

Analyzing Topological Transformation Probabilities of DNA Using Computational Models of CRE Recombinase

STELLA LI¹, JANANI SEKAR², JEFFREY YANG³

¹*Robert Louis Stevenson School*, ²*James Logan High School*, ³*The Harker School*

COSMOS @ UC Davis

August 3, 2018

1 Abstract

This study was undertaken to better understand the enzymatic activity of DNA recombinases, specifically *CRE-lox* recombinase reactions. To model this enzyme activity, a simulation through the topological modeling program KnotPlot™ and the BFACF algorithm were used to study recombination on all knots with up to seven crossings[4]. The data collected from the computational simulations was analyzed to produce a transition probability matrix in order to predict the topological transformations of DNA knots treated with Cre Recombinase. The probability matrix was used to determine the steady state for the recombination reaction and the efficiency of each knot transforming into unknot. The accuracy of each simulation trial was analyzed and compared to experimental data performed on the transformation of PSC1.3i by Cre recombinase. This research has potential pharmaceutical applications that can be furthered to improve the efficiency of enzyme activity in transforming circular DNA chains into the unknotted form.

2 Introduction

This topic was selected as our area of research because of our interest in studying the applications of knot theory and topology in biological systems. Furthermore, the idea that the probability of the different knots that appear after recombination has yet to be modeled using computational methods

motivated us to try and understand this enzyme better. As an intermediate step, recombination is a step in which enzymes may repair damaged DNA or shift the topology of the DNA itself. For this project, we accept the general assumption within the biological community that an unknot is the preferred state for an organism's DNA despite a few possible advantages to knotting [3]. We also included a wet lab portion of our experiment where we looked at the transitions on knot types after Cre reactions with plasmid. Essentially, we reacted the plasmid with Cre twice: once to knot it and once to create a different knot. We hypothesize that Cre activity is not inherently random as the enzyme will attempt to unknot the substrate it binds to. Overall, our goals for this project are to better understand the behavior of Cre enzyme recombinase and compare our model's accuracy to real laboratory data.

3 Methods

All simulations were done using KnotPlotTM, a program that displays projections and manipulates different mathematical knots. KnotPlotTM allows the user to run different algorithms, analyze characteristics, and collect data on various knot types. A KnotPlotTM script randomly transformed a knot into another knot based on a chosen site for recombination. A Java program was then used to generate a probability matrix, a table where both axis have all the knots, and the corresponding box is the probability out of one that the specific knot transition occurs, and MatLabTM was used to produce a visual representation of the data. In the wet lab experiments, Cre recombinase that we used was from New England Bio along with the plasmid DNA.

3.1 KnotPlotTM Simulation

The first step to begin the simulation was to modify a script in KnotPlotTM that recombined all knots between 0_1 and 7_7 which was written by Michelle Flanner, a member of the Arsuaga - Vázquez lab [3]. This script was run for two trials for two versions: the first KnotPlotTM script was written for 15 knots between 0_1 and 7_7, and the second script was written for 27 knots between 0_1 and 7_7*. The second version included the mirrored image knots, making a distinction between the two data tables. For the KnotPlotTM script, each knot was first loaded in as a smooth knot as seen by Figure 1. The knot was then converted to a simple cubic lattice model of a knot, as in Figure 2, so that it can expand and contract for more possible recombination sites to be chosen from. BFACF is a valid algorithm to model recombination because it is proven to explore the entire chain uniformly instead of modeling recombination on a local area of DNA.

Furthermore, BFACF employs the Markov chain algorithm that deforms cubic lattice links for stochastic Monte-Carlo sampling. This depiction with the beads is necessary so that the BFACF algorithm can expand or contract the knot based on the move that is randomly chosen, but it preserves knot type. After BFACF is run for 100,000 times with a z-value that changes the distribution of randomly simulated moves, a random location is chosen where the recombination process will be simulated [3]. These programs, created by members of the Arusaga-Vázquez lab, run in the platform provided by KnotPlot™. For the purpose of this simulation, the program only models sites in inverted repeat, meaning the sites are in opposite directions, ensuring that the result of recombining a knot will be a knot, instead of a link. Then, the knot is contracted, so it can be more easily identified by the built-in ID command in KnotPlot™. Java program was used to count the number of times each knot transformation happens, creating a spreadsheet of the normalized data which does not take into account the unknown knots and the knots that are over 7 crossings. In order to normalize the data, we have to keep uniform parameters for each knot, so any knots that are created over seven crossings will not be considered along with knots unknown by the ID method. We summed the total number of these two types of knots and subtracted it from the total number of trials for each knot. Then, we divided the total number of a knot transition over that value, scaling the probability to the right proportion. The Java Code is in this GitHub repository: <https://github.com/Jyang2602/COSMOSFinalProject>.

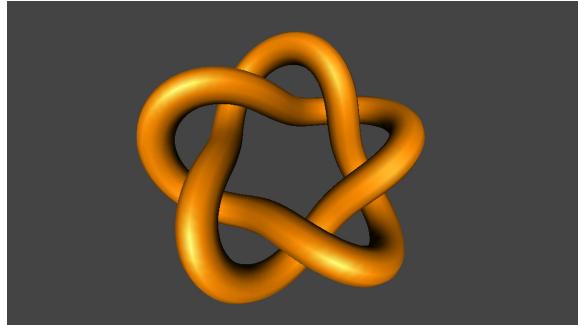


Figure 1: Example of 5_1 from KnotPlot™ [2]

