Toward Entropic Trapping of DNA near Solid-State Nanopores

Lucas Eggers

Brown University Department of Physics

(Dated: May 2, 2013)

Abstract

In this study we investigated the viability of entropically trapping DNA near solid-state nanopores. We used modified nanopore structures in which we etched 400nm tall cavities adjacent to the nanopores in the same silicon wafer. Our preliminary attempts to entropically trap molecules failed due to flawed methodology and too-short cavities. To better understand trapping, we performed a theoretical calculation of minimum cavity height which yielded approximately 450nm. We also advanced other nanopore experiments (more specifically, experiments involving asymmetrical pore translocations and molecular "ping-pong") by writing data analysis code in MatLab. The analysis code for ping-pong experiments required developing a new heuristic for translocation detection which will prove useful in writing data analysis code for trapping and FPGA code to improve ping-pong experiments.

Contents

INTRODUCTION

Better nanofluidic control over DNA will yield exciting leaps in technology, such as biochemical labs-on-a-chip and so-called DNA hard drives. We intend to get a few steps closer by making an isolation chamber to hold single DNA molecules in place. Such an isolation chamber would allow for the outcome of biochemical experiments to be observed on a per-molecule basis. It could also act as a storage medium for data encoded in DNA which can hold the equivalent of one million CDs in a single gram for 10,000 years[1]. We believe we can create such an isolation chamber based on entropic trapping, and combine it with a molecular detector called a nanopore.

In order to isolate a single DNA molecule in the chamber, we need a way to deliver it and to know when it enters. Nanopores are the ideal device for such purposes. A nanopore is a small hole in a membrane, be it biological or synthetic. Solid-state nanopores consist of a hole approximately 10 nanometers across in some larger solid-state membrane. For contrast, double-helical DNA is approximately 2.5 nanometers across. When placed between two reservoirs of ionic solution, solid-state nanopores can detect the passage of a single DNA molecule. This passage event is also known as a translocation. When a voltage bias is placed across the two reservoirs, current forms from ions flowing through the pore. DNA, driven by electrophoretic forces on its slight negative charge, travels toward the pore, eventually getting sucked through. Its passage displaces some flowing ions, causing an observable drop in the ionic current through the pore.

DNA can assume many shapes while passing through the pore, the most common of which are represented in Fig. 1. The most important feature of the current blockage is that the amount of current blocked is linearly proportional to the number of strands of DNA in the pore. That is, a molecule folded once blocks twice as much current as an unfolded molecule. This linear relationship allows us to define a conserved quantity: Event Charge Deficit, or ECD. ECD is the total amount of charge blocked by the molecule during its translocation, or, the area under the curve. It has been experimentally shown that ECD is a constant for identical molecules and driving voltages, making it useful for studying translocation dynamics.

In order to trap DNA in its isolation chamber, we intend to take advantage of the DNA's configurational entropy. At equilibrium, DNA forms a random coil whose spherical shape

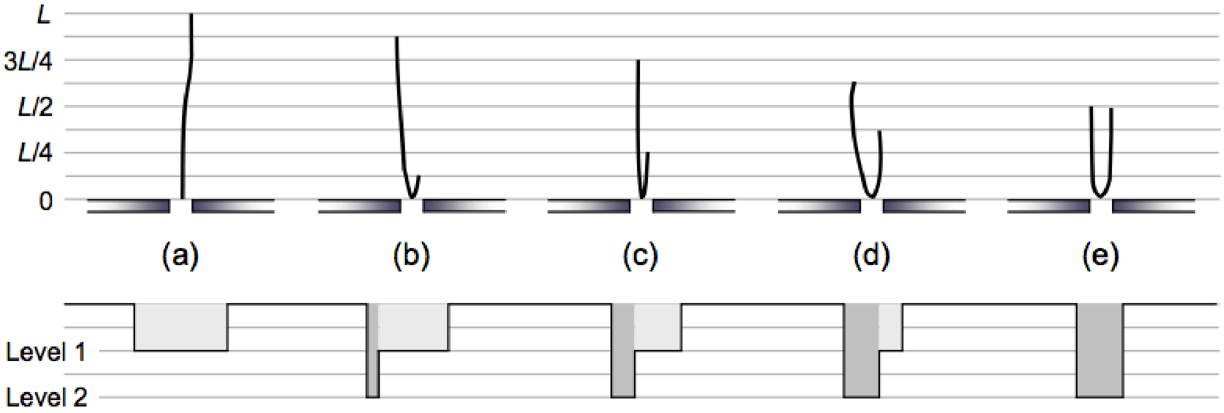


FIG. 1: A schematic of how the stacked-top-hat shape of current vs time data depends on how DNA enters the nanopore. Missing are schematics of multiply-folded molecules but, the linear relationship between current blockage and strands in the pore is made clear. Almost every thin spike in 7 is of one of these forms. (Taken from Nick Hagerty's thesis.)

has a size characterized by the radius of gyration (R_g). If we could get DNA in a cavity that is roughly the size of the radius of gyration and that has holes smaller than the radius of gyration on either side, we predict that the molecule will have a near-zero probability of diffusing out; it would have to squeeze too far out of equilibrium to exit this entropic trap.[2]

The device we envisioned to trap individual DNA molecules is shown in Fig. 2. (Note that the molecule pictured is not trapped.) Capturing a DNA molecule in the cavity can be thought of as a competition between two timescales: the time required to push the center of mass of the DNA through the structure and the time it takes the DNA molecule to equilibrate in the cavity so it cannot traverse the tunnel. A translocation starts when the leading tip of DNA - or someplace nearby on the polymer - enters the nanopore. A drop in current is observed. The molecule is driven through the cavity by local electric fields. As it is driven, Brownian motion causes the molecule to start to equilibrate and expand to fill the chamber. However, translocation happens much faster than equilibration, so the molecule remains "skinny" when compared to the tunnel; continuing to push it will cause it to exit the structure. Thus, the probability of exit is dependent on the cavity length and hole diameter. When the molecule finishes translocating the current will return to its original baseline value. If the cavity is large enough, turning off the driving voltage at the right moment will allow the molecule to equilibrate inside the cavity, trapping it.

