}, we see that there are two distinct types of paths that bring us from A to C. We can walk from A to C directly (ϕ_{A-C}), but we must also include the possibility that we walked there via node B. Since the Laplace transform is linear, summing over different distinct paths simply becomes summing over the distinct transforms:

$$\Psi_{A-C}(s) = \phi_{A-B}(s)\phi_{B-C}(s) + \phi_{A-C}(s) \tag{4}$$

The simple form of the composite first passage probability is not the only reason for working with Laplace transforms. Let us look at the derivative of $\Psi(s)$:

$$\begin{aligned}
\Big(\frac{\mathrm{d}\Psi}{\mathrm{d}s}\Big)_{s=0} &= \Big(\frac{\mathrm{d}}{\mathrm{d}s}\int_0^\infty \Psi(t)e^{-st}\mathrm{d}t\Big)_{s=0} \\
&= \Big(\int_0^\infty \frac{\mathrm{d}}{\mathrm{d}s}(\Psi(t)e^{-st})\mathrm{d}t\Big)_{s=0} \\
&= \int_0^\infty -t\Psi(t)e^0\mathrm{d}t = \int_0^\infty -t\Psi(t)\mathrm{d}t \\
&\equiv -\langle t\rangle
\end{aligned} \tag{5}$$

We see that $\Psi(s)$ gives us direct access to the average first passage time $\langle t\rangle$. Higher order moments of the first passage time are found by taking higher order derivates of Ψ. The Laplace transform of the first passage probability is said to be a moment generating function.

$$\langle t^n\rangle = (-1)^n\Big(\frac{\mathrm{d}^n\Psi}{\mathrm{d}s^n}\Big)_{s=0} \tag{6}$$

The 0^{th} order moment has a special interpretation:

$$P \equiv \Psi(0) = \int_0^\infty \Psi(t)\mathrm{d}t \tag{7}$$

It equals the total probability of walking a specified type of path. In the example above, $\Psi_{A-C}(0)$ tells us what the probability is that we will ever reach node C when starting at node A. Note, that in principle $\Psi(t)$ should be a normalized probability density, for which P=1. Later in this report however, we will encounter Ψ's that do not consider all possible paths one could walk on the board. In those cases P might differ from 1. This will become clear later on. Often, the arguments of the distributions will be omitted. From the context it should become clear whether we are dealing with a Laplace transform or a function in the time domain.

Before moving forward to the model for DNAP, there are two more "rules of the game" that will come in handy. First, let us look at the following example:

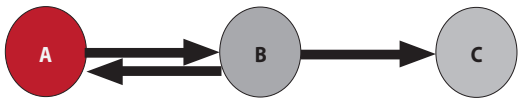


Figure 3: *Example 3*

Just as in the previous examples we search for the probability density that when starting at A we arrive at node C within a given time interval. In this example, we will directly work with the Laplace transforms. The board reveals that we cannot reach node C within a single step. We must walk to node B first. However, there are now many other ways in which we can reach node C for the first time. After walking to node B, we can decide to walk back to point A, then back to B and finally to C. As a matter of fact, we can decide to walk back and forth between A and B as often as we want as long as we end with a step to B and one to C. Using the convolution property and the linearity of the Laplace transform we find:

$$\begin{aligned}\Psi_{A-C} &= [1 + (\phi_{A-B}\phi_{B-A}) + (\phi_{A-B}\phi_{B-A})^2 + (\phi_{A-B}\phi_{B-A})^3 + ...]\phi_{A-B}\phi_{B-C} \\ &= \sum_{n=0}^{\infty}(\phi_{A-B}\phi_{B-A})^n\phi_{A-B}\phi_{B-C} \\ &= \frac{\phi_{A-B}\phi_{B-C}}{1-(\phi_{A-B}\phi_{B-A})}\end{aligned} \tag{8}$$

The last line follows from the fact that we have a geometric series. Another handy rule becomes clear when considering a board that contains the following:

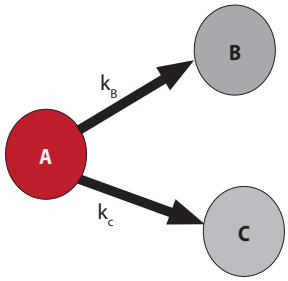


Figure 4: *Example 4*

If we stand at node A, there are two possible moves we can make. Either we step to node B or to node C. Consequently there are two types of first passage times. The k's in the figure denote the different rates at which we leave node A via the different paths. The rates k have units of s^{-1} (1/time). Their meaning becomes more apparent if we calculate the probability densities $\phi_{A-B}(t)$ and $\phi_{A-C}(t)$. If we let $P_A(t)$ denote the probability of being at node A at time t:

$$\partial_t P_A = -(k_B + k_C)P_A \tag{9}$$

$$k_B P_A \mathrm{d}t = \phi_B \mathrm{d}t \tag{10}$$

$$k_C P_A \mathrm{d}t = \phi_C \mathrm{d}t \tag{11}$$

Imagine that there are multiple players involved in this board game. The first of these equations reflects the fact that the number of players's pieces that we will find at node A at any time can change by players deciding to walk from A to B or C. The correct physical interpretation is that it tells us how the total number of particles (or polymerase) can change through outflow towards nodes B and C. Therefore, the probability of finding a particle gets altered in the same manner (by dividing by the total number of particles). The other two equations can be interpreted as follows:

- **(Left hand side)Probability of leaving A within a time dt**: For this we multiply the rate at which we leave node A, k_B or k_C (depending on the desired path), by the time interval dt and the probability of finding a particle at node A to begin with $P_A \equiv P_A(t)$.
- **(Right hand side)Prob. of arriving at end point within a time dt**: By definition, this equals $\phi_B(t)\mathrm{d}t$ or $\phi_C(t)\mathrm{d}t$.

Also, we make use of the assumption that all transitions are nearly instantaneous. Now we can solve for the $\phi(t)$'s. First we use equation (9), to find $P_A(t)$:

$$P_A(t) = e^{-(k_B+k_C)t} \tag{12}$$

Using this in equations (10) and (11) we find that:

$$\phi_B(t) = k_B e^{-(k_B+k_C)t} \tag{13}$$

$$\phi_C(t) = k_C e^{-(k_B+k_C)t} \tag{14}$$

The ϕ's describe Poisson distributions. We now see that after a time $(k_B + k_C)^{-1}$ the probability of leaving node A has decreased significantly (to a factor of 1/e of its original value). Let us calculate the generating functions, by first considering the Laplace transform of an exponential:

$$\begin{aligned} \mathcal{L}\{e^{-at}\} &= \int_0^\infty e^{-at} e^{-st} \mathrm{d}t = \int_0^\infty e^{-(s+a)t} \mathrm{d}t \\ &= \frac{-1}{s+a}[e^{-(s+a)t}]_0^\infty = \frac{1}{s+a} \ \forall a > 0 \end{aligned} \tag{15}$$

Combining equations (13),(14) and (15) we find that:

$$\phi_B(s) = \frac{k_B}{s + k_B + k_C} \tag{16}$$

$$\phi_C(s) = \frac{k_C}{s + k_B + k_C} \tag{17}$$

When looking at $\phi(0)$'s we see that the k's have another interpretation:

$$\phi_B(0) = \frac{k_B}{k_B + k_C}$$

This tells us what the probability that our next move will take us to node B is. Hence, if both rates are equal there is a 50% chance that we make either of our possible moves.

2.2 Inverting Laplace Transforms

Typically, as will become clear in subsequent sections, we will be facing a board game and first attempt to find the generating function of the desired first passage probability. This function, $\Psi(s)$, nearly always gives us all information we need. However, there are exceptions for which we will need the "physical" probability density. Inverting a Laplace transform is often quite difficult. Fortunately, we will often be dealing with ratios of polynomials in s, as in the example above.

$$\Psi(s) = \frac{Q(s)}{P(s)} \tag{18}$$

Let $P(s)$ be an n^{th} order polynomial in s. One can in principle factorize P as follows:

$$\Psi(s) = \frac{Q(s)}{\prod_{i=1}^{n}(s - s_i)} \tag{19}$$

where the s_i denote the roots of $P(s)$. There are two methods of finding the inverse transform of such functions. One could use partial fraction decomposition to write $\Psi(s)$ as the sum of fractions:

$$\Psi(s) = \frac{Q(s)}{\prod_{i=1}^{n}(s - s_i)} = \sum_{i=1}^{n} \frac{c_i}{(s - s_i)} \tag{20}$$

Using the linearity of the $\mathcal{L}$-transform we see that inverting this function will lead to a sum of poisson distributions with additional pre factors c_i. When the $Q(s)$ is not a constant function, it becomes more difficult to find the correct fractions which form the decomposition. In this report, a more direct approach will be adopted using some complex analysis. By definition the inverse Laplace transform is given by the so called Bromwich integral:

$$\mathcal{L}^{-1}\{\Psi\} = \frac{1}{2\pi i} \lim_{R\to\infty} \int_{a-iR}^{a+iR} e^{st}\Psi(s)\mathrm{d}s \tag{21}$$

This integral can be evaluated by using a complex contour integral over a path containing the Bromwich line integral, horizontal line segments satrting from the end points of this vertical line op until the imaginary axis and a semicircle in the left half of the complex plane to close the contour (see [7, fig 7.6 on p.332]) . One can prove that the two integrals over the line segments as well as the integral over the semicircle all tend to zero as R tends towards infinity. Because we now effectively have rewritten the inverse Laplace transform as a closed contour integral, we may invoke the Cauchy Residue Theorem. In short, we find the $\mathcal{L}^{-1}$'s directly by finding the residues of the generating function:

$$\mathcal{L}^{-1}\{\Psi\} = \sum_{i} Res(e^{st}\Psi(s), s_i) \tag{22}$$

The roots of the denominator, P, will also equal the poles of Ψ. Evaluation of the residue of a (complex valued) function $f(z)$ for the case that the singularity is an n^{th} order pole at z_0 can be done with the following equation:

$$Res(f(z), z_0) = \frac{1}{(1-n)!} \lim_{z\to z_0} \frac{\mathrm{d}^{n-1}}{\mathrm{d}z^{n-1}}((z - z_0)^n f(z)) \tag{23}$$

3 General model for DNAP

In this section the general model used to describe the workings of DNAP will be introduced. This section will also serve as a demonstration of how the ideas introduced in the previous section can be put into practice.

3.1 The self-consistency equation

The workings of DNAP are described by the following diagram:

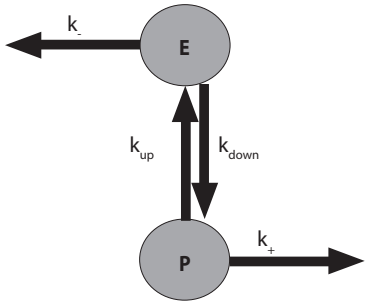


Figure 5: *Basic unit of the general model showing the different conformational states and different rates.*

As described previously, the polymerase knows two basic conformational states. In the polymerase state (P) the active site is positioned such that it can add an additional nucleotide to the replica DNA strand at a rate k_+. In the exo-state (E), the active site is positioned such that the newly synthesized DNA is ripped open and the polymerase can start cleaving off (excising) nucleotides one by one at a rate k_-. The polymerase switches between its two states at rates denoted by $k_\uparrow$ and $k_\downarrow$. First passage probabilities belonging to single steps in this diagram will be denoted by ϕ_+, ϕ_-, $\phi_\uparrow$ and $\phi_\downarrow$ respectively. The connected diagram displayed above forms just a single basic unit of the entire model. The following figure displays the full board game the T7 DNAP will be engaged in:

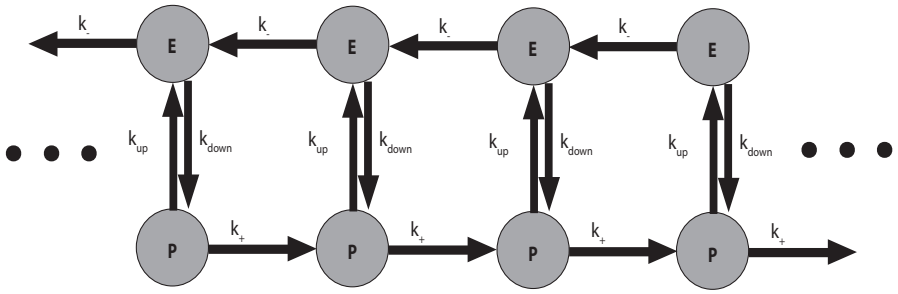


Figure 6: *Entire board game. The black dots indicate that the lattice extends infinitely.*