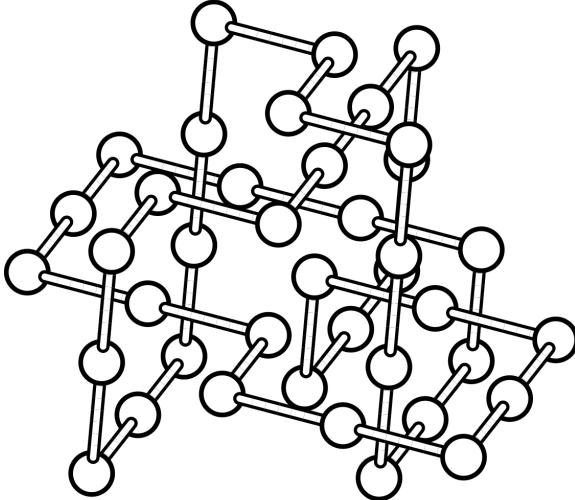


Figure 2: Knot 5_1 in a Simple Cubic Lattice [2]

3.2 Finding Steady State and Verifying Simulation Accuracy

To analyze the data produced from the simulation, a transition probability matrix was created. This was done by organizing the data for every trial in a spreadsheet using the Java program, then using MatLabTM to calculate the eigenvectors and eigenvalues. Because the KnotPlotTM program recombined every input knot, the total of all the decimal probabilities for every knot transition for each input knot would ideally sum to one. To test that the KnotPlotTM simulation worked correctly which was indeed the case, a MatLabTM program was used to calculate the eigenvalues and corresponding eigenvectors for the probability matrix. The first eigenvalue was one, showing that our data was normalized without any outliers, and the total transformation probability for every knot column was one. A steady state for the corresponding eigenvector displayed the values that recombined knots would converge to after multiple repetitions.

3.3 Data Validation Using Chi-Square Test

Four chi-square tests were performed on our data: The first test was done to test the accuracy and reproducibility of the simulation without considering chirality of the trial which consists of 15 data points, one for each knot from 0_1 to 7_7 . The second test was done to test the accuracy and reproducibility of

the simulation considering chirality with 27 data points, one for each knot. The third test was done to evaluate the quality of the approximation of the simulation without considering chirality: the idea that a knot is different from its mirror image. 15 data points were taken from the average steady state vectors of the first simulation; 15 other data points were taken by combining the possibilities of a knot and its chiral counterpart in the second simulation. A fourth test was conducted as a positive control to validate the the test. 27 data points were taken from the average steady state vectors of the first simulation; 27 other data points were obtained by a random uniform distribution probability.

3.4 Improving Simulation by Setting More Parameters

To better simulate Cre activity, the KnotPlotTM script for recombination had restrictions for choosing the location for the process: the site where the knot is cut to be recombined is chosen between one-third the length of the entire knot because that is the accurate tendency for Cre. The code was changed in the parameter for the distance along the arc of the length where the recombination site can be chosen from. Knot length was also set to a certain length using the KnotPlotTM command, bfacf run to length, where x is the length for the knot. Boundaries for recombination were set between a length of 1 and 50 since a BFACF algorithm set every knot's length to 150. The same data excel sheet was created, except with 2,000 trials for each knot. The two data tables were compared to each other to find discrepancies. Refer to figure[14] the appendix.

3.5 Laboratory Experiments on *In Vitro* Cre Reactions

In order to obtain experimental data for comparison to the KnotPlotTM simulations, *in vitro* Cre reactions were conducted in a laboratory environment.

3.6 Testing the Effect of GelRed on Cre Reaction

The first *in vitro* experiment was done to test the effect of GelRed on *Cre-loxP* reaction. This test was necessary because the DNA would be extracted from the stained gel and treated with Cre again. Before incubating the Cre reaction, GelRed was added so that the concentration of GelRed in the tube reached 1X, the same as the concentration in the agarose gel. A control was established by a standard Cre reaction. The control and the Cre+Nicking reaction with GelRed were loaded side-by-side into a gel and electrophoresis was run to analyze the effect of GelRed on Cre recombination. Looking at figure 3, the Cre reaction with GelRed in the tube is in lane 19, the Cre

reaction without GelRed in the tube is in lanes 7 and 20. In lanes 7 and 20, there is band defined a little bit under “1” marker on the right, suggesting the presence of visible knotted DNA there; however, in lane 19, there is no band, suggesting that the knot is more supercoiled due to the GelRed. In the interest of our experiment, we will deal with the assumption that GelRed does not change the topology enough to effect the second Cre reaction.