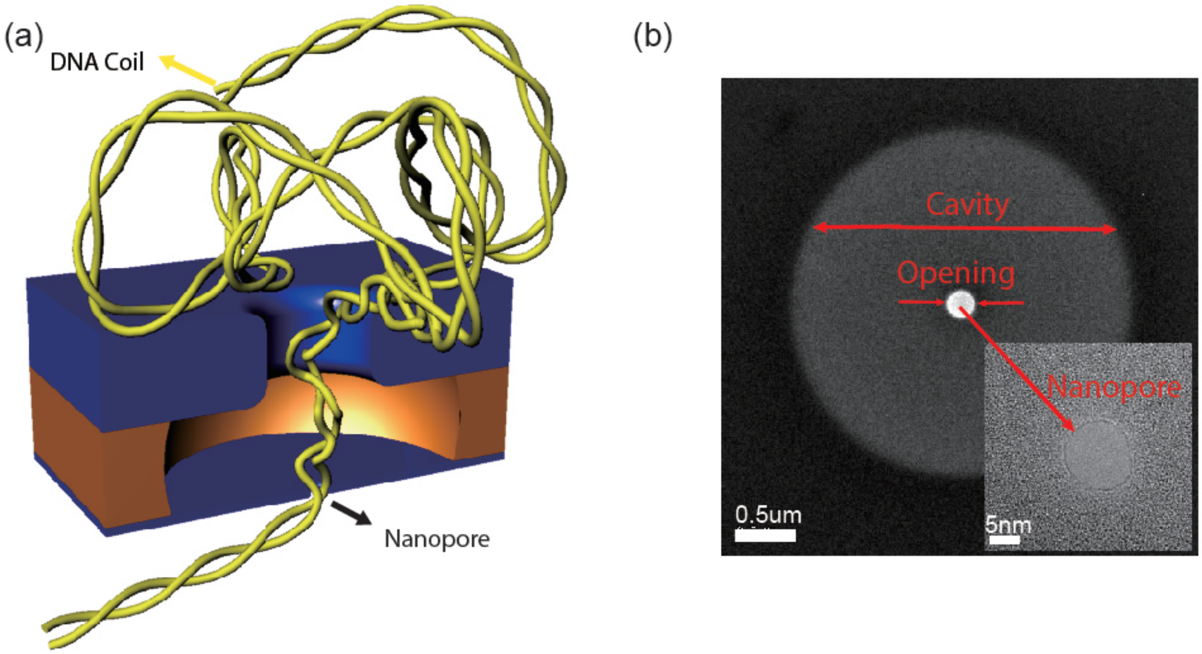


FIG. 2: (a) A schematic of a silicon structure containing a nanopore and adjacent chamber during a DNA translocation event. The structure has 3 main layers. From bottom to top they are the nanopore, the cavity, and a tunnel with diameter 500nm on average. (b) A photo taken with an electron microscope of one of our pores. The layers are so thin that they are not completely opaque.

Placing such a structure between two reservoirs of ionic solution yields the translocation dynamics described above with a twist: the structure's asymmetry means translocations from either side of the pore are not equivalent. Although the effects of the asymmetric structure on translocation dynamics will be touched on, Karri DiPetrillo's thesis takes a deeper look at those phenomena. The focus of this thesis will be trapping DNA molecules approaching the exposed nanopore (the bottom layer in Fig. 2). Our objectives are thus twofold: to develop the hardware (pores-plus-chambers) and the software (control electronics and data analysis) to make our vision a reality. My contributions were software for analysis and recommendations for programming control electronics informed by my preliminary experiments and theoretical calculations. In addition to creating more intricate labs-on-a-chip as in Fig. 3 and possibly storing DNA hard drives, creating such entropic traps would yield deeper understanding of DNA as a polymer more generally.

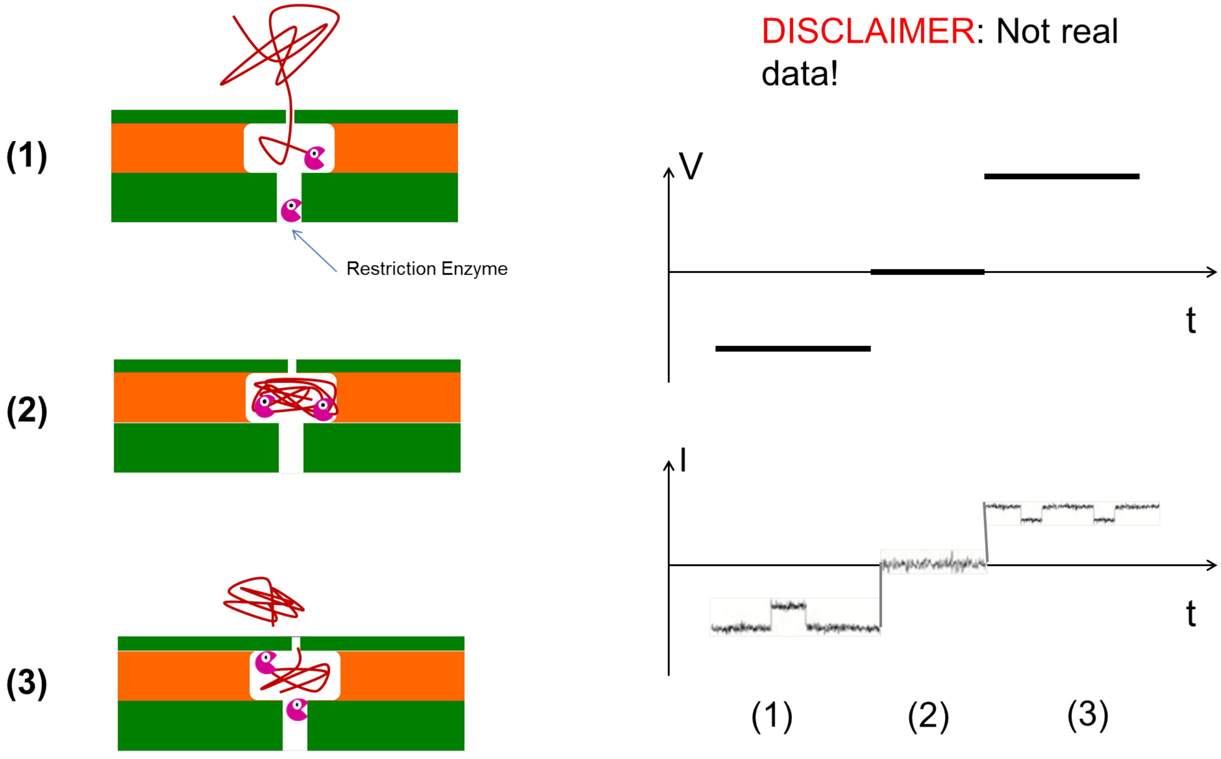


FIG. 3: An schematic of the envisioned nanopore device in which an entropic trap would effectively be a nanoscale test tube. We should be able to trap a DNA molecule by turning off voltage at the proper time (1), perform some biochemistry on it (such as segmenting it using a restriction enzyme) (2), and test whether our chemistry proceeded as planned by using the detection properties of the nanopore (3).