Since typical genome lengths are much longer than the typical amount of bp that are inside the DNAP's machinery, we might as well imagine the board to be infinitely long as in the figure.
We are looking for the generating function, Ψ, that describes the first passage of a polymerase at node N+1 if it starts at node N (on pol.-line). Here, we are looking for the Ψ that takes into account every

possible path that ends up at node N+1, including those that cut back (to N-1,N-2, etc.) first. The infinite lattice possesses translational symmetry. If you walk forward (or backward) by one node, your environment is exactly the same. Therefore, if making a single step starting on the node at N results in a Ψ, so does making a single step forward from N$\pm$1 or any other node on the polymerase line. This fact will enable us to find a self-consistency equation that Ψ must satisfy. To do so, let us redraw the board making use of this translational symmetry:

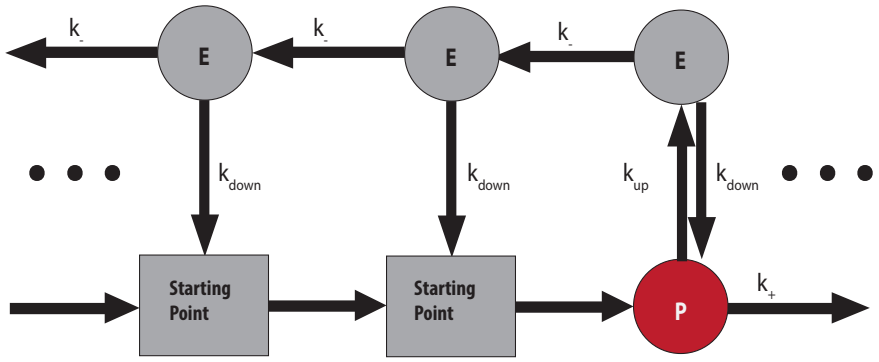


Figure 7: *Using the translational symmetry. Starting point indicated in red. Standing on a square is equivalent to standing on the starting point.*

The squares now indicate a special kind of node. If we stand on a square, our environment is exactly the same as it was for our very first move. However, we do still keep track of the actual position we are at on the polymerase line. The beauty of redrawing the board as above is that we can treat a square just the same as a normal node with Ψ denoting the first passage at its right neighbor node. By construction, this will take care of all the possible paths we might walk "inside the square". We can now construct the generating function by considering the different types of paths we may take:

- **Walk straight forward without cutting back:**

$$\sum_{n=0}^{\infty}(\phi_\uparrow\phi_\downarrow)^n\phi_+ = \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow}$$

- **Excise nulceotides/ Walk backwards before walking forwards:** If we excised (at least) a single base pair, then we walked the following path:
  - **Take a step upwards, after having the option of walking back and forth between P and E:**

$$\sum_{n=0}^{\infty}(\phi_\uparrow\phi_\downarrow)^n\phi_\uparrow$$

  - **Walk backwards for one step:**

$$\phi_-$$

  - **Enter the P state at site N-1:**

$$\phi_\downarrow$$

  - **Make ourselves back to our original position**

$$\Psi$$

– **Now there are again many different paths one can take that will finally result in the first passage at site N+1:**

$$\Psi$$

Multiplying the terms above will result in one of the terms contributing to Ψ. We get the total contribution of all paths that walk backwards before walking forwards by summing over the number of nucleotides excised during the first cut. The convenient way of redrawing the board reveals that including the Ψ's in the last two terms mentioned will take care of all paths that make multiple cuts. In total:

$$\sum_{m=1}^{\infty}\sum_{n=0}^{\infty}(\phi_\uparrow\phi_\downarrow)^n\phi_\uparrow\phi_-^m\phi_\downarrow\Psi^m\Psi$$

In conclusion, the self-consistent equation that Ψ must satisfy is:

$$\begin{aligned}
\Psi &= \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow} + \sum_{m=1}^{\infty}\sum_{n=0}^{\infty}(\phi_\uparrow\phi_\downarrow)^n\phi_\uparrow\phi_-^m\phi_\downarrow\Psi^m\Psi \\
&= \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow} + \frac{\phi_\uparrow\phi_\downarrow}{1-\phi_\uparrow\phi_\downarrow}\sum_{m=1}^{\infty}\phi_-^m\Psi^{m+1} \\
&= \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow} + \frac{\phi_\uparrow\phi_\downarrow}{1-\phi_\uparrow\phi_\downarrow}\Big[\sum_{m=0}^{\infty}\phi_-^m\Psi^{m+1} - \Psi\Big] \\
&= \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow} + \frac{\phi_\uparrow\phi_\downarrow}{1-\phi_\uparrow\phi_\downarrow}\Big[\frac{\Psi}{1-\phi_-\Psi} - \Psi\Big] \\
&= \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow} + \frac{\phi_\uparrow\phi_\downarrow}{1-\phi_\uparrow\phi_\downarrow}\frac{\phi_-\Psi^2}{1-\phi_-\Psi}
\end{aligned} \tag{24}$$

The result is a quadratic equation that determines Ψ. No explicit solution for $\Psi(s)$ will be given here, since it is not very instructive. However, finding the total probability $\Psi(0)$ does serve as an important check wether the applied reasoning is true. Any constructed Ψ should be capable of describing the physical system at hand. Before finding the probability, let us write the ϕ's in terms of the k's. This can be done by the same reasoning used in the previous section and applying it to figure (5):

$$\phi_+ = \frac{k_+}{s+k_++k_\uparrow} \tag{25}$$

$$\phi_- = \frac{k_-}{s+k_-+k_\downarrow} \tag{26}$$

$$\phi_\uparrow = \frac{k_+}{s+k_++k_\uparrow} \tag{27}$$

$$\phi_\downarrow = \frac{k_-}{s+k_-+k_\downarrow} \tag{28}$$

The solutions to the quadratic equation (24) after plugging in equations (25)-(28) and setting s equal to zero are as follows:

$$\Psi(0) = \frac{1}{2}\frac{2k_+k_- + k_+k_\downarrow + k_\uparrow k_- \pm |k_\uparrow k_- - k_+k_\downarrow|}{k_-k_+ + k_-k_\uparrow} \tag{29}$$

Naturally, there is only one actual Ψ. Since there are two solutions, one should be discarded based on physical grounds. We may interpret $k_\downarrow k_+$ as a measure for how fast we can add a new nucleotide when starting in the exo-state. Similarly, $k_\uparrow k_-$ is a measure for the effective backward rate. A physically correct functioning polymerase should eventually replicate the entire DNA. Therefore, the forward rate should exceed the backward rate.

$$k_+k_\downarrow > k_-k_\uparrow \tag{30}$$

In this limit Ψ should be normalized, since it now is certain that one will eventually always take a step forward on the board. Using this:

$$\Psi(0) = \frac{1}{2}\frac{2k_+k_- + k_+k_\downarrow + k_\uparrow k_- \pm (k_+k_\downarrow - k_-k_\uparrow)}{k_-k_+ + k_-k_\uparrow} \tag{31}$$

The two solutions are:

$$\Psi(0) = 1 \quad \textit{(Physical)} \tag{32}$$

$$\Psi(0) = \frac{k_+k_- + k_+k_\downarrow}{k_-k_+ + k_-k_\uparrow} \quad \textit{(Unphisical, discard)} \tag{33}$$

We see that when adapting the limit of equation (30) we must use the solution with the (-)-sign. Apparently, the used method of constructing Ψ results in a normalized probability density under physically relevant conditions.

3.2 The effect of forces

Before we are able to calculate the DNAP's velocity we must consider the effect of an applied force on the system. Without any forces applied, a sketch of a typical free energy landscape in which the polymerase moves looks as follows:

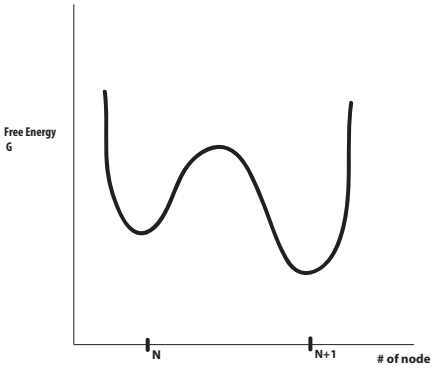


Figure 8: *Free energy landscape without forces*

A base further down the DNA is drawn at lower energy to indicate that the polymerase will eventually replicate the entire DNA. In other words, we are working in the limit of equation (30). When an external force is applied, the free energy gets altered:

$$\Delta G \to \Delta G - l_{BP}F \tag{34}$$

Here, $l_{BP} \approx 0.34$nm is the typical length of a single base pair, the distance in between nodes. Effectively, this is the distance over which the applied force does work. We define a positive force such that it will help our DNAP to move faster in the forward direction. Graphically, the change in free energy can be represented as follows:

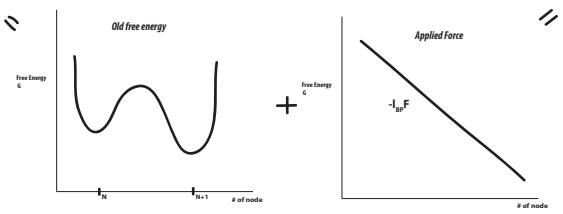


Figure 9: *Illustration indicating the change in free energy due to the external force.*

This results in a new free energy landscape that is sketched below:

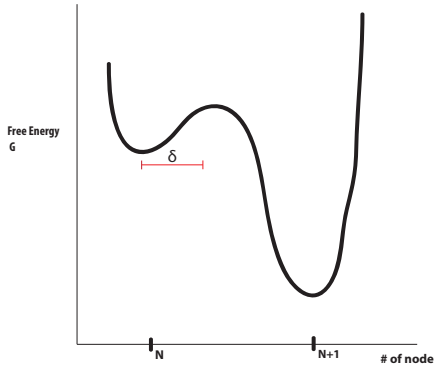


Figure 10: *Free energy landscape including an external force*

We see that the energy barrier for walking N→N+1 has decreased, hence the corresponding rate has increased. Conversely, the energy barrier for walking N→N-1(or N+1→N) has increased.

$$k_+(F) = k_+(0)e^{\delta F} \tag{35}$$

$$k_-(F) = k_-(0)e^{-(l_{BP}-\delta)F} \tag{36}$$

The distance δ is indicated in the figure above. It may be interpreted as the effective distance over which one needs to do work against the applied force if one wants to reach the node at N+1. In what follows, δ wil be set to $0.5l_{BP}$.

3.3 The Velocity

Generally speaking, the velocity of the protein machine is given by the following relation:

$$\langle V \rangle = [P_{pol}k_+ - P_{exo}k_-]l_{BP} \tag{37}$$

That is, we need the probability of finding the DNAP in any of its conformational states and multiply by the rates taking it from N → N±1. The most instructive way of obtaining $\langle V \rangle$ is by solving the steady state Master equations under the following assumptions:

- In steady-state, the probability of finding the DNAP in one of its states will not depend on the node we are looking at.

$$P_{pol}(N \pm 1) = P_{pol}(N) \equiv P_{pol} \forall N$$
$$P_{exo}(N \pm 1) = P_{exo}(N) \equiv P_{exo} \forall N$$

- The DNAP certainly will be found in one of these states:

$$P_{pol} + P_{exo} = 1$$

The Master equations describe how the population of DNAPs in a particular state can alter. In view of figure (5):

$$\frac{\partial P_{pol}}{\partial t} = -k_{\uparrow} P_{pol}(N,t) - k_{+} P_{pol}(N,t) + k_{\downarrow} P_{exo}(N,t) + k_{+} P_{pol}(N-1,t) \tag{38}$$

$$\frac{\partial P_{exo}}{\partial t} = -(k_{\downarrow} + k_{-}) P_{exo}(N,t) + k_{\uparrow} P_{pol}(N,t) + k_{-} P_{exo}(N+1,t) \tag{39}$$

The steady-state equations under the first assumption listed above:

$$0 = -k_{\uparrow} P_{pol} + k_{\downarrow} P_{exo} \tag{40}$$

$$0 = -k_{\downarrow} P_{exo} + k_{\uparrow} P_{pol} \tag{41}$$

Essentially, the two equations are reduced to a single equation. Using the second assumption we find:

$$k_{\uparrow} P_{pol} = k_{\downarrow} P_{exo} = k_{\downarrow}(1 - P_{pol}) \tag{42}$$