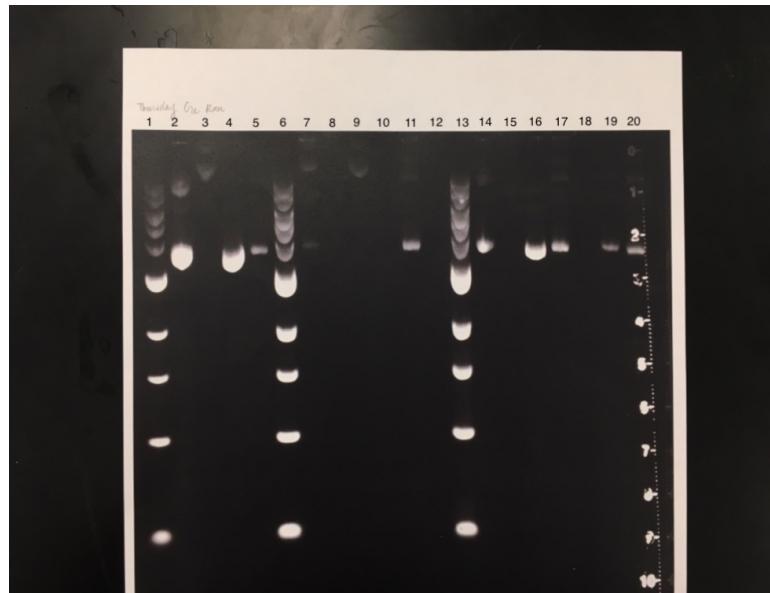


Figure 3: Lab Graph for Cre Proof

3.7 Cre + Nicking reaction on plasmid DNA

3.7.1 Preparing Reaction tubes

In order to nick the products of Cre recombination, we made a total reaction volume of 50 microliters stored in an eppendorf consisting of 40.9 microliters of distilled water, 1 microgram of PCS1.3I plasmid DNA of density 325 micrograms per milliliter, 5 microliters of 10x Cre reaction buffer, and 1 microliter of 15x CRE. Afterwards, we incubated the reaction at 37oC for 60 minutes. Then, we removed 6 microliters from the solution and replaced it with 5 microliters of CutSMart nicking buffer and 1 microliter of Nb.BbvCl enzyme: This step was taken to rid the extra torsion or supercoiling in the DNA. We terminated the reaction by adding 10 microliters of Loading Purple Dye.

3.7.2 Making a Separate Gel

To make sure that some DNA without any GelRed can be isolated, a gel that has GelRed in one half of the gel and no GelRed in the other half was made by placing a ruler in the middle of the gel and pouring gel solutions of the same concentration but one without GelRed. The gel with the ruler in the middle is displayed by figure 4, and the gel is split up so we can control the effect of GelRed on Cre in the experiment. The product of the gel is in figure 5.

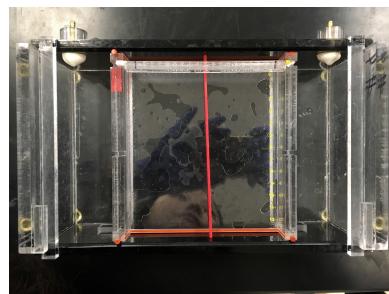


Figure 4: Setup for Gel

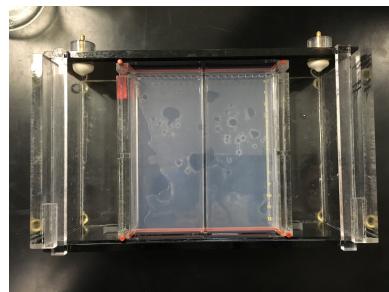


Figure 5: Finished Gel

3.8 Isolate DNA of a Certain Knot Type and Rerun Cre Reaction

We loaded our Cre into this gel separately with the same items on both sides. For the left side of the gel, the one with GelRed, lane 1 was the ladder, lane 2 was plasmid, lane 3 was nicked plasmid, lane 4 was linear plasmid, and lane 5 to lane 9 were nicked cre. The same lane loading was done for the other side of the gel without GelRed. We ran the gel for 20 hours at 40 Volts. Our second gel was run with slightly different procedures. Because not enough DNA showed up, we allowed the Cre reaction to run for 2 hours instead

so the band will be brighter. Then, we identified each knot type based on the distance that the DNA fragments migrate on the half of the gel with GelRed loaded. To quantify the probability More complicated knot types would migrate further from the anode. We cut along straight of both sides of the gel for all bands that we were interested in, and we suspended the DNA in a buffer solution to let it sink over two day. Once that happened, we obtained the DNA of each knot type. Alternatively, a dialysis machine can be used to remove GelRed from DNA solutions. Afterwards, we treated the DNA with ligase to rebuild the phosphodiester bond broken by Nb.BbvCl. In order to look at the probability of knot transitions, we performed another Cre + Nicking reaction on the obtained plasmid DNA, following the same procedure and volumes to prepare the reaction. Run a gel overnight again and look at changes in probability based on light intensity.

3.8.1 Gel Electrophoresis Analysis and Comparison to Simulation

We will quantitatively analyze the percentage of each resulting knot type by examining the brightness of each band after the second Cre reaction. Then, we will compare the lab data with the one found through the computational simulation.

4 Data

To visualize the probability matrices generated by the Java program, probability maps were made for each type of trial using MatLabTM. The following graphs show in order: the average recombination probability of knots in the first two trials, the trials did not separate mirrored knots, the average recombination probability found in the next 2 trials with the mirrored image, and the recombination probability of knots in the final trial which further limited the arc length to model Cre. Each section will have a graph of a heat map of the probabilities with the raw data in the appendix.

4.1 First KnotPlotTM Simulation Combining Chirality

The first trial of the KnotPlotTM simulation did not distinguish between a knot and its mirror image. The following data table was created to show how many times each knot transformed into every other knot. The sum of every column in the table is 500 because each knot was run 500 times. The same process was repeated for one more trial. Refer to image[15] in

appendix for the raw data. This heat map shows the general trends for the normalized probabilities.