Attributions

Because Karri DiPetrillo and I were working on different uses of the same experimental setup, our work overlapped. We started by working together toward the common goal of entropic trapping of DNA but diverged when preliminary experiments failed. Karri worked toward understanding our pores' geometry's effects on translocation dynamics; I focused on making sense of the data produced by experiments in our lab and better understanding why trapping did not occur. Due to the connectedness of our work we collaborated on the Background and Experiments sections to determine the best information to include. However, our words are our own.

THEORY

Statistical Properties of DNA Polymers

We wish to trap individual DNA molecules in a micrometer-scale cavity next to a nanopore. In this section, we consider the statistical and dynamical properties of long polymers that will inform our success. We know the molecule can be entropically trapped when the radius of gyration R_g is greater than the constraining length, in our case the tunnel opposite the nanopore. The relaxed, randomly coiled molecule at equilibrium would have to squeeze itself in an energetically unfavorable way to fit through the tunnel.[2] The molecule equilibrates to R_g in a characteristic time we also need to know to asses the feasibility of trapping; R_g does not matter if the molecule has exited the structure. We used linearized λ-DNA in our experiments so I will use its properties to get values in units like meters and seconds from equations.

The simplest models of polymer statistics at equilibrium consider only two properties of the molecule: its total contour length L and persistence length l_p which is normally assumed to be small, i.e. $l_p \ll L$. These two numbers describe a freely-jointed chain of length L in which rigid lengths of polymer of size l_p are connected by joints that can take any bond angle. l_p can be thought of as a measure of the stiffness of the molecule. The resulting shape of such an object is, on average, a sphere. Mathematically the chain can be described by a three-dimensional random walk. As such, one would expect the radius of the sphere, also known as the radius of gyration, to scale with the square root of the length of the chain: $R_g \propto L^{1/2}$. The flaw in such reasoning is that real polymers are self-avoiding, meaning that links in the chain cannot occupy the same space. The Flory model modifies the random walk by using a mean field approach (where segments encounter one another with equal probability) to define a new parameter, the Flory exponent ν_F, where $R_g \propto L^{\nu_F}$. The Flory model predicts that, for self-avoiding DNA, $\nu_F = \frac{3}{5}$. Simulations and experiments yield an exponent of approximately .588.[3] Numerical values for λ-DNA are summarized in Table I.

Property	Symbol	Value
Number of base pairs	N_{bp}	48.5kbp
Contour Length	L	$\approx 19\mu$m
Persistence length	l_p	53nm
Radius of gyration	R_g	0.73 μm

TABLE I: Statistical properties of DNA as summarized by Dorfman. [4]

The Rouse model takes the Flory model and makes it dynamic by taking into account Brownian motion. The Rouse model describes a polymer as a collection of beads connected by springs in which beads feel the effects of thermal forces and drags. It does not, however, take into account hydrodynamic interactions. Zimm added the missing hydrodynamic interactions between different parts of the chain as mediated by the solvent to create his model. The Zimm model's predictions most exactly agree with diffusion relaxation experiments. The Zimm model predicts that a polymer enters its equilibrium state from any other state within a characteristic relaxation time[5]

$$\tau_z = \eta(\sqrt{N}l_0)^3/\sqrt{3\pi}k_BT \approx 170\text{ms}. \tag{1}$$

where $\eta = 1$m·Pa·s is viscosity, That is, if one stretched a DNA molecule to be completely straight, it would reach a spherical conformation in τ_z seconds, maximum. The process of reaching such a sphere is known as relaxation. Our translocations last roughly 2 ms, so molecules do not have time to equilibrate until long after they translate. Thus our translocations are considered fast translocations. To paint a physical picture of what that means, we imagine a randomly coiled rope being pulled off a table. As it is pulled, individual folds are sequentially straightened and pulled off the table. Only the most recently straightened fold (a small segment known as the moving length) is involved in the motion. The rest of the rope is stationary. The same things happens with DNA, but at approximately 10^{-7} of the rope's length scale. Folds of the molecule are sequentially straightened as they are sucked through the pore while the rest of the molecule is effectively frozen.

The proceeding paragraphs assume that the DNA molecule is directly next to the nanopore. With our pore-plus-cavity structures this is not always the case; DNA approaching from the tunnel side will encounter a barrier before reaching the pore. We believe that a long strand of DNA approaches the nanopore through the tunnel and cavity while the rest of the bundled molecule is held outside the chamber, causing greater fluid drag and thus slower translocations. The theory for how exactly the tunnel impacts translocation dynamics is explored in much greater depth Karri DePetrillo's thesis. This same barrier serves a dual purpose: to trap the DNA molecule.

Trapping in More Detail

Trapping can be thought of as a competition between the movement of DNA through the pore-plus-cavity structure and the equilibration of the molecule in the cavity. The major force pushing the DNA molecule though the structure is electrophoretic. From charge conservation we know that, to a good approximation, the current traveling through the pore (our measured quantity) is equal to the current passing through any hemisphere enclosing the pore:

$$I = 2\pi R^2 J(R) \tag{2}$$

where I is current, R is distance from pore, and $J(R)$ is the current density. Solving Eq. 2 for $J(R)$ we see

$$J(R) = \frac{I}{2\pi R^2}.$$

By Ohm's Law we know

$$\overrightarrow{J}(R) \equiv \sigma \overrightarrow{E}(R),$$

where σ is conductivity. Combining these equations and solving for E yields:

$$E(R) = \frac{I}{\sigma 2\pi R^2} \tag{3}$$

We know the velocity of the DNA molecule in an electric field:

$$v_{\mathrm{DNA}} = \mu_{\mathrm{DNA}} E(R) \tag{4}$$

where μ_{DNA} is the electrophoretic mobility of DNA. Combining Eqs. 3 and 4 we find the velocity of a DNA molecule a distance R from the pore:

$$v_{\mathrm{DNA}} = \frac{\mathrm{d}R}{\mathrm{d}t} = \mu_{\mathrm{DNA}} \frac{I}{\sigma 2\pi R^2}. \tag{5}$$

Integrating yields the tip's distance from the pore and the elapsed time:

$$\int_{R=0}^{R(\Delta t)} \frac{\sigma 2\pi R^2}{\mu_{\mathrm{DNA}} I} \mathrm{d}R = \int_{t=0}^{\Delta t} \mathrm{d}t \tag{6}$$

yields an equation

$$\frac{1}{3}\frac{\sigma 2\pi R^3}{\mu_{DNA} I} = \Delta t, \tag{7}$$

which can be solved for $R(\Delta t)$:

$$R(\Delta t) = \sqrt[3]{\frac{3\mu_{DNA} I}{\sigma 2\pi}\Delta t}. \tag{8}$$

Eq. 8 describes the location of the tip as a function of time. If $R(\Delta t)$ is less than the height of the cavity, the molecule should be trapped. Even if the tip could travel outside of the chamber (that is, $R(\Delta t)$ greater than cavity height), we believe the molecule could still be trapped with high probability if the center of mass is within the cavity. The center of mass, the point from which R_g is measured, should be at approximately $R(\Delta t)/2$.