From which we find that:

$$P_{pol} = \frac{k_{\downarrow}}{k_{\uparrow} + k_{\downarrow}} \tag{43}$$

$$P_{exo} = \frac{k_{\uparrow}}{k_{\uparrow} + k_{\downarrow}} \tag{44}$$

Plugging this into equation (37) yields the velocity:

$$\langle V \rangle \equiv \frac{k_{\downarrow} k_{+} - k_{\uparrow} k_{-}}{k_{\uparrow} + k_{\downarrow}} \tag{45}$$

where we omit the length scale. In other words, we are interested in " the velocity per unit length" only. In order to describe a well functioning polymerase, the velocity should always be positive. For this to be true, it should hold that

$$k_{+} k_{\downarrow} > k_{-} k_{\uparrow}$$

This is precisely the same limit considered in equation (30) and which al ready was discussed to be the physically relevant limit. When read in reverse it merely states that if the polymerase moves forward (on average), its velocity should be positive. If we alter the rates according to equations (35) and (36) we find an expression for the velocity as a function of an applied force:

$$\langle V \rangle (F) = \frac{k_{\downarrow}}{k_{\uparrow} + k_{\downarrow}} k_{+} e^{+\frac{F}{2}} - \frac{k_{\uparrow}}{k_{\uparrow} + k_{\downarrow}} k_{-} e^{-\frac{F}{2}} \tag{46}$$

When the distribution function,$\Psi(s)$, of a passage problem is known, this can also be used to find an approximate expression for the force. The velocity per unit length:

$$\langle V \rangle = \langle \frac{1}{t} \rangle \tag{47}$$

The distribution function is the moment generating function (see equation (6)):

$$\langle V \rangle = \langle \frac{1}{t} \rangle \approx \frac{1}{\langle t \rangle} = \Big(\frac{\partial \Psi}{\partial s} \Big|_{s=0} \Big)^{-1} \tag{48}$$

Using a computer program, such as Maple or Mathematica, to evaluate the derivative we find after quite some rewriting and simplifying:

$$\langle V \rangle (F) = \frac{(k_\downarrow k_+ e^{\frac{F}{2}} - k_\uparrow k_- e^{-\frac{F}{2}}) e^{-\frac{F}{2}} (k_\uparrow + k_+ e^{\frac{F}{2}})^2}{k_+^2 e^{\frac{F}{2}} k_\downarrow + e^{-\frac{F}{2}} k_\uparrow^3 + 2k_+ k_\uparrow^2 + k_\uparrow^2 e^{-\frac{F}{2}} k_\downarrow + k_+^2 e^{\frac{F}{2}} k_\uparrow + 2k_+ k_\uparrow k_\downarrow} \tag{49}$$

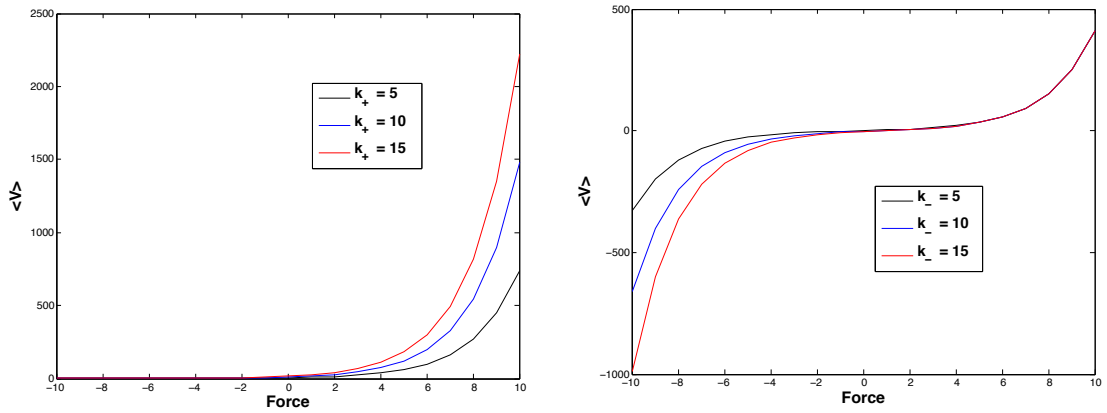
It turns out that the denominator can be rewritten quite conveniently:

$$\begin{aligned} \langle V \rangle (F) &= \frac{(k_\downarrow k_+ e^{\frac{F}{2}} - k_\uparrow k_- e^{-\frac{F}{2}}) e^{-\frac{F}{2}} (k_\uparrow + k_+ e^{\frac{F}{2}})^2}{(k_\uparrow + k_\downarrow) e^{-\frac{F}{2}} (k_\uparrow + k_+ e^{\frac{F}{2}})^2} \\ &= \frac{k_\downarrow}{k_\uparrow + k_\downarrow} k_+ e^{+\frac{F}{2}} - \frac{k_\uparrow}{k_\uparrow + k_\downarrow} k_- e^{-\frac{F}{2}} \end{aligned} \tag{50}$$

From which we recover the exact same expression as with the more instructive method of using the Master equations. Note however, that this is somewhat lucky, since in principle:

$$\langle \frac{1}{t} \rangle \neq \frac{1}{\langle t \rangle}$$

Below, in figure (11(a)), a plot of $\langle V \rangle (F)$ is shown for the case that $k_\uparrow = k_- = 0$, such that equation (30) is al ways satisfied.

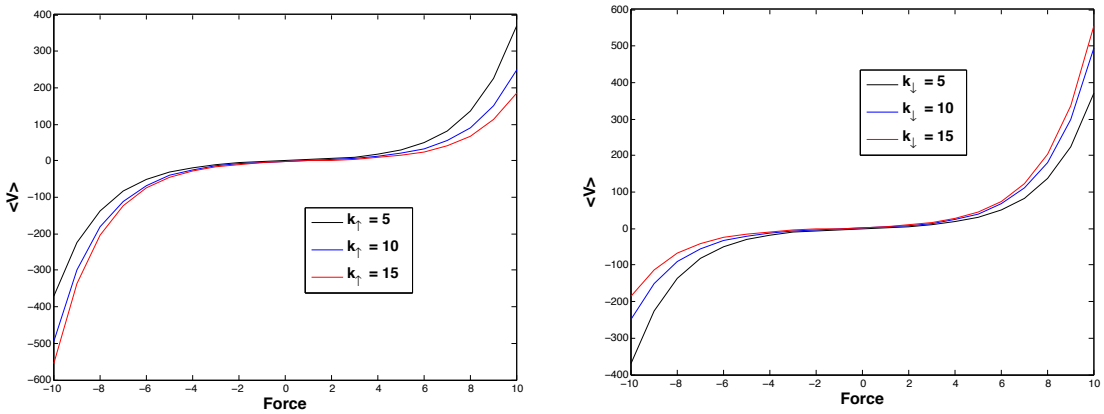


(a) *Velocity VS Force for different $k_+(0)$.*

(b) *Velocity VS Force for different $k_-(0)$.*

Figure 11: *Figure (a) at zero $k_\uparrow$,k_- and constant $k_\downarrow = 5$.Figure (b)at $k_\uparrow$=4 ,k_+=5 and $k_\downarrow = 5$.*

The velocity is positive at all forces, as expected. Its slope is proportional to k_+. This makes sense, since a higher forward rate should make the DNAP walk faster. Figure (11(b)) shows the effect of changing k_-, the backward rate. The constant values of the other rates are chosen such that initially $k_+(0)k_\downarrow > k_-(0)k_\uparrow$. As we increase the force, the velocity increases, just as in the previous case. However, when we decrease the force (stronger force in opposite/negative direction), the velocity becomes negative. For negative forces the backward rate increases (and forward rate decreases). The velocity becomes negative at the point where the force has decreased until the point that equation (30) is no longer satisfied. Any part of the graph beyond this point does not represent any physical system any more and therefore has no relevance. The point at which $\langle V \rangle$ becomes negative moves closer towards the origin as $k_-(0)$ increases. Or put differently, the negative slope becomes steeper as we increase the backward rate. Finally, let us consider the effect of altering $k_\uparrow$ or $k_\downarrow$.



(a) *Velocity VS Force for different* $k_{\uparrow}$

(b) *Velocity VS Force for different* $k_{\downarrow}$

Figure 12: *Figure (a) at* k_{+}*=5* $,k_{-}$*=5 and* $k_{\downarrow}$*=5. Figure (b) at* k_{+}*=5* $,k_{-}$*=5,*$k_{\uparrow} = 5$.

Altering upward or downward rate(s) effects both the slope in the physical regime and the point at which the non-physical regime starts. This is as expected, since both rates appear both in the numerator and denominator of equation (37). Also, we see that increasing $k_{\uparrow}$ qualitatively has the opposite effect of increasing $k_{\downarrow}$.

4 Model for T7 DNA polymerase

The general model discussed in the previous section left out three (conformational) states the polymerase can be found in.

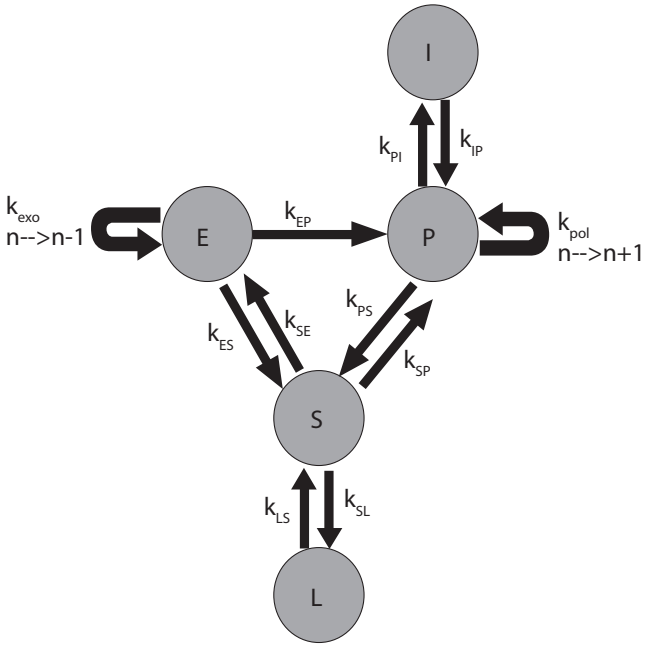


Figure 13: *Connected diagram including all possible conformational states for the T7 DNAP. As adapted from [5].*

One state accounts for the possibility that the polymerase unbinds from the DNA and remains free in solution (S). When looking at experimental traces, one observes that a DNAP will not only add or excise nucleotides, but it may also pause for a certain amount of time while doing neither of the two. In the shown diagram the state (L) indicates such a (long) pause state. The incorporation pause (I) can only be entered from and exited to the (P) state. Experimental data revealed that a polymerase that was adding nucleotides before it paused, will be more more likely to continue adding new nucleotides afterwards. Although this explains the placing of the I-node, it is not known what the exact conformational state looks like.The diagram shown can be translated into a board game that possesses translational symmetry, similar to figure (6). . Instead of solving for the generating function from scratch, we will employ the general solution given in the previous section. There, Ψ was given in terms of the ϕ's, the exact same relation will still hold. All we need to do is find out to what $\phi_{\uparrow}$,$\phi_{\downarrow}$,ϕ_{-} and ϕ_{+} correspond to in terms of the ϕ_{XY} belonging to the diagram above. Before we can embark on doing so, there is one more "rule of the game" that we will have to use. This rule concerns standing on a node as shown below:

4.1 One last rule: Combinatorics of stepping back and forth

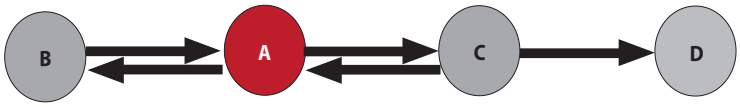


Figure 14: *Example 5*

We start on node A and seek the generating function describing our first arrival at node D. We see that, before eventually walking to D, we are allowed to walk back and forth between A and B several times. Similarly, we are aloud to walk back and forth between A and C as often as we like. We even have the option of walking the path A–B–A–C–A–B–A–C–D, or any other combination in which we toggle between the nodes A,B and C before walking to D. Previous examples showed how dealing with a single "two-way-bond" will result in geometric series with common ratios such as $\phi_{A-B}\phi_{B-A}$ or $\phi_{A-C}\phi_{C-A}$. Notice that even in the most complicated paths these two factors still appear. Every ϕ_{A-B} always forms a pair with a ϕ_{B-A}. Similarly, for every ϕ_{C-A} there is a ϕ_{A-C}. Of course there is one additional ϕ_{A-C} at the end of every path. Naively, one may think that Ψ looks as follows:

$$\Psi_{AD} = \sum_{n=0}^{\infty}(\phi_{A-B}\phi_{B-A})^n \sum_{m=0}^{\infty}(\phi_{A-C}\phi_{C-A})^m \phi_{A-C}\phi_{C-D}$$

Although it is true that any acceptable path can be grouped into one of the terms present in the above, we are still missing an important feature. There are many different paths that will lead to the same first passage time. A first passage time for which there are more paths leading to it should become more likely. So even though the exact order of terms in a product does not matter, we must still count the number of paths that lead to the same product term. The factor we are missing is a combinatorial factor describing the number of ways the pairs of $\phi_{A-B}\phi_{B-A}$ and $\phi_{A-C}\phi_{C-A}$ can be commuted through the product term. To find the correct factor, we fortunately do not have to do explicit combinatorics. In stead, we will approach the problem in a similar fashion as was done for the more simpler examples. There, we characterized a particular path by the number of times one steps back and forth. To illustrate how the same can be done here, consider the following example. Let us say that we walk back and forth twice, but we do not know which exact nodes we will use. The following paths are possible:

- **Use AB twice:** walk A-B-A-B-A(-C-D)

$$(\phi_{A-B}\phi_{B-A})^2$$

- **Use AC twice:** walk A-C-A-C-A(-C-D)

$$(\phi_{A-C}\phi_{C-A})^2$$

- **Use AB once and AC once:** walk A-B-A-C-A(-C-D)

$$\phi_{A-B}\phi_{B-A}\phi_{A-C}\phi_{C-A}$$

  or walk A-C-A-B-A(-C-D)

$$\phi_{A-C}\phi_{C-A}\phi_{A-B}\phi_{B-A}$$

We see that we need a contribution of:

$$(\phi_{A-B}\phi_{B-A})^2 + 2\phi_{A-B}\phi_{B-A}\phi_{A-C}\phi_{C-A} + (\phi_{A-C}\phi_{C-A})^2 = (\phi_{A-B}\phi_{B-A} + \phi_{A-C}\phi_{C-A})^2$$

to Ψ_{AD}. Writing down all possible paths in which you walk back and forth more often will result in higher orders of $(\phi_{A-B}\phi_{B-A} + \phi_{A-C}\phi_{C-A})^n$ contributing to Ψ_{A-D}. The correct combinatorial factors happen to equal the numbers appearing in Pascal's triangle. Therefore, we conclude that when dealing with a node such as A

$$\Psi_{A-D} = \sum_{n=0}^{\infty} (\phi_{A-B}\phi_{B-A} + \phi_{A-C}\phi_{C-A})^n \phi_{A-C}\phi_{C-D} \tag{51}$$

is the correct form of the distribution function that must be used.

4.2 Finding the effective $\phi's$

Now we can calculate the effective ϕ_+,ϕ_-,$\phi_\uparrow$,$\phi_\downarrow$ in terms of the new ϕ's. Let us start with $\phi_\downarrow$:

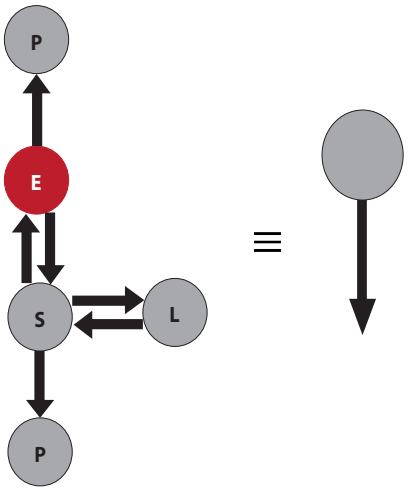


Figure 15: *Figure indicating all possible first passages at P starting from E*

Starting in the exo-state, the first arrival at P is characterized by:

$$\begin{aligned} \phi_\downarrow &= \phi_{EP}\Big[1 + \phi_{ES}\sum_{n=0}^{\infty}(\phi_{ES}\phi_{SE} + \phi_{SL}\phi_{LS})^n\phi_{SE}\Big] + \phi_{ES}\sum_{m=0}^{\infty}(\phi_{ES}\phi_{SE} + \phi_{SL}\phi_{LS})^m\phi_{SP} \\ &= \phi_{EP}\frac{1-\phi_{SL}\phi_{LS}}{1-\phi_{SL}\phi_{LS}-\phi_{SE}\phi_{ES}} + \frac{\phi_{ES}\phi_{SP}}{1-\phi_{SL}\phi_{LS}-\phi_{SE}\phi_{ES}} \end{aligned} \tag{52}$$

Now let us seek for the upward rate, $\phi_\uparrow$. Figure (16) indicates all first passages at E starting from P. A convenient way of redrawing the diagram (renaming/defining some nodes) makes it easy to find $\phi_\uparrow$.

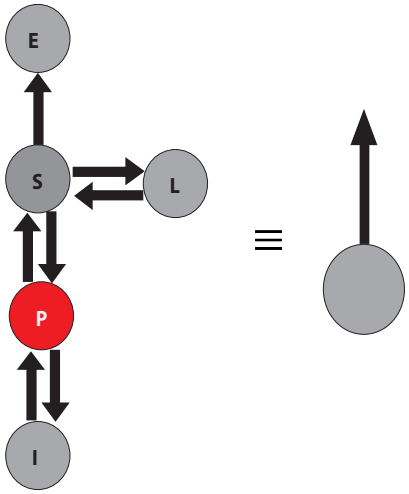


Figure 16: *Figure indicating all possible first passages at E starting from P*

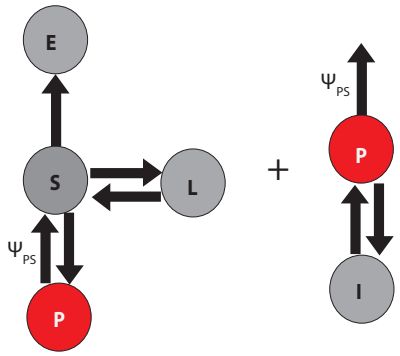


Figure 17: *Convienient way of drawing the diagram of figure (16).*

Using the convenient split up shown in the figure, we see that:

$$\Psi_{PS} = \sum_{n=0}^{\infty} (\phi_{PI}\phi_{IP})^n \phi_{PS} \tag{53}$$

Using this results in:

$$\begin{aligned}\phi_{\uparrow} &= \Psi_{PS}\sum_{m=0}^{\infty}(\phi_{SP}\Psi_{PS}+\phi_{SL}\phi_{LS})^m\phi_{SE}\\ &= \frac{\Psi_{PS}\phi_{SE}}{1-\phi_{SP}\Psi_{PS}-\phi_{SL}\phi_{LS}}\\ &= \frac{\phi_{PS}}{1-\phi_{IP}\phi_{PI}}\times\frac{\phi_{SE}}{1-\frac{\phi_{SP}\phi_{PS}}{1-\phi_{IP}\phi_{PI}}-\phi_{SL}\phi_{LS}}\\ &= \frac{\phi_{PS}\phi_{SE}}{1-\phi_{IP}\phi_{PI}-\phi_{SP}\phi_{PS}-\phi_{SL}\phi_{LS}+\phi_{SL}\phi_{LS}\phi_{IP}\phi_{PI}}\end{aligned} \tag{54}$$

To find ϕ_{-}, we could draw a diagram as in figure (16), but now one that would indicate all types of first passages/arrivals at a node E at site n-1 starting from an E at n. If we start at the drawn E and record our arrival at its neighboring E that is not shown here:

$$\begin{aligned}\phi_{-} &= \phi_{exo}\Big[1+\sum_{n=0}^{\infty}(\phi_{ES}\phi_{SE}+\phi_{SL}\phi_{LS})^n\phi_{ES}\phi_{SE}\Big]\\ &= \phi_{exo}\Big(1+\frac{\phi_{ES}\phi_{SE}}{1-\phi_{SL}\phi_{LS}-\phi_{ES}\phi_{SE}}\Big)\\ &= \phi_{exo}\times\frac{1-\phi_{SL}\phi_{LS}}{1-\phi_{SL}\phi_{LS}-\phi_{ES}\phi_{SE}}\end{aligned} \tag{55}$$

Finally, we search for ϕ_{+}. For this we must consider all types of first passages at P at n+1 starting from a P at n. Just as in finding $\phi_{\uparrow}$ a convenient manner of splitting up the problem is helpful: We define:

$$\Psi_{PS} = \sum_{n=0}^{\infty}(\phi_{IP}\phi_{PI})^n\phi_{PS} \tag{56}$$

$$\Psi_{pol} = \sum_{n=0}^{\infty}(\phi_{IP}\phi_{PI})^n\phi_{pol} \tag{57}$$

We conclude:

$$\begin{aligned}\phi_{+} &= \Psi_{pol}\Big[1+\sum_{m=0}^{\infty}(\phi_{SP}\Psi_{PS}+\phi_{SL}\phi_{LS})^m\Psi_{PS}\phi_{SP}\Big]\\ &= \Psi_{pol}\Big(1+\frac{\Psi_{PS}\phi_{SP}}{1-\phi_{SP}\Psi_{PS}-\phi_{SL}\phi_{LS}}\Big)\\ &= \Psi_{pol}\times\frac{1-\phi_{SL}\phi_{LS}}{1-\phi_{SP}\Psi_{PS}-\phi_{SL}\phi_{LS}}\\ &= \frac{\phi_{pol}}{1-\phi_{IP}\phi_{PI}}\times\frac{1-\phi_{SL}\phi_{LS}}{1-\frac{\phi_{SP}\phi_{PS}}{1-\phi_{IP}\phi_{PI}}-\phi_{SL}\phi_{LS}}\\ &= \phi_{pol}\times\frac{1-\phi_{SL}\phi_{LS}}{1-\phi_{IP}\phi_{PI}-\phi_{SP}\phi_{PS}-\phi_{LS}\phi_{SL}+\phi_{SL}\phi_{LS}\phi_{IP}\phi_{PI}}\end{aligned} \tag{58}$$

To complete the translation of this specific model for T7 DNAP to the more general model described previously we must find the ϕ_{XY}'s in terms of the k_{XY}'s.

4.3 Finding the ϕ_{XY}'s in terms of the k_{XY}'s

For every type of node (E,P,S,L and I) we must consider first passages at neighboring nodes. When standing on a P node:

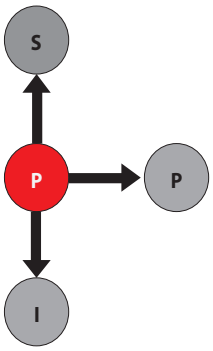


Figure 18: *Figure indicating all first passages/arrivals at neighboring nodes,starting on a P.*

The first passages are characterized by:

$$\partial_t P_P = -(k_{PI} + k_{PS} + k_{pol})P_P \tag{59}$$
$$k_{PI}P_P\mathrm{d}t = \phi_{PI}\mathrm{d}t \tag{60}$$
$$k_{PS}P_P\mathrm{d}t = \phi_{PS}\mathrm{d}t \tag{61}$$
$$k_{pol}P_P\mathrm{d}t = \phi_{pol}\mathrm{d}t \tag{62}$$

A P_P denotes the probability of finding a DNAP on a node P. The solution to the set of equations are:

$$\phi_{PS}(s) = \frac{k_{PS}}{s + k_{PI} + k_{PS} + k_{pol}} \tag{63}$$
$$\phi_{PI}(s) = \frac{k_{PI}}{s + k_{PI} + k_{PS} + k_{pol}} \tag{64}$$
$$\phi_{pol}(s) = \frac{k_{pol}}{s + k_{PI} + k_{PS} + k_{pol}} \tag{65}$$

Similar reasoning for standing on a E or S, yields:

$$\phi_{ES}(s) = \frac{k_{ES}}{s + k_{ES} + k_{EP} + k_{exo}} \tag{66}$$
$$\phi_{EP}(s) = \frac{k_{EP}}{s + k_{ES} + k_{EP} + k_{exo}} \tag{67}$$
$$\phi_{exo}(s) = \frac{k_{exo}}{s + k_{ES} + k_{EP} + k_{exo}} \tag{68}$$
$$\phi_{SP}(s) = \frac{k_{SP}}{s + k_{SP} + k_{SE} + k_{SL}} \tag{69}$$
$$\phi_{SE}(s) = \frac{k_{SE}}{s + k_{SP} + k_{SE} + k_{SL}} \tag{70}$$
$$\phi_{SL}(s) = \frac{k_{SL}}{s + k_{SP} + k_{SE} + k_{SL}} \tag{71}$$

Although, we have not encountered any node with only one type of neighbor, the equations for it are actually quite simple:
Since,

$$\partial_t P_L = -k_{LS}P_L \tag{72}$$
$$k_{LS}P_L\mathrm{d}t = \phi_{LS}\mathrm{d}t \tag{73}$$

we have that:

$$\phi_{LS}(s) = \frac{k_{LS}}{s + k_{LS}} \tag{74}$$

In the same way we find that:

$$\phi_{IP}(s) = \frac{k_{IP}}{s + k_{IP}} \tag{75}$$

Now the translation is complete. To check wether the relations actually hold, a simple check can be performed. Since we know that it is certain that eventually you will leave the node you are standing on:

$$\phi_{\uparrow}(0) + \phi_{+}(0) = 1 \tag{76}$$

$$\phi_{\downarrow}(0) + \phi_{-}(0) = 1 \tag{77}$$

From the given equations:

$$\phi_{\uparrow}(0) = \Big(\frac{\phi_{PS}\phi_{SE}}{1 - \phi_{IP}\phi_{PI} - \phi_{SP}\phi_{PS} - \phi_{SL}\phi_{LS} + \phi_{SL}\phi_{LS}\phi_{IP}\phi_{PI}}\Big)_{s=0} \tag{78}$$