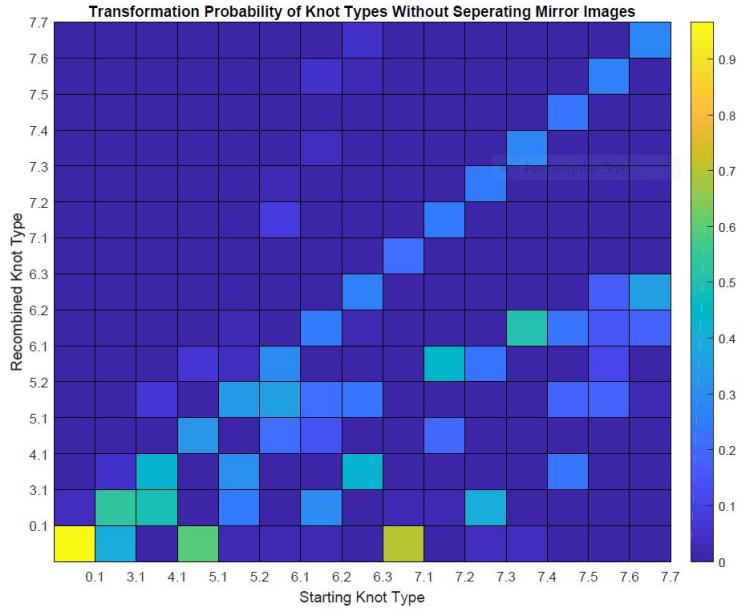


Figure 6: Probability Map for Knot Types without its Mirrored Image

4.2 Second KnotPlot™ Simulation Considering Chirality

The next two trials in KnotPlot™ did distinguish between a knot and its mirror image: in the mirror image of a knot, the orientation of the crossings is the opposite. Knots 0.1, 4.1, and 6.3 are achiral, or can be redrawn to look identical to their mirror image. Thus, these knots were considered the same as their mirror image in our simulation. We wanted to have both graphs and data to test a theory of accuracy if we consider chirality or not. The following heat map was created from the average of the trials with separate mirrored knots. The raw data for the trials with mirrored images can be found in figure [16].

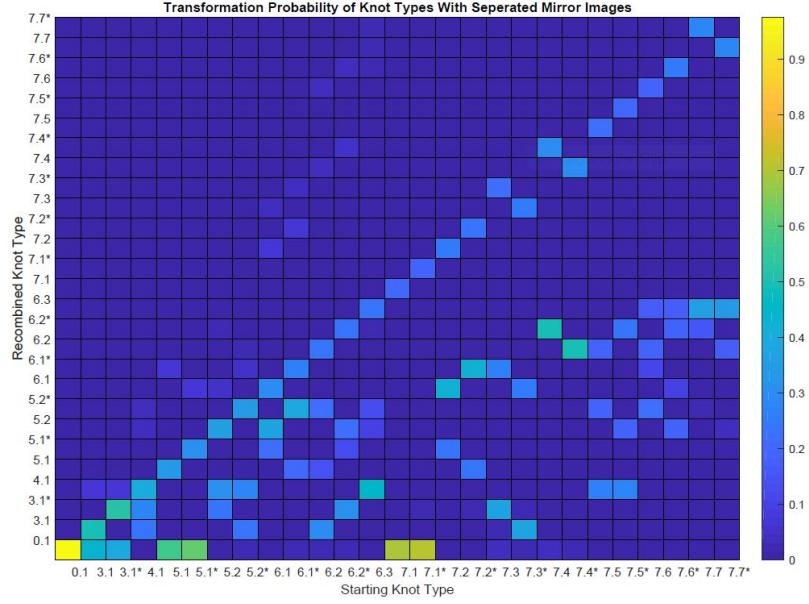


Figure 7: Data for Trial with Mirrored Images

4.3 KnotPlot™ Simulation with More Restrictions

The final trial in this study involved a simulation that limited the arc length between recombination sites to $1/3$ the length of the knot to more accurately model the distance between recombo sites for Cre. The data from this trial and all the previous trials were then normalized using the Java program turned into a square probability matrix. The following heat map shows the data from the normalized probability matrix from the final trial. The raw data for these trials can be found in figure [17].

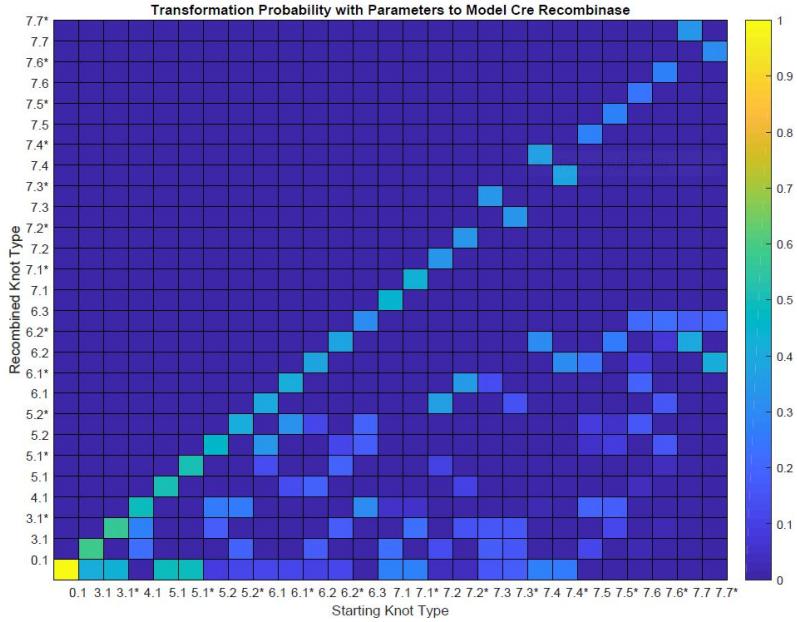


Figure 8: Probability Map for Trial with Cre Boundaries

5 Results and Conclusion

5.1 Data Validation Using Chi-Square Test

Four chi-square tests were performed on our data: The first test was done to test the accuracy and repeatability of the simulation without considering chirality with 15 data points for each trial. The second test was done to test the accuracy and repeatability of the simulation considering chirality with 27 data points, one for each knot. The third test was done to evaluate the quality of the approximation of the simulation without considering chirality: the idea that a knot is different from its mirror image. 15 data points were taken from the average steady state vectors of the first simulation; 15 other data points were taken by combining the possibilities of a knot and its chiral counterpart in the second simulation. A fourth test was conducted as a positive control to validate the the test. 27 data points were taken from the average steady state vectors of the first simulation; 27 other data points were obtained by a random uniform distribution probability.