Now we will find a numerical value for cavity size. Plugging in numbers from our experiments, $\mu_{DNA} = 3.75 \times 10^{-7}\text{m}^2/\text{V}\cdot\text{s}$ [6] $I = 2 \times 10^{-8}\text{A}, \Delta t = 2 \times 10^{-3}\text{s}$ and $\sigma = 10$ S/m[7], we see that the molecule travels

$$R(\Delta t) = 895\text{nm}$$

in the time it takes to translocate. Half of this (that is, 450nm) gives us a lower bound for cavity length. As an upper bound we can use the height of a fully relaxed DNA molecule, or $2 \cdot R_g \approx 1.5\mu m$. We now know what trapping takes and to conceptualize the problem as a race. The next section will discuss how we probed the theory.

EXPERIMENT

Synthesis of Nanopores

We fabricated our nanopore-plus-cavity structures to entropically trap DNA molecules using methods similar to Zandbergen et al.[8] We started with large wafers with the layers seen in Fig. 4 manufactured by the Cornell Nanofabrication Facility. The top layers of the wafer were present due to constraints of the manufacturing process and useless to us. We used two types of etching to burn through them to gain access to layers in which we made more interesting structures: thermal plasma etching to remove the top silicon nitride layer followed by potassium hydroxide base to remove the pure silicon. We then used a Focused Ion Beam (FIB) to create holes in the 400nm middle layer of SiN with diameters ranging from 150 to 900 nanometers. This diameter is the constraining feature of the setup that

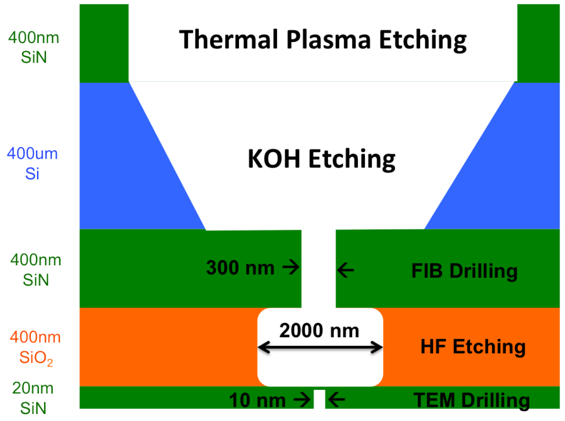


FIG. 4: A more specific schematic of our structure including methods of feature creation. Note: Not to scale.

entropically traps DNA and increases the moving length of the molecule in translocation experiments. The FIB hole allowed access to the silicon oxide layer which was etched using hydrofluoric acid, creating a chamber 1-2μm in diameter and 400nm tall in the our experiments. Recently ordered wafers have a taller layer of silicon oxide, enabling us to make taller chambers. Last, but not least, we created the nanopore in the bottom layer of silicon nitride roughly 10nm in diameter and 20nm tall using a Transition Electron Microscope. This novel structure is ideal for entropic trapping because it places the cavity that will trap the molecule directly adjacent to the molecular detector. Thus we can necessarily know when a candidate for trapping enters the chamber.

Apparatus and Setup

Nanopores are excellent single-biomolecule detectors. As such, we needed to take steps to ensure that the only biomolecules present in our experiments were DNA. To remove all foreign contaminants (things like skin cells, dust, possibly hair) we cleaned our nanopores in Nano-Strip (Cyantek Corporation, 90% sulfuric acid, 5% peroxymonosulfuric acid, 5% water) at 75°C for 2.5 hours. Nano-Strip also made our pores hydrophilic which helped reduce noise in our data. Pores were then rinsed with deionized water and placed in a custom-made PVC chuck pictured in Fig. 5. We rinsed the setup by pushing degassed, deionized water and isopropyl alcohol through the fluid inlets. We then prepared to add DNA to the setup by filling reservoirs with a millipore-filtered, degassed buffer solution of

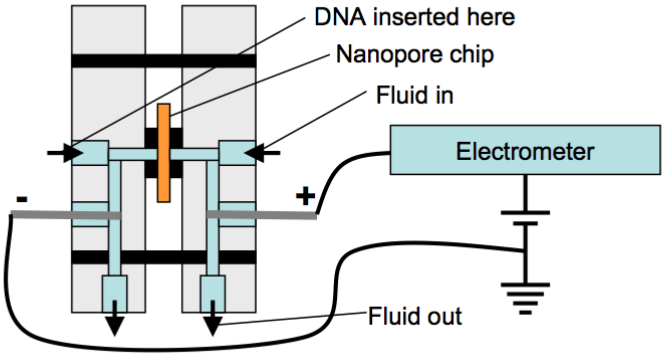


FIG. 5: A schematic of translocation experiment setup. (Taken from Nick Hagerty's thesis.)

1M KCl, 10mM Tris-HCl and 1mM EDTA buffer with pH 8.0. We placed the chucks in Faraday cages, plugged in the electrodes, and turned on our data collection equipment. To establish a control baseline - typically 15nA - we applied a 100mV voltage bias across the pore and began recording.

To record data we used an Axon Axopatch 200B high-speed electrometer which converts the analog signals received from the elctrodes into digital, numerical values. We sampled the analog signal at a rate of 250kHz. To control voltages we used a National Instruments SCB-68 field-programmable gate array (FPGA) card. FPGAs receive voltage inputs and send voltage outputs as functions of the inputs. FPGAs are designed to be customized with machine description code after purchase to implement complex computations (like our data analysis) on-the-fly to control experimental apparatuses. An FPGA could, for example, respond to a disturbance in current indicative of a translocation and respond by changing the applied voltage. FPGAs feature plentiful computation and fast memory resources to ensure that computations and their requisite reactions are executed as quickly as possible.