After plugging in the relations in terms of the rates, multiplying numerator and denominator by $(k_{PS} + k_{PI} + k_{pol})(k_{SP} + k_{SE} + k_{SL})$ and expanding both one finds:

$$\phi_{\uparrow}(0) = \frac{k_{PS}k_{SE}}{k_{PS}k_{SE} + k_{pol}k_{SP} + k_{pol}k_{SE}} \tag{79}$$

By similar means, one finds that:

$$\begin{aligned} \phi_{+}(0) &= \Big(\phi_{pol} \times \frac{(1 - \phi_{SL}\phi_{LS})}{1 - \phi_{IP}\phi_{PI} - \phi_{SP}\phi_{PS} - \phi_{LS}\phi_{SL} + \phi_{SL}\phi_{LS}\phi_{IP}\phi_{PI}}\Big)_{s=0} \\ &= \frac{k_{pol}k_{SP} + k_{pol}k_{SE}}{k_{PS}k_{SE} + k_{pol}k_{SP} + k_{pol}k_{SE}} \end{aligned} \tag{80}$$

Equations (79) and (80) indeed add up to one. In the exact same way, by plugging in all the $\phi(0)$'s, one will find:

$$\phi_{\downarrow}(0) = \frac{k_{SP}k_{ES} + k_{EP}k_{SP} + k_{EP}k_{SE}}{k_{EP}k_{SP} + k_{SP}k_{ES} + k_{EP}k_{SE} + k_{exo}k_{SP} + k_{exo}k_{SE}} \tag{81}$$

$$\phi_{-}(0) = \frac{k_{exo}k_{SP} + k_{exo}k_{SE}}{k_{EP}k_{SP} + k_{SP}k_{ES} + k_{EP}k_{SE} + k_{exo}k_{SP} + k_{exo}k_{SE}} \tag{82}$$

These, in turn, add up to one as well. Satisfying equations (76) and (77) ensures that $\Psi(s)$ will be normalized once again. Now we know that making a step forward in figure (6) with adjustments according to figure (13) results in a $\Psi(s)$ given by the self-consistent equation (24) in which we plug in the relations just derived.

5 Dwell Time distributions

Dwell Time distributions will serve as an important link between single-molecule experiments and the random walk theory.

5.1 Defining the dwell time

For an RNA polymerase (RNAP), it is somewhat easier to understand what these distributions are. A trace of a RNA polymerase, as found in single-molecule experiments, typically looks as follows

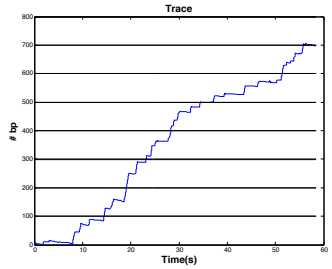


Figure 19: *A typical trace for RNAP with a resolution of 100bp. This trace was obtained by using the Gillespie simulation for DNAP (see section 6) with the settings k_+=100, k_-=2, $k_\uparrow$=5 and $k_\downarrow$=1. Simulation of 1000 iterations.*

Note that in these experiments, one only observes the RNAP to walk forward. From the trace, an experimenter would make a histogram of the lengths of the observed plateau's. Because the RNAP will either pause or walk forward, this histogram (when normalized correctly) is an approximation to the first passage time probability density for arrival at site N+1. More precisely, the theory of first passage times forms the mathematical construct necessary to calculate whatever it is these histograms approximate. The histograms are said to approximate the so called dwell time distribution. The Biological processes of transcription and replication are highly stochastic. In addition, any real measurement data is corrupted by noise and measurement errors. For these reasons the measurement data does not produce a smooth trace. Therefore, it typically becomes impossible to determine the dwell-time for walking from N→N+1. In stead, a window, or resolution, is set of n base pairs for which the dwell times/first passage times are recorded. Noise and "natural stochasticity" might be of the same order (if any, the measurement errors are larger), so smoothing the data until the point that hardly an noise is observed also removes relevant features of the data. Using a resolution as shown in figure (19)(there it is 100bp for the safe of illustration) has the advantage that the data does not have to be smoothened beforehand. Due to the characteristics of the RNAP, the dwell time would in principle be found as Ψ^n (Ψ for a single step forward). Unfortunately, this will not work when dealing with DNAP.

5.2 Difficulty for DNAP

In the model we use, we already saw that a DNA polymerase is allowed to walk backwards. This behavior is also observed in experiments. A simulated trace of DNAP is shown in figure (25). Again, we could set a resolution and record the dwell times. However, problems may arise when the DNAP walks backwards before eventually walking forward.

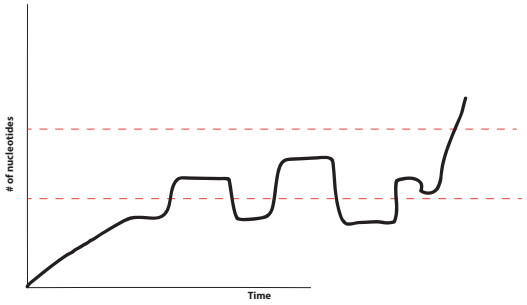


Figure 20: *Sketch indicating that recording dwell times for DNAP becomes more difficult. Resolution indicated with red dotted lines.*

We would want to record the time between the first passage of the lower dottedline and the first exit by passing the top line. However, we must prevent that we accidentally record the time between the second or third entrance and exit. Also, we see that a DNAP can exit below in stead of at the top. The same holds for entrance from above, which we do not want to interfere with a the measurement of the previous dwell time.

5.3 Solution: Adapt a different approach

Generally speaking, there are four possible ways of entering/exiting the window for the first time:

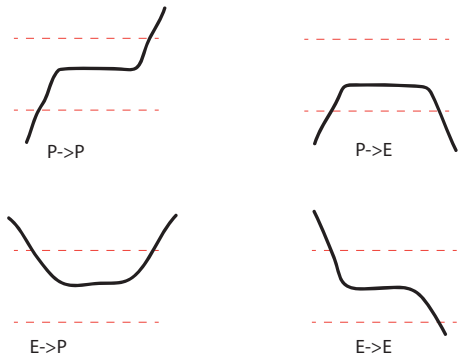


Figure 21: *Sketch indicating the four different dwell times for a DNAP trace. For every resolution bin,resolution indicated with red dotted lines, the DNAP can both enter and exit it from the polymerase or the exo-state line for the first time.*

Consequently, there are four different types of dwell times. We will now find generating functions belonging to each of these dwell time distributions separately. For a resolution of n base pairs, these will be denoted by $\Psi^{(n)}_{\uparrow\uparrow}$, $\Psi^{(n)}_{\uparrow\downarrow}$, $\Psi^{(n)}_{\downarrow\uparrow}$ and $\Psi^{(n)}_{\downarrow\downarrow}$ respectively. Let us start with the first of these. The desired dwell time distribution is found swiftly, once one realizes that there is a recursive relation that will determine it.

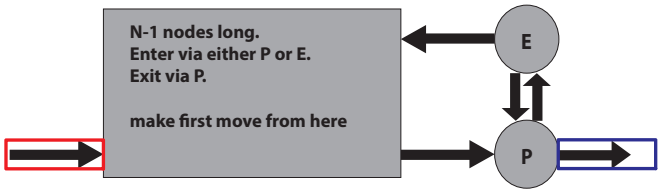


Figure 22: *Diagram that will determine* $\Psi^{(n)}_{\uparrow\uparrow}$. *The rectangle is their to indicate that it has both* $\Psi^{(n-1)}_{\uparrow\uparrow}$ *and* $\Psi^{(n-1)}_{\downarrow\uparrow}$ *associated with it. The boxed arrows indicate your first move (red) and last move (blue). The dwell time is measured from the moment you enter the n-long region. Hence, the red boxed arrow does not contribute to the dwell time.*

This figure will be our new board game. If we start on the square, what is the probability of walking out of this figure at the right hand side? By using all the rules and tricks described earlier we can see that:

$$\begin{aligned}\Psi^{(n)}_{\uparrow\uparrow} &= \Psi^{(n-1)}_{\uparrow\uparrow}\sum_{m=0}^{\infty}(\Psi_{\uparrow}\phi_{-}\Psi^{(n-1)}_{\downarrow\uparrow})^{m}\Psi_{+} \\ &= \frac{\Psi_{+}\Psi^{(n-1)}_{\uparrow\uparrow}}{1-\Psi_{\uparrow}\phi_{-}\Psi^{(n-1)}_{\downarrow\uparrow}} \end{aligned} \tag{83}$$

$$\Psi^{(1)}_{\uparrow\uparrow} = \Psi_{+} \tag{84}$$

In which, (the last two will make there appearance soon):

$$\Psi_{+} = \frac{\phi_{+}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{85}$$

$$\Psi_{\uparrow} = \frac{\phi_{\uparrow}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{86}$$

$$\Psi_{-} = \frac{\phi_{-}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{87}$$

$$\Psi_{\downarrow} = \frac{\phi_{\downarrow}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{88}$$

Note, that it is also possible to find $\Psi^{(n)}_{\uparrow\uparrow}$ by using the exact mirror image of the board used now. This yields

$$\Psi^{(n)}_{\uparrow\uparrow} = \Psi^{(n-1)}_{\uparrow\uparrow}\sum_{m=0}^{\infty}(\Psi_{\downarrow}\phi_{+}\Psi^{(n-1)}_{\uparrow\downarrow})^{m}\Psi_{+}$$

which is precisely the same as that of equation (83). Let us find $\Psi^{(n)}_{\downarrow\uparrow}$ by using the following board/diagram:

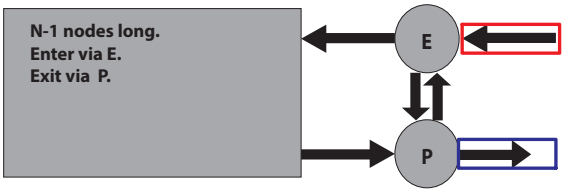


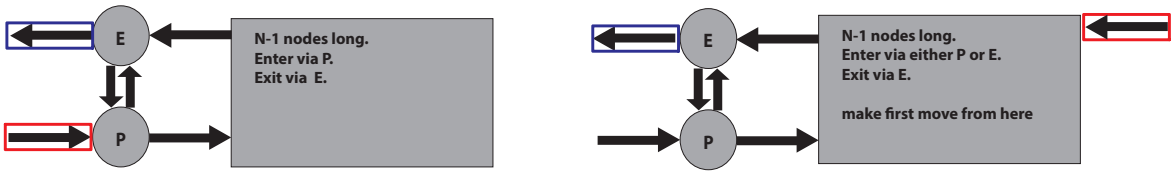
Figure 23: *Diagram that will determine* $\Psi^{(n)}_{\downarrow\uparrow}$. *The rectangle is their to indicate that it has* $\Psi^{(n-1)}_{\downarrow\uparrow}$ *associated with it. Red boxed arrow indicates that you enter via an E node. Blue boxed arrow is your last move, exiting via a P node.*

We start on the red node. We are looking for the time at which we leave the board at the same side from the polymerase line.

$$\begin{aligned}\Psi_{\downarrow\uparrow}^{(n)} &= \phi_{\downarrow}\sum_{m=0}^{\infty}(\Psi_{\uparrow}\phi_{-}\Psi_{\downarrow\uparrow}^{(n-1)})^{m}\Psi_{+} + \phi_{-}\Psi_{\downarrow\uparrow}^{(n-1)}\sum_{m=0}^{\infty}(\Psi_{\uparrow}\phi_{-}\Psi_{\downarrow\uparrow}^{(n-1)})^{m}\Psi_{+} \\ &= \frac{\Psi_{+}[\phi_{\downarrow}+\phi_{-}\Psi_{\downarrow\uparrow}^{(n-1)}]}{1-\Psi_{\uparrow}\phi_{-}\Psi_{\downarrow\uparrow}^{(n-1)}}\end{aligned} \tag{89}$$

$$\Psi_{\downarrow\uparrow}^{(1)} = \phi_{\downarrow}\Psi_{+} = \frac{\phi_{\downarrow}\phi_{+}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{90}$$

The first term of equation (89) describes all paths in which you go to the pol.-line before entering the square. The second term describes all paths that explicitly take you to the square first. The following figures show the diagrams used for the $\Psi_{\uparrow\downarrow}^{(n)}$ and $\Psi_{\downarrow\downarrow}^{(n)}$.