To determine whether the null hypothesis was accepted or rejected, the χ^2 value for each test was compared to the chi square distribution table with corresponding degrees of freedom. With a degree of freedom of 14, the

χ^2 value of 3.3582 from the trials from the first simulation gave a p-value less than 0.05. Therefore the null hypothesis was accepted, suggesting that $V_1 = V_2$ for this simulation, where V_1 is the steady state vector of trial 1 and V_2 is the steady state for trial 2 of the first set of data.

With a degree of freedom of 26, the χ^2 value of 1.9234 from the trials from the second simulation which consider chirality to be different gave a p-value less than 0.05. Therefore the null hypothesis was accepted, suggesting that $V_1 = V_2$ for this simulation, where V_1 is the steady state vector of trial 1 of simulation 2 and V_2 is the steady state for trial 2 of the second set of data.

With a degree of freedom of 15, the χ^2 value of 10.4782 from third test gave a p-value less than 0.05. Although this χ^2 is considerably higher than the χ^2 obtained from testing two trials of the same simulation, the p-value is still within 0.05. Therefore the null hypothesis was accepted, suggesting that $V_1 = V_2$ for this simulation. In this test, V_1 is the average steady state vector for the simulation that does not consider chirality, and V_2 is the average steady state vector for the simulation that does consider chirality. This test showed that the simulation which did not distinguish chirality is a good approximation of the simulation considering chirality, which further confirmed our hypothesis that chirality does not influence recombination probability.

With a degree of freedom of 26, the χ^2 value of 826.0035 from the positive control gave a p-value greater than 0.05. Therefore the null hypothesis was rejected, suggesting that $V_1 \neq V_2$ for this simulation, and the steady state vector does not conform to a uniform distribution. The results of the positive control test suggested that the chi square test is effective in determining whether two sets of data are precise to a certain degree.

5.2 General Trends in the Data

5.2.1 Common Transitions

Based on our data, efficiency calculations, and analysis of results, we are able to model the general process of recombination. The probability maps that were generated in MatLabTM show a high probability that a knot will recombine to itself. Furthermore, our data suggests that knots are likely to recombine into a unknot, which supports the belief that DNA prefers to be in an unknotted state. Our data also shows that the knots most likely to recombine into a unknot, besides the unknot itself, are knots 3.1, 5.1, and 7.1, which can be found in figure [9]. Once again, the orientation of the knots is important as inverted repeat sites as the knots will purely be transformed into knots and not links.

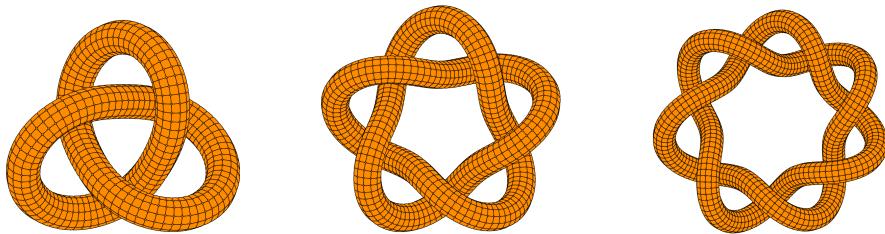


Figure 9: Torus Knots $3_1, 5_1, 7_1$ [2]

These knots are all classified as torus knots because of their shape, which may contribute to why they were able to recombine into an unknot in fewer steps than other knot types. After performing a type 1 tangle (inverted orientation) at any position of the torus knot, the knot transforms into a twisted unknot. Then a Reidemeister I move can be performed on the twisted knot to reach the unknotted state.

5.2.2 Considering Pathway Efficiency

Using the probability matrix from the simulation differentiating chirality, a probability table was computed to represent the probability that one knot transforms into the unknot in a certain number of steps. The probability of a knot transforming to an unknot in one step is given by the probability matrix. To calculate the probability that a certain knot transforms into an unknot in two steps, the row vector of each knot transforming to an unknot in one step was multiplied by the column vector of the knot transforming into each knot. This operation summed up the possibility of a knot transforming into another knot then transforming into an unknot. For $n+1$ steps of recombinations, the row vector of each knot transforming to an unknot in n step was multiplied by the column vector of the knot transforming into each knot. This is shown in figure 10, where all the knot's probabilities of being an unknot after 20 steps is shown.

For the second pathway probability graph, referring to figure 11, the assumption that an unknot is the optimal state for DNA inside the cell was accepted. Therefore, in the transformation pathway model, a knot stays unchanged as an unknot once it reaches the unknotted state. Essentially, we changed the table so that once a knot reaches an unknot, it will always stay there. To match the probability matrix with this assumption, the probability that an unknot converts into an unknot was set to be 1. The calculation to get the matrix and the graph are the same as in the first simulation. This graph

represents the probability that a knot transforms into an unknot within certain steps. Assuming our model stays consistent, at a larger number of steps, all knots should converge to one value.