Procedures

We added DNA to our setup by filling the reservoirs with the same buffered solution as above with the addition of linearlized λ-DNA (48.5kbp, New England Biolabs) diluted to a concentration of 500ng/ml. This setup generalizes well to multiple types of nanopore experiment.

Monodirectional Translocations

Our monodirectional translocation experiments using asymmetrical nanopores seek to understand the effect of our structure's tunnel on translocation dynamics. We use thousands of events to measure the statistical difference in ECD when a molecule's moving length is altered. A typical monodirectional translocation experiment involves the application of a constant driving voltage for a predetermined experimental duration. We applied voltages ranging from 140mV to 60mV in 20mV steps biased in both directions for 10 minutes each. All voltages were set by hand. The exact order of tested voltages varied per-experiment but both polarities were run before moving to the next voltage.

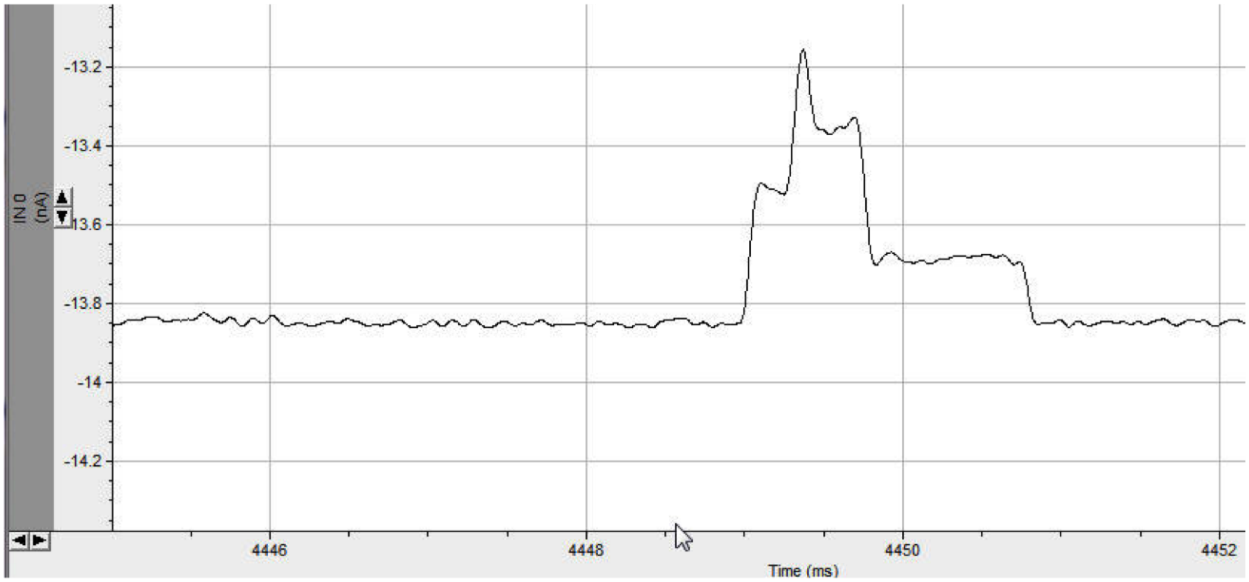


FIG. 6: A single translocation event. Notice that each displacement is an integer multiple of the displacement of the lowest level. Most events do not have such complicated structure.

Translocation events look like stacked-top-hat-shaped deviations from the baseline as seen in Fig. 6, indicating a DNA molecule is in the pore displacing ionic current. Current traces of translocations often feature multiple levels of displacement, indicating the presence of multiple strands of DNA. Typically multiple strands enter the pore because the DNA molecule is folded; rarely we see fluke events in which multiple DNA molecules translocate together. Rarer still are knots of DNA molecules that nearly completely block any current from flowing through the pore. Current can also be blocked by bubbles in solution lodging themselves in the nanopore. These two types of blockage create drastically different current

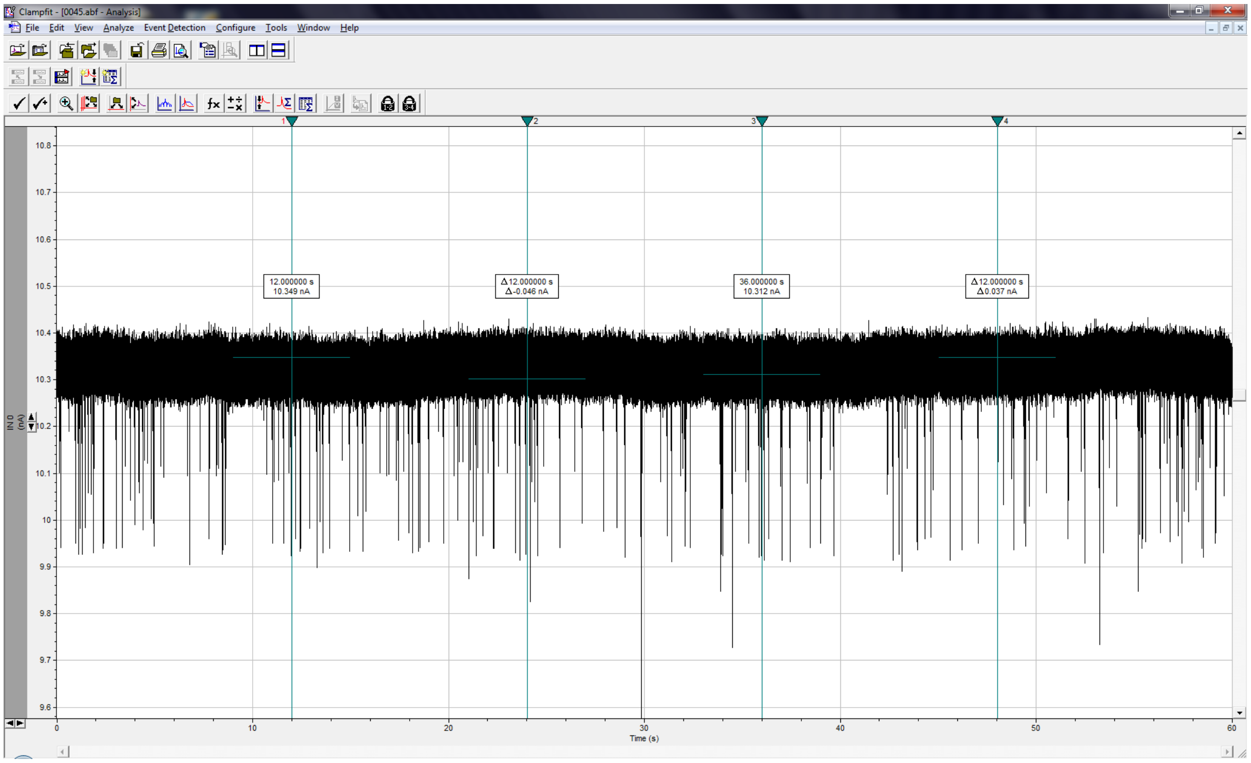


FIG. 7: A typical minute of monodirectional translocation data. Notice the current scale, noise, slightly unstable baseline and brevity of translocation events.