(a) (b)

Figure 24: *(a)Diagram that will determine $\Psi_{\uparrow\downarrow}^{(n)}$. The rectangle is their to indicate that it has $\Psi_{\uparrow\downarrow}^{(n-1)}$ associated with it. Red boxed arrow indicates that you enter via a P node. Blue boxed arrow is your last move, exiting via an E node. (b)Diagram that will determine $\Psi_{\downarrow\downarrow}^{(n)}$. The rectangle is their to indicate that it has both $\Psi_{\downarrow\downarrow}^{(n-1)}$ and $\Psi_{\uparrow\downarrow}^{(n-1)}$ associated with it. Red boxed arrow indicates that you enter via an E node. Blue boxed arrow is your last move, exiting via an E node.*

We see that these are the exact mirror images of figures (22) and (23). Therefore, we see that all we need to do is replace a "↑" by a "↓" and every "+" by a "-", and vice versa, in equations (83),(84),(89) and (90).

$$\Psi_{\uparrow\downarrow}^{(n)} = \frac{\Psi_{-}[\phi_{\uparrow}+\phi_{+}\Psi_{\uparrow\downarrow}^{(n-1)}]}{1-\Psi_{\downarrow}\phi_{+}\Psi_{\uparrow\downarrow}^{(n-1)}} \tag{91}$$

$$\Psi_{\uparrow\downarrow}^{(1)} = \phi_{\uparrow}\Psi_{-} = \frac{\phi_{\uparrow}\phi_{-}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{92}$$

$$\begin{aligned}\Psi_{\downarrow\downarrow}^{(n)} &= \Psi_{\downarrow\downarrow}^{(n-1)}\sum_{m=0}^{\infty}(\Psi_{\downarrow}\phi_{+}\Psi_{\uparrow\downarrow}^{(n-1)})^{m}\Psi_{-} \\ &= \frac{\Psi_{-}\Psi_{\downarrow\downarrow}^{(n-1)}}{1-\Psi_{\downarrow}\phi_{+}\Psi_{\uparrow\downarrow}^{(n-1)}}\end{aligned} \tag{93}$$

$$\Psi_{\downarrow\downarrow}^{(1)} = \Psi_{-} = \frac{\phi_{-}}{1-\phi_{\uparrow}\phi_{\downarrow}} \tag{94}$$

Although the fact that the dwell time distributions satisfy recursive relations made it relatively simple to construct them, solving for them becomes difficult. Also, we see that the distributions are not independent of each other. To simplify matters somewhat, we will not adapt the relations of the ϕ's in terms of the k's of the full T7 DNAP model. In stead we will use those of equations (25)-(28) in what follows. Also, the resolution will be set to n=2.

6 The Gillespie Algorithm and Maximum Likelihood Fitting

The dwell time distributions of equations (83),(91),(89) and (93) provided with equations (25)-(28) and a resolution of n=2 will be used to fit onto simulated data. In principle it should be possible to fit it onto experimental data. This section describes the used algorithm and fitting method.

6.1 The Gillespie algorithm

The Gillespie algorithm is a Monte Carlo algorithm that is well suited for the type of first passage problems considered here. Initially, we will assume that we start on the P-node on site n. The parameter s will keep track of the type of node we are at. Let s=0 on a P and s=1 on an E. The main loop consists of three conceptual steps:

- **Generate a time:** Generate a time (interval) from the Poissonian distribution:

$$P(\Delta t|k_\uparrow + k_+) = (k_\uparrow + k_+)e^{-(k_\uparrow + k_+)\Delta t}$$

  to decide how long it will take to leave the present node. As shown before, the probabilities of leaving a node are decreased significantly after a time $(k_\uparrow + k_+)^{-1}$, making the Poisson distribution appropriate.

- **Select an event/ a jump:** To decide on what node we are going to walk to after having waited for the generate amount of time, the probabilities $\phi_\uparrow(0)$ and $\phi_+(0)$ can be used:

$$P_\uparrow = \frac{k_\uparrow}{k_+ + k_\uparrow}$$

$$P_+ = \frac{k_+}{k_+ + k_\uparrow}$$

  That is, at all times , it is more likely for us to leave the node we are at via the route with the highest rate.

- **Perform step and repeat:** Store the time ,$t \to t + \Delta t$, and the node you are at $n \to n$ and $s \to 1$ or $n \to n + 1$ and $s \to 0$.

At the end of the simulation we plot the node number versus the time. The following figure shows the result of such a simulation.

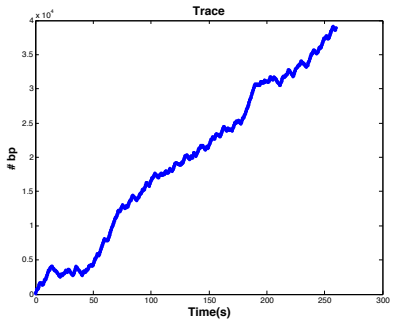


Figure 25: *Simulated trace. Parameter values of :* k_+*=500,*k_-*=250,*$k_\uparrow = k_\downarrow$*=1.*

This precisely mimics an experimental trace of a DNAP.

6.2 Maximum likelihood fitting

Goal of maximum likelihood fitting is to find the set of parameter values , the rates, such that the underlying model is most likely to reproduce the generated data. In general, one possesses some model with parameters $\{k_j\}_{j\in[1,N]}$ and a data set $\{t_i\}_{i\in[1,T]}$. The model will predict the probability of obtaining the dataset for a given set of parameter values, $P(\{t_i\}|\{k_j\})$. In our case this is simply the dwell time distribution as a function of the set of rates. We may ask ourselves: *" If the model correctly describes the data, what parameter values are the ones we (most likely) must use to mimic the data?"*. Mathematically, this translates into the following statement:

$$\partial_{k_j} P(\{k\}_j|\{t\}_i) = 0 \quad \forall j \tag{95}$$

In here, $P(\{k\}_j|\{t\}_i)$ denotes the probability of obtaining a particular set of parameter values given the dataset. How do we go about doing this? Let us use the defining property of a conditional probability:

$$\begin{aligned} P(\{t_i\},\{k_j\}) &= P(\{t_i\}|\{k_j\})P(\{k_j\}) \\ &= P(\{k_j\}|\{t_i\})P(\{t_i\}) \end{aligned} \tag{96}$$

By assumption, $P(\{k_j\})$, will be chosen to be uniform. Hence, it is parameter independent. This reflects that we in principle have no a priori knowledge of what the (most likely) parameter values should be. Under this assumption, we see that:

$$P(\{k_j\}|\{t_i\}) \propto P(\{t_i\}|\{k_j\}) \tag{97}$$

We define the Likelihood function as follows:

$$L(\{k_j\},\{t_i\}) = P(\{t_i\}|\{k_j\}) = \prod_i P(t_i|\{k_j\}) \tag{98}$$

The second equality expresses that the measured data points are independent of each other. In practice, one does not maximize the likelihood, but minimizes the negative logarithm of L or negative loglikelihood:

$$\begin{aligned} ML(\{k_j\},\{t_i\}) &= -ln(L(\{k_j\},\{t_i\})) \\ &= -\sum_i ln(P(t_i|\{k_j\})) \end{aligned} \tag{99}$$

The set of parameters that minimize ML,

$$\partial_{k_j} ML = 0 \quad \forall j \tag{100}$$

are called the maximum likelihood estimators. In our case, we have four different models for four different types of datasets. Each of them belonging to a particular type of dwell time (distribution):"$\uparrow\uparrow$, $\uparrow\downarrow$, $\downarrow\uparrow$ and $\downarrow\downarrow$". The maximum likelihood function gets altered slightly:

$$ML_\alpha(\{k_j\},\{t_i\}) = -\sum_i ln(P_\alpha(t_i^\alpha|\{k_j\})) \ \forall\alpha = 1,2,3,4 \tag{101}$$

By equation (97) we know that we need the logarithm of $\mathcal{L}^{-1}\{\Psi_\alpha^{(2)}(s)\}$'s to construct ML, these are the physical dwell time distributions.

6.3 Finding the inverse $\mathcal{L}$-transforms

Only two out of four inverse Laplace transforms will be given explicitly. The other two our found by interchanging upward and downward rates and forward and backward rates. Let us start with, $\Psi_{\downarrow\uparrow}^{(2)}$, since

it is not dependent on one of the other dwell time distributions. Starting from equations (89) and (90) we find:

$$\begin{aligned}\Psi^{(2)}_{\downarrow\uparrow} &= \frac{\Psi_+[\phi_\downarrow + \phi_-\Psi^{(1)}_{\downarrow\uparrow}]}{1-\Psi_\uparrow\phi_-\Psi^{(1)}_{\downarrow\uparrow}} \\ &= \frac{\Psi_+[\phi_\downarrow + \phi_-\phi_\downarrow\Psi_+]}{1-\Psi_\uparrow\Psi_+\phi_\downarrow\phi_-} \\ &= \frac{\phi_+}{1-\phi_\uparrow\phi_\downarrow} \times \frac{\phi_\downarrow(1-\phi_\uparrow\phi_\downarrow)^2 + \phi_-\phi_\downarrow\phi_+(1-\phi_\uparrow\phi_\downarrow)}{(1-\phi_\uparrow\phi_\downarrow)^2 - \phi_+\phi_-\phi_\uparrow\phi_\downarrow} \\ &= \frac{\phi_+\phi_\downarrow(1-\phi_\uparrow\phi_\downarrow) + \phi_+^2\phi_-\phi_\downarrow}{1-2\phi_\uparrow\phi_\downarrow + \phi_\uparrow^2\phi_\downarrow^2 - \phi_+\phi_-\phi_\uparrow\phi_\downarrow} \\ &\equiv \frac{k_+k_\downarrow(s+k_++k_\uparrow)(s+k_-+k_\downarrow) - k_+k_\uparrow k_\downarrow^2 + k_+^2k_-k_\downarrow}{As^4+Bs^3+Cs^2+Ds+E}\end{aligned} \tag{102}$$

with:

$$A = 1 \tag{103}$$

$$B = 2(k_+ + k_- + k_\uparrow + k_\downarrow) \tag{104}$$

$$C = (k_+ + k_\uparrow)^2 + 4(k_+ + k_\uparrow)(k_- + k_\downarrow) + (k_- + k_\downarrow)^2 - 2k_\uparrow k_\downarrow \tag{105}$$

$$D = 2(k_- + k_\downarrow)(k_+ + k_\uparrow)^2 + 2(k_+ + k_\uparrow)(k_- + k_\downarrow)^2 - 2k_\uparrow k_\downarrow(k_+ + k_- + k_\uparrow + k_\downarrow) \tag{106}$$

$$E = (k_+ + k_\uparrow)^2(k_- + k_\downarrow)^2 - 2k_\uparrow k_\downarrow(k_+ + k_\uparrow)(k_- + k_\downarrow) + k_\uparrow^2k_\downarrow^2 - k_+k_-k_\uparrow k_\downarrow \tag{107}$$

By using the Residue theorem, equations (22) and (23),it is now readily found that:

$$\begin{aligned}\mathcal{L}^{-1}\{\Psi^{(2)}_{\downarrow\uparrow}\} = &\frac{k_+k_\downarrow(s_1+k_++k_\uparrow)(s_1+k_-+k_\downarrow) - k_+k_\uparrow k_\downarrow^2 + k_+^2k_-k_\downarrow e^{s_1t}}{(s-s_2)(s-s_3)(s-s_4)} \\ &+ \frac{k_+k_\downarrow(s_2+k_++k_\uparrow)(s_2+k_-+k_\downarrow) - k_+k_\uparrow k_\downarrow^2 + k_+^2k_-k_\downarrow e^{s_2t}}{(s-s_1)(s-s_3)(s-s_4)} \\ &+ \frac{k_+k_\downarrow(s_3+k_++k_\uparrow)(s_3+k_-+k_\downarrow) - k_+k_\uparrow k_\downarrow^2 + k_+^2k_-k_\downarrow e^{s_3t}}{(s-s_1)(s-s_2)(s-s_4)} \\ &+ \frac{k_+k_\downarrow(s_4+k_++k_\uparrow)(s_4+k_-+k_\downarrow) - k_+k_\uparrow k_\downarrow^2 + k_+^2k_-k_\downarrow e^{s_4t}}{(s-s_1)(s-s_2)(s-s_3)}\end{aligned} \tag{108}$$

where s_1 through s_4 denote the four roots of the denominator of $\Psi^{(2)}_{\downarrow\uparrow}$. Fortunately, the roots of a fourth order polynomial can be found explicitly. For higher order polynomials this is not the case. The possibility of inverting the Laplace transform was one of the reasons for choosing a resolution of 2bp.