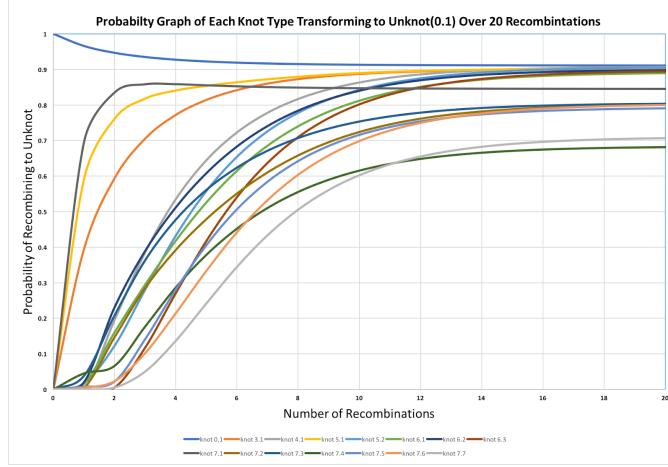


Figure 10: Pathway Probability at 20 steps

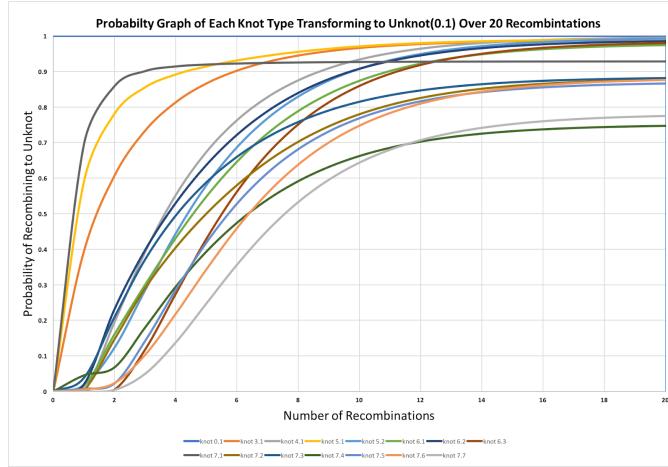


Figure 11: Pathway Probability within 20 steps

5.2.3 Looking at the Data Trial with Restrictions

The final trial in our experiment that limited the arc length between recombination sites to one-third the knot length, which is proportional to the distance between LoxP sites, altered our results. The first major difference in this trial was that the probability of becoming an unknot increased for most knot types, especially the ones with more intersections. More specific-

cally, only six out of twenty-seven knots had a lower probability of turning into an unknot in this trial: 3_1, 4_1, 5_1, 5_1*, 7_1 and 7_1*. These results suggest that if a knot is more complicated, when the recombination sites are restricted to a certain distance, Cre is more likely turn the knot into an unknot. The transformation probability of the simpler knots, however, stayed largely the same. This may be because of the simpler structure of these knots which allow them recombine into an unknot at a high probability already. In addition, knots 3_1, 4_1, and 7_1 are all a similar shape, the twist knot family, so this may contribute to why their probability of turning into an unknot decreases during this simulation. To summarize, referencing the probability map of the data, see figure 8, with the boundaries for Cre, the shades on the map are all slightly darker for the knot transition to itself, shifting the data towards less variance in knot types.

5.3 Wet Lab Results

Looking at the two gel images, there were clear bands at different lengths considering the ladder. From figure 12 and 13, each well from four and on are Cre reactions with DNA. Furthermore, each band represents different knot complexities, though there are many limitations with this experiment. First, we are unable to accurately differentiate which knot each band corresponds to, and to make this model better would require more advanced procedures. One idea would be to manually look at each knot in the microscope and then label the eppendorfs based on manual classification. However, two important conclusions from this experiment is that Cre reacts with DNA topology affected by GelRed and that some complex knots may occur in high probabilities. The bright bands that indicate relatively complex knots, however, are all above the band for the plasmid. This means that the original plasmid DNA is more dense than the resulting knots, indicating that the original plasmid is of a more complex knot type. This agrees with our hypothesis that Cre tends to convert DNA strands into less complex forms. From the two images, each one has a few bands that do not follow the general trend of decreasing probabilities. These bands should be further studied to see which knot type it is, and if this trend continues, perhaps there is a biological advantage to these knots.

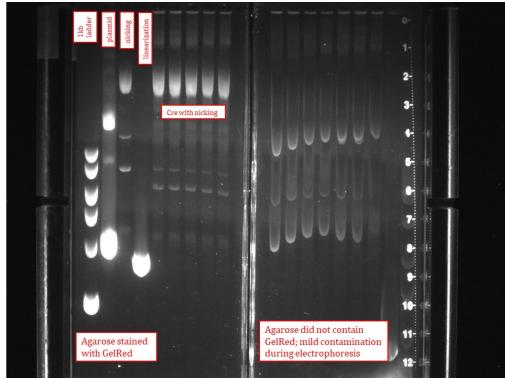


Figure 12: Cre Lab Trial 1

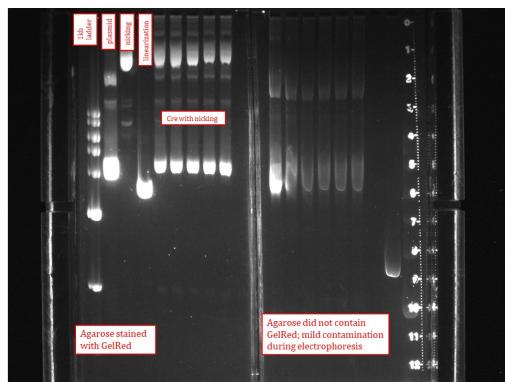


Figure 13: Cre Lab Trial 2

5.4 Sources of Error for Computational Model

The first source of error arises in the number of trials: our first data tables were run with 500 iterations for each knots, and our second data table was run with 2000. Despite the number of iterations, more trials are needed to find a more accurate average of the number of trials. For example, the difference in probabilities between trials is not significant for all knots up to 6 crossings. However, knots above 7 crossings have a significantly greater difference in terms of probability, suggesting a need for more trials. Another error occurs in choosing the z-value of the BFACF algorithm[3]. The z-value represents the probability that the polygon chain knot grows larger or smaller with each step. Each individual knot has its own z-value which should allow the knot to stabilize at a length of 200. Due to the time constraint, we were unable to determine the z-value manually for each knot, and thus, we made the assumption that its effect is minimal as we included

a command to always run the knot to a certain length before applying recombination. This assumption, however, may have skewed the data in a certain direction. Round-off error is an issue as well because the eigenvector for the column corresponding to eigenvalue one does not add up to 1. This inconsistency is most likely attributed to double float point rounding issues in the design of Java.