signatures: bubbles yield a smooth, uninteresting baseline that is much smaller than expected whereas bundled DNA yields an incredibly noisey baseline also much smaller than a clog-free pore. To dislodge blockages we “zap” the system with large voltage spikes and flip the polarity several times in rapid succession. A typical minute of collected data is shown in Fig. 7.

Trapping

Our preliminary endeavors to trap DNA molecules yielded data which we later realized was inconclusive; we only could show that molecules were translocating, not staying in the chamber. The procedure followed that of monodirectional translocation experiments except that DNA was placed only on one side of the nanopore. A voltage (of varying value per experiment) was applied to push DNA across the pore for approximately ten seconds. We flipped the polarity of the voltage to see if trapped molecules would reenter the pore. We then observed molecules translocating back across the pore! However, those translocations proved

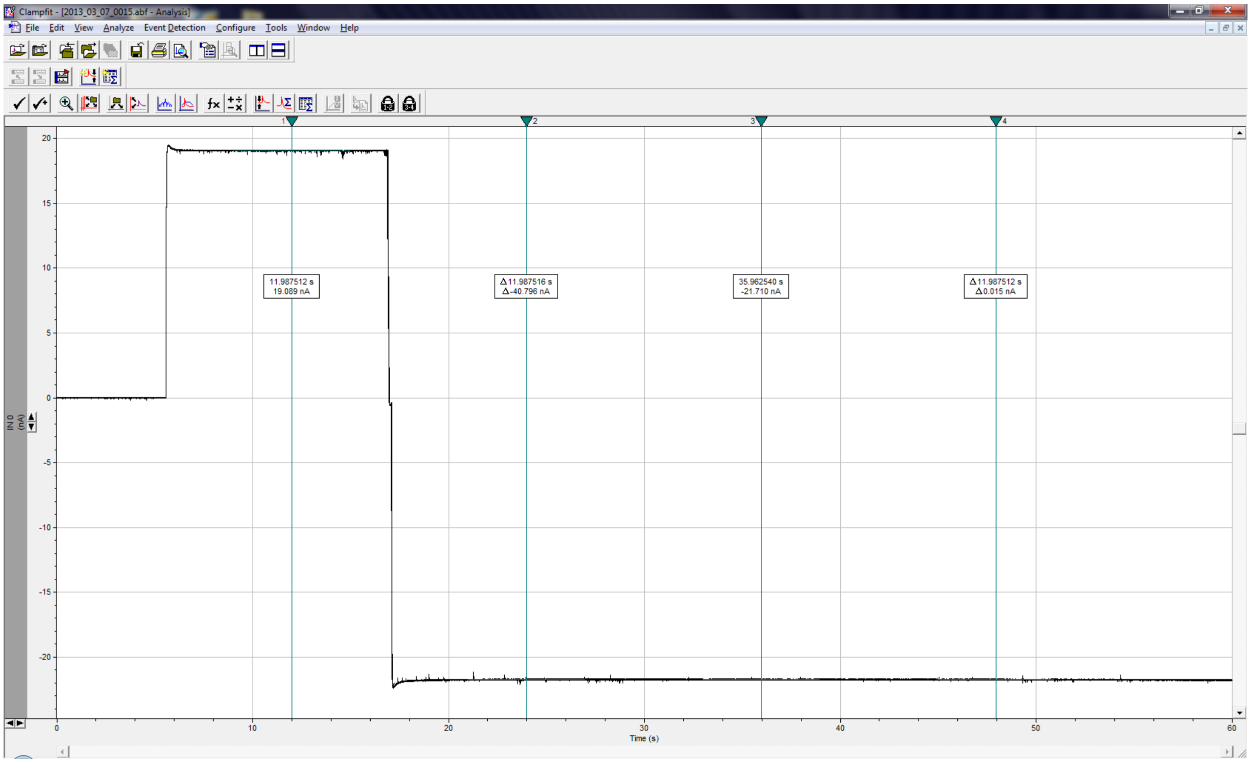
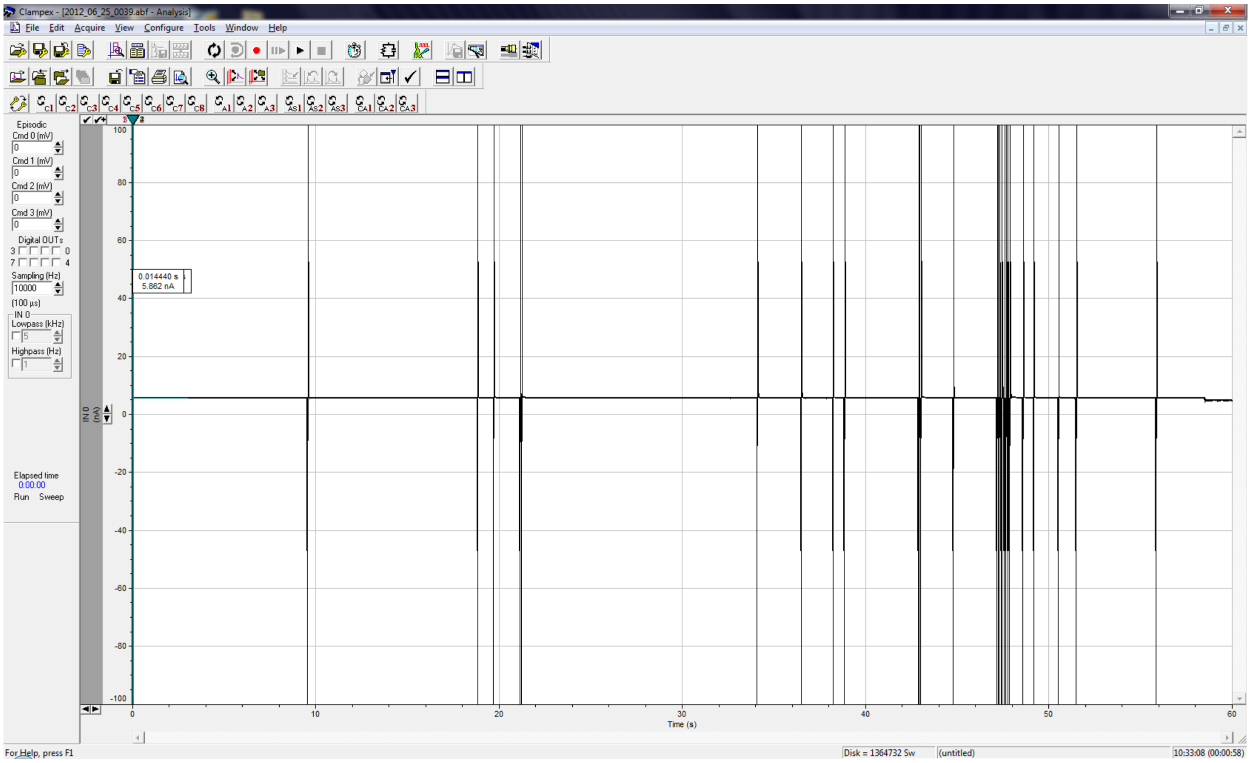