Let us now seek for $\mathcal{L}^{-1}\{\Psi^{(2)}_{\uparrow\uparrow}\}$. Similar to the above, starting from equations (83) and (84), plugging in the definitions of the $\Psi_{+,-,\uparrow,\downarrow}$, plugging in the relations of the ϕ's in terms of the rates and going through some steps of straight forward algebraic manipulations (expanding terms or adding fractions).

$$\begin{aligned}\Psi^{(2)}_{\uparrow\uparrow} &= \frac{\Psi_+\Psi^{(1)}_{\downarrow\uparrow}}{1-\Psi_\uparrow\phi_-\Psi^{(1)}_{\downarrow\uparrow}} \\ &= \frac{\Psi_+^2}{1-\Psi_\uparrow\phi_-\phi_\downarrow\Psi_+} \\ &= \frac{\phi_+^2}{(1-\phi_\uparrow\phi_\downarrow)^2} \times \frac{1}{1-\frac{\phi_\uparrow\phi_-\phi_\downarrow\phi_+}{(1-\phi_\uparrow\phi_\downarrow)^2}} \\ &\equiv \frac{k_+^2(s+k_-+k_\downarrow)^2}{As^4+Bs^3+Cs^2+Ds+E}\end{aligned} \tag{109}$$

With, somewhat fortunately, the exact same A, B, C, D, E as above. Again, by virtue of the Residue theorem:

$$\begin{aligned}\mathcal{L}^{-1}\{\Psi^{(2)}_{\uparrow\uparrow}\} = &\frac{k_+^2(s_1+k_-+k_\downarrow)^2 e^{s_1 t}}{(s-s_2)(s-s_3)(s-s_4)} \\ &+ \frac{k_+^2(s_2+k_-+k_\downarrow)^2 e^{s_2 t}}{(s-s_1)(s-s_3)(s-s_4)} \\ &+ \frac{k_+^2(s_3+k_-+k_\downarrow)^2 e^{s_3 t}}{(s-s_1)(s-s_2)(s-s_4)} \\ &+ \frac{k_+^2(s_4+k_-+k_\downarrow)^2 e^{s_4 t}}{(s-s_1)(s-s_2)(s-s_3)}\end{aligned} \tag{110}$$

6.4 Results

To obtain the data sets a Gillespie simulation is run for 10^5 iterations using the parameter values: k_+=500, k_-=250,$k_\uparrow$=1, $k_\downarrow$=1. These values are inspired by the measurement results presented in [5]. The minimization of the dwell time distributions with respect to the different parameters, needed for the Maximum likelihood fitting, is performed using the built-in MATLAB function for the Simulated Annealing method. The maximum likelihood estimators , from MATLAB, are : k_+=499.8354, k_-=251.6270, $k_\uparrow$=0.9379 and $k_\downarrow$=1.0622. Figure (26) shows the histograms of the datasets and curves of the dwell-time distributions with parameters equal to the maximum likelihood estimators.

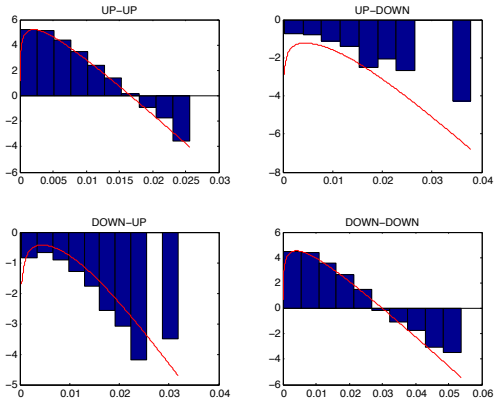


Figure 26: *Fit of data. Logarithm of normalized histograms of datasets in blue. Normalization is performed by using that* $\Psi^{(2)}_{\uparrow\downarrow}+\Psi^{(2)}_{\uparrow\uparrow}=1$ *and* $\Psi^{(2)}_{\downarrow\downarrow}+\Psi^{(2)}_{\downarrow\uparrow}=1$. *Red curves show the logarithm of the model with the parameter equal to the maximum likelihood estimators.Four different fits belong to the four different types of dwell times/ datasets*

The blue histograms are normalized in such a way that the total probability of exiting a n-long bin is 1. Mathematically:

$$\Psi^{(2)}_{\uparrow\downarrow} + \Psi^{(2)}_{\uparrow\uparrow} = 1 \tag{111}$$

$$\Psi^{(2)}_{\downarrow\downarrow} + \Psi^{(2)}_{\downarrow\uparrow} = 1 \tag{112}$$

We see that the red curves approximate the blue curves, or vise versa. However, the fit worked better for the $\uparrow\uparrow$ dwell times than for the $\uparrow\downarrow$ ones. This is simply due to the fact that the trace of figure (25) exhibits more

↑↑ than ↑↓ transitions. This is dictated by the choice of parameter values. The forward and downward rates are much larger than the upward and downward rates. Therefore, a DNAP walking forward is extremely likely to keep walking forward. Similar, a backward walking DNAP will continue to walk backwards. This explains that there are more data points acquired of the ↑↑ and ↓↓ dwell times than of the other types, resulting in better approximations of the red curves to the histograms in these cases.

7 Proofreading and error correction

So far we considered a model in which the DNAP is allowed to excise base pairs without actually asking ourselves what the biological reason for doing so would be. When replicating DNA, the polymerase places an incorrect nucleotide at the end of the chain every so many base pairs. That is, a nucleotide that does not match up correctly with the corresponding nucleotide on the template strand. The inability of these two nucleotides to form a correct Watson-Crick base pair makes placing an error energetically unfavorable. Based on energetics alone, so called pre-incorperation proofreading, the DNAP on average would place an erroneous nucleotides one every 500 sites. However, in practice DNAPs reach error fractions (# incorrect nucleotides / # correct nucleotides) of the order of 10^{-7}-10^{-9}. The polymerase must have developed, over the course of evolution, some kind of error correction mechanism. Here, we propose the post incorporation proofreading (PIP) scheme. The DNAP is allowed to excise base pairs, such that it will remove erroneous nucleotides.

7.1 Adjustments to the model

In order for the DNAP to actually reduce the error content in its product, the probability of removing an error should be higher than for removing a correct nucleotide. This features is incorporated into the model by letting the upward rates ($k_{\Uparrow}$), be higher for a small region of n nodes around the error. Since, the DNAP is not longer than a few base pairs, n should be some small number of order 1. We will use n=3-5. The energy landscape gets altered due to incorporation of an error. The rates into the exo-state are altered due to the energy difference ΔG. Rates out of the exo-state remain the same, since the energy barrier for walking from E-P does not change when an error is incorporated. We therefore have:

$$k_{\Uparrow} = k_{\uparrow} e^{-\frac{\Delta G}{k_B T}} \tag{113}$$

Experiments have tried to measure this ΔG in different ways. Based on the difference in binding energy of correct/incorrect base pairs one finds $\Delta G \approx 6 k_B T$. These experiments fail to include any effect of the DNAP. Experiments in which the DNAP was genetically engineered such that it has a malfunctioning exo-state active site give $\Delta G \approx -ln(10^{-6}) k_B T$. Drawback of these experiments is that the precise mutation used will effect the outcome. We expect the true ΔG to lie somewhere in between these to extremes.

7.2 Analogy with dwell times

We are interested in finding the probability of leaving an error in the final transcript. Below it will be explained how this will enable us to find the error fraction, the measure for the fidelity of the DNAP. To simplify matters, we assume that if the DNAP passes through the n-long region, the error will not get corrected anymore. Hence, we are looking for the probability of entering at site 1 on P and exiting on a P n sites further. This is precisely the dwell-time distribution $\Psi^{(n)}_{\uparrow\uparrow}(0)$. We will also need the probability of correcting an error. Given that it has been placed previously this becomes the probability of entering via the polymerase line and cutting back behind the error/exiting via the exostate line. This probability is given by $\Psi^{(n)}_{\uparrow\downarrow}(0)$. Off course, all upward rates must be adjusted numerically according to equation (113).

7.3 Efficiency measures and Evolutionary choices

To quantify the performance of a DNAP we introduce the following efficiency measures:

- **Error fraction:** By definition, the error fraction equals the number of incorrectly placed nucleotides divided by the number of correct nucleotides. The total number of nucleotides is approximately equal to the number of correct nucleotides. We will adapt the following form of the error fraction:

$$r = r_{pre} r_{post} \quad \text{with:} \tag{114}$$

$$r_{pre} = e^{-\frac{\Delta G}{k_B T}} \tag{115}$$

$$r_{post} = \frac{\Psi^{(n)}_{\uparrow\uparrow}(0)}{1 - \Psi^{(n)}_{\uparrow\uparrow}(0)} \approx \Psi^{(n)}_{\uparrow\uparrow}(0) = (1 - \Psi^{(n)}_{\uparrow\downarrow}(0)) \tag{116}$$

  where the same energy difference ,ΔG, is used as used in the altered upward rate. From the given definition of r it is not directly clear that it reduces to the above product of two terms. Never the less, the equation still has a physical interpretation. Namely, r will be a measure for the fraction of errors left in the transcript after both pre- and post incorporation proofreading.

- **Elongation efficiency:** Achieving high fidelity is not the only goal of a polymerase. DNA sequences must be transcribed with a sufficient rate to ensure the production of sufficient proteins inside the cell. We introduce a measure for how much is lost in terms of velocity by the use of PIP. Without PIP (that is, without any exo-state present) the polymerase steps forward with an average rate of k_+^{-1}. Including the exostate, the complete generating function of making a single step forward, $\Psi(s)$, has been calculated. From the generating function one can calculate the average time it takes to take such a step. The elongation efficiency is defined by:

$$\eta_{el} = \frac{\frac{1}{k_+}}{\langle t \rangle} = \frac{1}{k_+} \frac{1}{\langle t \rangle} \approx \frac{1}{k_+} \langle \frac{1}{t} \rangle \equiv \frac{\langle V \rangle}{k_+} \tag{117}$$

- **Producing functional transcripts:** If the entire transcript involves L base pairs, the probability of producing a functional (error free) transcript becomes

$$P_L(r) = (1 - r)^L \approx 1 - Lr \approx e^{-rL} \tag{118}$$

  In the last two approximations the Taylor series expansions of $(1 - r)^L$ and e^{-rL} are used.

- **Combined efficiency measure:** We define:

$$\chi_{el} = \eta_{el} P_L(r) \approx \frac{\langle V \rangle}{k_+} e^{-rL} \tag{119}$$

  This combined performance measure should be maximal for those set of parameter values for which error free transcripts can be produced with a sufficient rate.

Studies have shown [6] that different organisms, containing different DNAPs and having different genome lengths, actually all contain about the same amount of errors per produced genome. Apparently, evolution has kept the total number of mutations per generated copy of DNA constant amongst different organisms. To achieve this, the error rates are tuned by several orders of magnitude. Drake et al. report on an average amount of errors of 0.0034 amongst different organisms, and an amount of 0.0040 for those using a T-family DNAP. The DNAPs of bacteriophages T2 and T4 are similar to the T7 type studied here. Why did evolution not reduce the number of errors to zero, making the DNAP doing its job correctly all the time? One possible explanation is that bacteriophages (and some other organisms) need to mutate to prevent to be driven to extinct. Bacteriophages must "out mutate" their target host cells and our human capability of inventing new vaccines. A finite mutation rate also makes them capable of contaminating different types of host cells.

7.4 Results

Figure (27) shows the different performance measures as a function of the backward rate k_-. The other rates are set to k_+=500,$k_\downarrow$=1 and $k_\Uparrow$=$e^{\frac{\Delta G}{k_B T}}$. Other parameter settings were chosen as : ΔG=6$k_B T$, L=10^4 (appropriate for the type of bacteriophages considered) and n=4. Tuning k_- tunes the amount of nucleotides a DNAP excises in a single run on average. This is an appropriate parameter reflecting the amount of PIP/error correction.

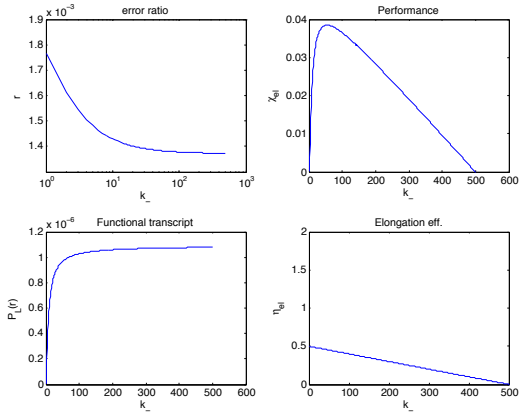


Figure 27: *Top left panel shows the error ratio (semi-logarithmic scale). Bottom left panel shows probability of producing a functional transcript of length L=10^4bp. Bottom right panel shows the elongation efficiency. Top right panel shows the combined efficiency measure. All are plotted as a function of k_- ranging from 0 to 500. Other settings: k_+=500,$k_\downarrow$=1, $k_\Uparrow$=e^6, ΔG=6$k_B T$ and n=4.*

As expected, the error ratio decreases as the amount of PIP increases. Also, the elongation efficiency becomes worse, whilst the probability of producing an error free transcript increases with increasing amount of PIP. Multiplying the last two results in the performance measure, χ_{el}, which has a clear maximum. In figure (28), this efficiency measure is plotted as a function of k_- for different values of ΔG between 6 and $-ln(10^-6)\approx 13.8 k_B T$.