5.5 Sources of Error for Lab

The most pertinent error in our lab was the fact that DNA bands still appeared on the gel without GelRed. Without the stain, the DNA should not have appeared under the UV light, but this fact may also be attributed to the purple loading dye added to stop the reaction. Our initial plan was to use the stained gel as a marker for where to cut the unstained gel, but since both gels had DNA bands under the UV light, we suspended both the DNA in solution. This, however, means that our DNA will be affected by GelRed for the second Cre reaction. Even though the Cre reaction worked with the DNA affected by GelRed, figure 3 shows that the GelRed band is still slightly lower. A second reaction may magnify this difference.

6 Future Research

Further research for this project entails first and foremost generating more experimental data. By scaling up the lab experiments in the gel, more DNA can be extracted so that more data points can be generated. Our experiment only includes one gel, so there is not enough data to be compared to our probability map. With more advanced programs, we could run more trials where the script stops once the knot is transformed an unknot. This method would be more accurate in determining the efficiency of enzymes on turning knots into unknots. Changing the recombo range could also be interesting to see how the data is affected by how size of the range of the recombo site. If the arc length where the recombination site can be chosen from increases, interesting further research could be to mathematically model the steady state value of the knot becoming an unknot. Our hypothesis for this sub-topic would be that as the arc length increases, the chance of an unknot in the steady state decreases. Clearly, our data changed quite a bit when we limited the range between 1 and 50. We would presume that limiting the range even more would cause the knot to turn to itself or an unknot even more often. Looking at the accuracy of our model, we could look into the Dr. Arena's paper on varying accuracy of DNA models: Trying out different applications to model DNA with Cre activity will allow us to narrow down

our errors as well[1]. Furthermore, we dismissed the fact that CRE only acts on negative writhe, and a next step in an accurate simulation would be to include that. Lastly, this research only considers inverted repeat DNA sequences for enzyme binding sites, ensuring that a recombinase will turn a not into a knot. A more comprehensive model would also look into direct repeat DNA sequences for enzymes that would create links between DNA strands instead of only knots.

7 Acknowledgements

A huge thank you to all of the people who supported us throughout this project. From our peers to our TA's and all of our professors. However, to those outside of the COSMOS circle who helped out, such as Michelle Flanner, your help is greatly appreciated as well. To especially Ali Heydari, Keith Fraga, Professor Vázquez, and Professor Arsuaga, your support day in and day out has made this research paper and program in general such a wonderful educational opportunity for us.

8 Appendix

```

4 alias configurereconnection "load $0; embed clk; bfacf load;
5 bfacf recombo distance $1 $2; bfacf recombo direct $3;" 
6 alias warmup "bfacf prob 0 0 1; bfacf step 100000; bfacf z
.204; bfacf step 100000;" 
7 alias configuregaussanddata "gauss open $0; gauss noblank yes;
data open $0.dat;" 
8 alias datathengauss "leng; bfacf prob 1 1 0; bfacf step 100000;
ago 200; centre; align axes; id; gauss; data;" 
9 alias performreconnection "bfacf recombo" 
10 alias closegaussanddata "gauss close; data close;" 
11 silent=t; 
12 configuregaussanddata 7_7*SimulatingCre; data format "/i, /l"; 
13 
14 frame command "configurereconnection 7_7* 1 50 off; warmup;
bfacf runtolength 150; performreconnection; datathengauss;" 
15 frame 1 to 2000; 
16 
17 closegaussanddata;
```

Figure 14: KnotPlotTM Script [2]

	0.1	3.1	4.1	5.1	5.2	6.1	6.2	6.3	7.1	7.2	7.3	7.4	7.5	7.6	7.7
0.1	488	196	0	316	6	7	10	1	353	3	17	25	4	4	0
3.1	11	261	240	0	133	0	132	0	9	15	196	0	4	5	0
4.1	0	26	219	0	159	0	0	214	0	0	0	0	108	0	0
5.1	0	0	0	135	0	118	70	0	0	99	0	0	0	2	0
5.2	0	7	34	0	162	172	112	118	0	0	0	0	94	94	0
6.1	0	0	0	33	21	149	0	0	0	236	115	0	0	58	0
6.2	0	1	0	6	9	0	126	9	0	0	0	250	122	75	96
6.3	0	0	3	0	1	0	0	124	0	0	0	0	0	85	17
7.1	0	0	0	0	0	0	0	0	101	0	0	0	0	0	0
7.2	0	0	0	6	0	34	0	0	0	102	0	0	0	0	0
7.3	0	0	0	0	0	0	10	0	0	0	113	0	0	0	0
7.4	0	0	0	0	0	0	15	0	0	0	0	107	0	0	0
7.5	0	0	1	0	1	0	1	0	0	0	0	0	96	0	0
7.6	0	0	0	0	3	0	27	6	0	0	0	0	0	115	0
7.7	0	0	0	0	0	0	1	19	0	0	0	0	0	0	10
unknown knots	0	1	0	1	2	4	4	4	34	39	40	49	41	37	4
B* crossings	1	8	3	3	3	6	2	5	3	6	19	69	31	25	7