FIG. 8: A typical minute of trapping data. Notice the several events before and after the reversal of voltage polarity.

nothing except that it is possible for molecules to stay close enough to the pore to reenter after they have translocated; we had no way of knowing whether or not the molecules were actually trapped. An alternative procedure we tried was manually turning off the driving voltage when we observed a translocation. Every time trapping was attempted this way the pore was irrevocably clogged. A typical minute of trapping data is shown in Fig. 8

"Ping-pong" Translocations

Our ping-pong experiments used symmetrical nanopores (without the cavity and accompanying structure) to test if ECD is conserved on a per-molecule basis and to better understand the "recapture" time (explained below). Running a single molecule back and forth through a pore without allowing it to relax would definitively prove (or disprove) that ECD is conserved irrespective of molecular configuration, i.e. that ECD truly only depends on driving voltage and molecule length. Recapture dynamics can be better understood by altering the time waited before a voltage reversal.



Clampex - [2012_06_25_0039.abf - Analysis]

Episodic

Cmd 0 (mV) 0

Cmd 1 (mV) 0

Cmd 2 (mV) 0

Cmd 3 (mV) 0

Digital OUTs

3 □□□□ 0

7 □□□□ 4

Sampling (Hz) 10000

(100 µs)

IN 0

Lowpass (kHz) □ 5

Highpass (Hz) □ 1

Elapsed time 0:00:00

0.014440 s

5.862 nA

For Help, press F1

Disk = 1364732 Sw

(untitled)

10:33:08 (00:00:58)

FIG. 9: A typical minute of ping-pong data. Important features are the “zaps,” where flipping the voltage created a capacitive discharge which overloaded our sensor equipment. Notice the current scale as it is several orders of magnitude larger than that of monodirectional translocations.

The FPGA card played a much larger role ping-pong experiments. The experiments began in the same way as monodirectional translocations but diverged as soon as a translocation was observed. In molecular ping pong, the FPGA measures deviations from the baseline current. When a molecule is seen translocating as in Fig. 6, a timer, call it the flip timer, begins counting down from a predefined value between two and ten milliseconds. When the timer hits zero, the FPGA card flips the polarity of the voltage, hoping to recapture the most recently translocated molecule. Recapture here takes a different meaning than one may expect: instead of having anything to do with trapping in the cavity, recapture refers to the return of a molecule that has recently completed a translocation through the pore. Ping-pong experiments at present do not use pores with any cavity structure affixed; pores are stripped down to only one nanoscale hole in one layer of silicon. Upon recapture, another timer, call it the recapture timer, set to a different starting value, begins counting down. If the molecule is observed translocating again before the recapture timer hits zero, the flip timer begins again and voltage flips again. This process repeats itself until the recapture timer hits zero, meaning that molecule has finally escaped. A typical minute of collected data is shown in Fig. 9. Zooming on on a single zap yields events akin to Fig. 10.