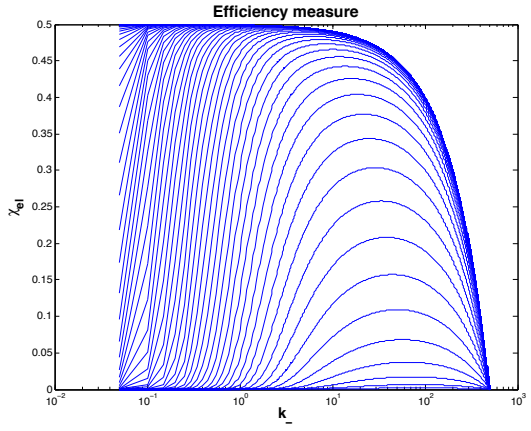


Figure 28: χ_{el} *versus* k_- *for* ΔG *in the ranging from* $6k_B T$ *to* $-ln(10^{-6})k_B T$. *Semilogarithmic scale is used. Higher laying curves correspond to* ΔG *closer to* $-ln(10^{-6})k_B T$.

As the energy difference decreases, higher laying curves, the peak in the curve shifts towards lower k_--values. Also, the peak gets broadened. It seems as if the curves tend to asymptotically grow towards the $\Delta G{=}{-}ln(10^{-6})$ curve. . The following figure shows the average total number of errors in the final transcript as a function of the amount of PIP for a genome length of $L{=}4 \times 10^4$ (Length of genome of T-family bacteriophages, see [6]). In what follows, $n{=}5$.

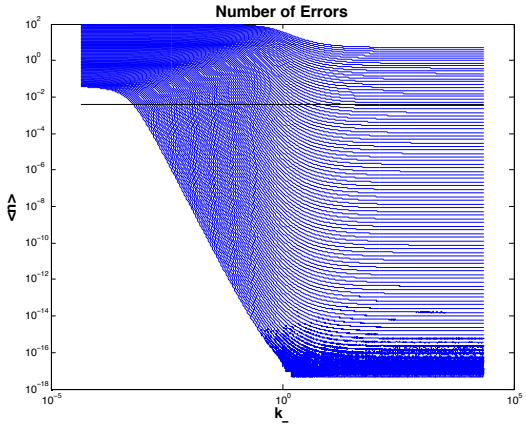


Figure 29: *Average number of errors* $\langle n \rangle = rL$ *for* $L{=}40\ 000$ *and* r *calculated from model. Black line indicates* $\langle n \rangle {=}0.0040$. *The higher the curve, the higher the energy difference* ΔG.

The average number of errors is calculated as $\langle n \rangle {=} r \times L = (1 - P_L(r))L$. Higher laying curves correspond to lower values of ΔG, closer to $6k_BT$. For these curves the total number of errors hardly decreases as the amount of PIP increases. This is because PIP is not only effected by k_-, but also by $k_{\Uparrow}$. As the energy difference decreases, so does $k_{\Uparrow}$. Too low upward rates effectively decrease the amount of PIP by making it less likely to switch into the exo-state. For higher ΔG values a lower k_- is needed to drastically decrease the amount of errors. This is also explained by considering "the total amount of PIP needed" to decrease $\langle n \rangle$. For higher energy values, there is a higher amount of PIP from the onset. Also, the overall positioning of the curves can be explained by the amount of PIP. With more PIP, less errors are present in the final product, as expected. This last feature can also be explained in terms of the error ratio. The error ratio decreases with increasing amount of PIP.
For an error to get corrected, the DNAP should be able to excise enough nucleotides before it shifts back to its polymerase state. The probability of removal increases as it becomes more likely for a DNAP in its exo-state to start cleaving off nucleotides than it is to switch into its polymerization state. All curves seem to stagnate towards the end. This is because $\phi_-(0)/\phi_{\downarrow}(0)$ does not increase significantly any more for such high k_- values compared to $k_{\downarrow}$. The black line in figure (29) indicates the constant $\langle n \rangle {=}0.0040$. The intersection points of the black line with the $\langle n \rangle$ curves were determined. Also, the k_--values that maximize χ_{el}, figure (28), at every ΔG-value were determined. The maximum was taken as the maximum value in the dataset used to plot the curves. The intersection was determined as the value in the dataset that comes closest to 0.0040. Plotting both groups of k_-'s values as a function of ΔG results in the following:

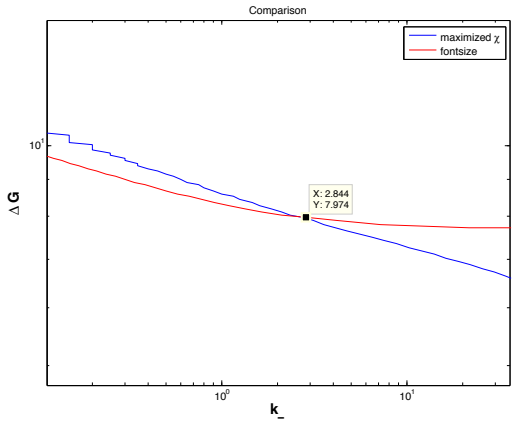


Figure 30: *Backward rates that maximize χ_{el} as a function of ΔG in blue. Backward rates that result in $\langle n \rangle$ equal to 0.0040 as a function of ΔG in red. Both plotted on a double logarithmic scale.*

We see that there actually exists an intersection point. This means that there is an amount of PIP that maximizes the DNAP's elongation efficiency while maintaining an average amount of 0.0040 mutations per cell devision. After determining the intersection point graphically (at $k_{-}\approx$2.8 and $\Delta G\approx$ 7.9 k_BT), a plot of the error ratio is given for the two ΔG values closest to the intersection value:

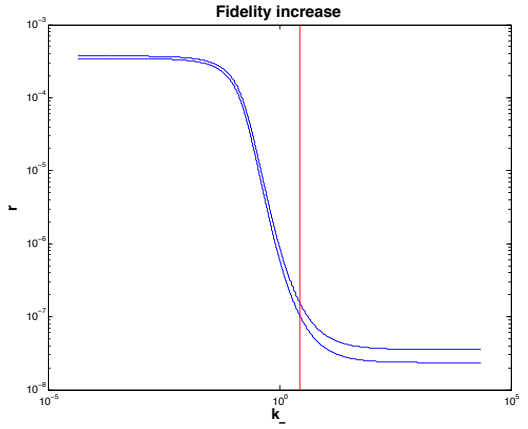


Figure 31: *Error ratio for the two ΔG values closest to the value at the intersection point in figure (30) of $\approx$ 7.9 k_BT. Red line indicates the intersection value for $k_{-}\approx$ 2.8.*

The blue curves show the error ratios for ΔG=7.6578 and ΔG=7.7368 and the red line indicates k_{-}=2.5. We see that the error ratio is not minimized. Possibly, this is favorable such that one obtains a finite amount of mutations per cell devision. This, however, does not explain this fully since the minimum error ratio is not equal to 0. Do organisms somehow know what a sufficient mutation rate is for them to survive? Do organisms want to maintain to possibility for a mutation to drastically increase its fitness, gaining some kind of trait that may be extremely favorable? Allot of unanswered questions remain.
Figures (32)-(35) show the equivalent results obtained by setting n=4 and n=3.

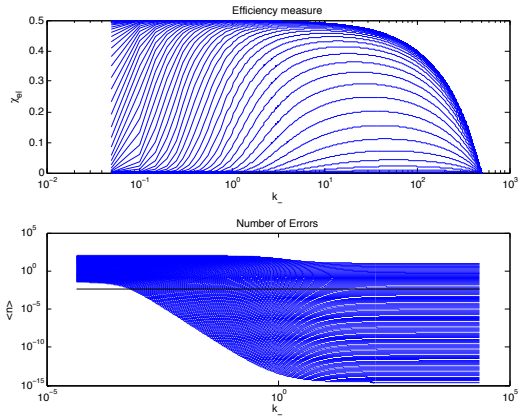


Figure 32: *Results for n=4. Top figure shows performance measure for different values of ΔG and bottom figure shows average numbers of errors in final transcript. Black line indicates $\langle n \rangle$=0.0040.*

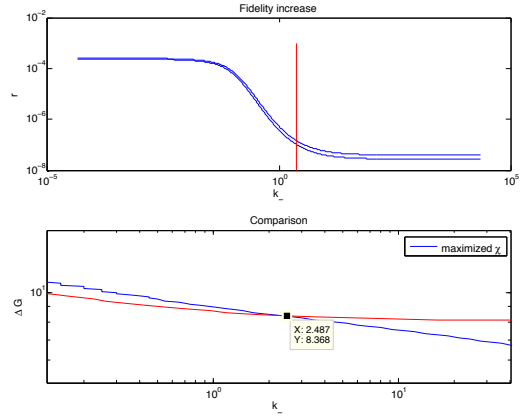


Figure 33: *Results for n=4. The intersection in the bottom figure is approximately at $k_{-} \approx 2.4$ and $\Delta G \approx 8.3 k_B T$. In top figure, error ratios for $\Delta G = 8.2894 k_B T$ and $\Delta G = 8.3683 k_B T$ are shown.*

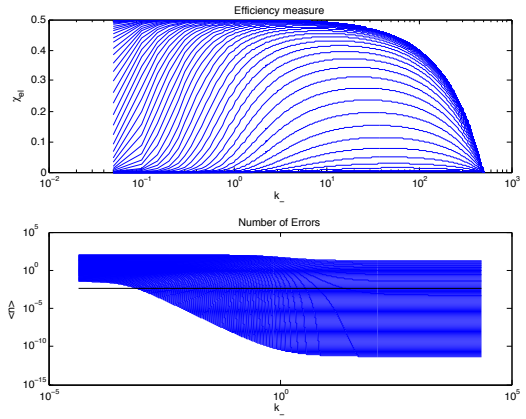


Figure 34: *Results for n=3. Top figure shows performance measure for different values of ΔG and bottom figure shows average numbers of errors in final transcript. Black line indicates $\langle n \rangle$=0.0040.*

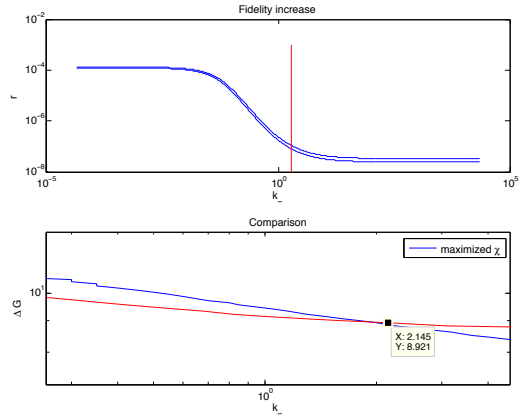


Figure 35: *Results for n=3. The intersection in the bottom figure is approximately at $k_{-} \approx 1.9$ and $\Delta G \approx 8.9 k_B T$. In top figure, error ratios for $\Delta G = 8.9209 k_B T$ and $\Delta G = 8.9999 k_B T$ are shown.*

The methods used for determining intersections and maximum values are not accurate. Unfortunately, there was no time to employ more accurate methods based on the actual functional forms of the curves.

Even so, we see that the mathematical model that was constructed based on the abstract theory of first passage problems holds the potential for explaining both (computer)experimental data and some of natures evolutionary choices regarding DNAP.

References

[1] B.Alberts et al., *Essential cell Biology*. Garland Science, New York (USA) and Londen (UK), 3rd Edition, 2010.

[2] *Optical Tweezers an introduction - http://www.stanford.edu/group/blocklab/Optical%20Tweezers%20Introduction.htm*

[3] *Optical Tweezers - Wikipedia, The Free Encyclopedia.*

[4] M.Depken, J.Parondo and S.W.Grill, *Irregular transcription dynamics for rapid production of high-fidelity transcripts*, arXiv, 2012

[5] T.P.Hoekstra, M.Depken, P.Gross, E.J.G.Peterman and G.J.L.Wuite, *The Cost of Being Right During Replication*, Biophysical Journal 100(3), 2012

[6] J.W.Drake, B.Charlesworth, D.Charlesworth and J.F.Crow, *Rates of Spontaneous Mutation* Genetics, 1998

[7] Y.K Kwok, *Applied Complex Variables for scientists and Engineers*, Cambridge university press, Cambridge (UK), 2nd edition, 2010