Figure 15: Data for Trial with No Mirrored Images

	0.1*	3.1*	4.1*	5.1*	5.5*	5.6*	5.2*	6.1*	6.1*	6.2*	6.3*	6.3	7.1*	7.1	7.2*	7.2	7.3*	7.3	7.4*	7.4	7.5*	7.5	7.6*	7.6	7.7*	7.7	
0.1	490	210	193	0	277	314	4	11	43	13	11	0	345	341	3	18	12	8	6	0	0	0	0	0	0	0	
5.5	248	0	114	0	0	0	123	0	0	144	0	0	8	4	0	10	192	0	0	5	0	1	0	0	0		
5.1*	4	0	261	133	0	0	111	0	0	0	162	0	0	13	0	9	174	16	0	0	0	5	0	4	0		
4.1*	0	0	0	0	0	0	146	147	0	0	0	0	233	0	0	0	0	0	0	0	0	0	0	0	0	0	
5.1	0	0	0	0	0	0	175	0	0	0	116	69	0	0	0	0	115	0	0	0	0	0	0	7	0	0	
5.1*	0	0	0	0	0	0	148	0	0	93	0	62	0	0	0	151	0	0	0	0	0	0	0	4	0	0	
5.2*	0	0	0	0	0	0	178	0	0	196	0	93	0	49	0	0	0	0	0	0	0	0	0	0	0	0	
5.2*	0	0	0	0	0	0	18	0	0	0	156	187	112	0	59	0	0	0	0	0	0	0	91	56	96	0	
6.1	0	0	0	0	0	0	31	0	0	154	0	0	0	220	0	0	124	0	0	0	0	0	0	44	0	0	
6.1*	0	0	0	0	0	0	37	0	0	21	0	122	0	0	0	0	202	132	0	0	0	0	0	0	53	0	0
6.2	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	243	0	0	0	0	0	0	0	0	0
6.2*	0	0	0	0	0	0	1	0	0	0	0	114	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6.5	0	0	0	0	0	0	5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	85	95	160	160	
7.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7.1*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7.2	0	0	0	0	0	0	0	0	0	32	0	0	0	0	0	0	106	0	0	0	0	0	0	0	0	0	
7.2*	0	0	0	0	0	0	0	0	0	31	0	0	0	0	0	0	99	0	0	0	0	0	0	0	0	0	
7.3	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	105	0	0	0	0	0	0	0	0	
7.3*	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	108	0	0	0	0	0	0	0	0	
7.4	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7.4*	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	103	0	0	0	0	0	0	0	0	
7.5	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	104	0	0	0	0	0	
7.5*	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	84	0	0	0	0	
7.5	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	
7.6*	0	0	0	0	0	0	0	0	0	1	0	0	13	14	0	0	0	0	0	0	0	0	0	0	0	0	
7.7	0	0	0	0	0	0	0	0	0	0	0	0	1	0	9	0	0	0	0	0	0	0	0	0	0	0	
7.7*	0	0	0	0	0	0	0	0	0	0	0	0	0	1	7	0	0	0	0	0	0	0	0	0	0	0	
Unknowns	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B* crossing	1	4	5	4	1	1	4	27	7	20	6	4	7	7	14	8	23	21	11	78	79	35	0	6	93	88	

Figure 16: Data for Trial with Mirrored Images

Figure 17: Data for Trial with Cre Boundaries

```

1  for i=0:999
2  pos1=datasample(1:30,15,'Replace',false) % Generate 15 random positions out of 30
3  X=AllData(pos1) % Randomly select 15 values out of the 30 elements in 2 eigen vectors
4  Thirty=[1:30]
5  Thirty(pos1)=[] % Produce the rest of the 15 position numbers
6  pos2=Thirty(randperm(length(Thirty))) % Randomly order the 15 remaining positions
7  Y=AllData(pos2) % Generate the other 15 elements in the vector
8  Dr=X-Y % Find the difference between each corresponding elements
9  Ti=bsxfun(@rdivide,Dr.^2,Y(:))
10 Chi2=sum(Ti) % Find the chi square value of the randomly generated vectors
11 filenames=sprintf('%s_%d','chisquare',i)
12 save(filename,'Chi2','-ascii') % Save the chi square value in a text editor file
13 i=i+1
14 end
15
16 % Using Terminal to compile all 1000 chi square values into one txt file
17
18 D=NoStar-Star % Find the difference between each pair of elements in the vectors
19 T=bsxfun(@rdivide,D.^2,Star(:))
20 chisquare=sum(T) % Calculate the chi square value of the two eigenvectors of our data
21
22 k=0
23 for i=0:999
24 a=randsample(e02, 1)
25 achisquare
26 if ans==1
27 k=k+1
28 else
29 k=k
30 end
31 end
32 P = k./i % Calculate the probability that the chi square value of our data is
            smaller than the randomly generated chi square values in the chi square distribution.

```

Figure 18: Script to Calculate Chi Value

References

1. Miguel Arenas. Computer programs and methodologies for the simulation of dna sequence data with recombination. *Frontiers in genetics*, 4:9, 2013.
2. Robert Scharein. Knotplot, 2018.
3. Robert Stoltz, Masaaki Yoshida, Reuben Brasher, Michelle Flanner, Kai Ishihara, David J Sherratt, Koya Shimokawa, and Mariel Vazquez. Pathways of dna unlinking: A story of stepwise simplification. *Scientific reports*, 7(1):12420, 2017.
4. EJ Janse Van Rensburg and SG Whittington. The bfaf algorithm and knotted polygons. *Journal of Physics A: Mathematical and General*, 24(23):5553, 1991.