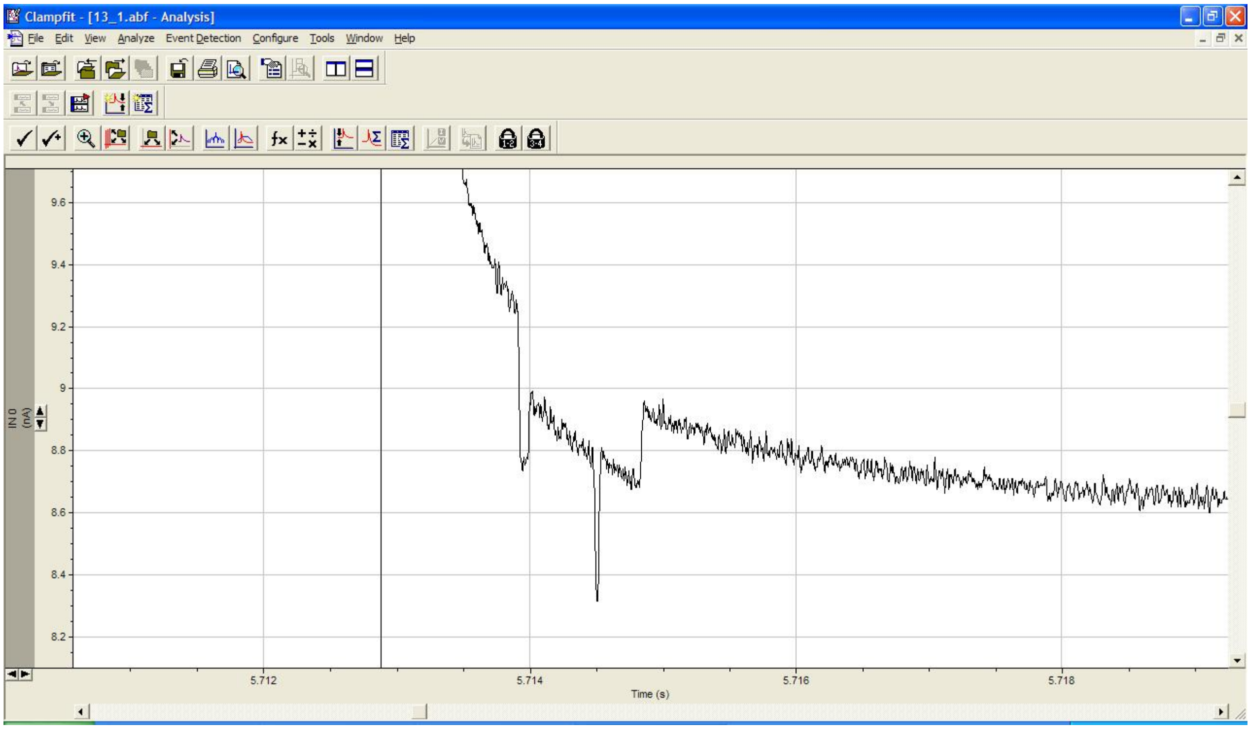

Clampfit - [13_1.abf - Analysis]
File Edit View Analyze Event Detection Configure Tools Window Help
IN 0 (nA)
9.6
9.4
9.2
9
8.8
8.6
8.4
8.2
5.712
5.714
5.716
5.718
Time (s)

FIG. 10: A particularly hard event to properly detect. The folded-ness of translocation and location on decaying baseline proved particularly tough to interpret.

Data Analysis with MatLab

After collecting current vs. time data at a rate of 250kHz we ran it through our MatLab data analysis pipeline to programmatically isolate events and determine ECD for each. Through automating data analysis I hoped to arrive at a heuristic for detecting translocations that had no false positives but also did not miss a single event. Each data file contains one minute of data. Current, measured in nanoamps, can be positive or negative (depending on driving voltage) but only its absolute value is used in computations to simplify algorithms. All analysis pipelines start by importing data files (either in pure binary or axon binary file formats) to MatLab-manipulable tables.

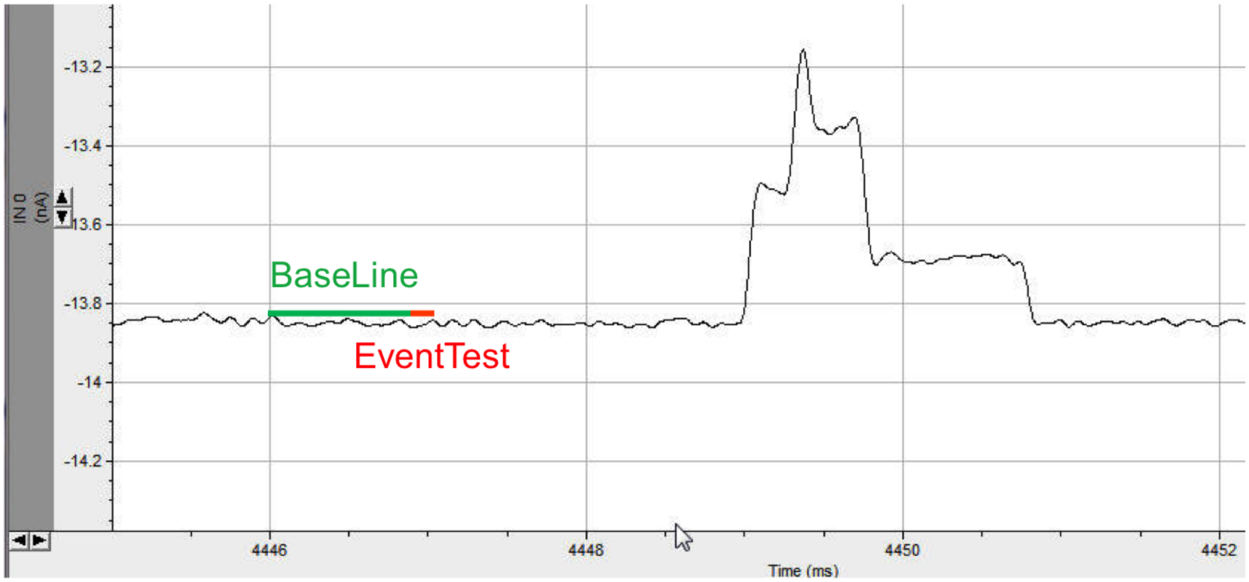


FIG. 11: A representation of how our MatLab script finds events in monodirectional data. Important features of the data are its relatively static baseline and event structure. The data pictured has been run through a 50kHz software filter for legibility; typical data is much noisier.

Monodirectional Translocations

Monodirectional experiments explored the effect of varying the moving length of translocating DNA on ECD. Xu Liu and I built a data analysis pipeline, using some of Angus McMullen's work, that takes the user through every step of data processing, from raw data to presentable figures. The data analysis for monodirectional translocations starts by detecting the start and end times of all events. Our heuristic for detecting translocations in monodirectional experiments is deviation from a baseline value β defined by the moving average of the previous 1ms of data (250 data points). We average the next 7 points, called the "lookahead" or "EventTest" Λ and see if it exceeds our threshold for an event. The last piece of data used to find event boundaries is the root mean square of the baseline $\Omega = \beta_{RMS}$. The RMS provides a good measure of the amount of variability, also known as noise, in the baseline. Each of these values is calculated tens of thousands of times per file as it is traversed from start to end with a step size of $1/50^{th}$ of a millisecond (5 data points).

If the lookahead is far enough away from the baseline ("far enough" Δ being dependent on driving voltage and specified by the user, typically one tenth of a nanoamp at 100mV), we know we may have found an event ξ:

$$|\beta_{start}| - |\Lambda| \geq \Delta \Rightarrow \xi = \top \tag{9}$$

where β is baseline value, Λ is lookahead value, Δ is the cutoff value, and ξ is a boolean value specifying whether we believe we are in an event. We save β_{start} to use when detecting the end of the event and continue taking the averages β, Λ and RMS measurements Ω as stated above. Instead of using a specified threshold as we did when detecting the start of an event, we determine the end of event by checking if the lookahead is greater than the saved baseline minus one tenth of the RMS (remember, all values are positive):

$$|\Lambda| \geq |\beta_{start}| - \frac{|\Omega|}{10}. \tag{10}$$

The other way an event can "end" is by being longer than 20 milliseconds at which point we know that the event flag fired erroneously, likely due to noise or a shifted baseline. Using Eq. 10 to determine when an event ends allows for some baseline variation during an event, but not so much that the first current blockage level is included. We finish data sanitation by subtracting β_{start} from all data points in an event to shift the baseline to zero and then set all points outside the recorded start and end (with some buffer) of the event to zero.

We then proceed to event classification. We construct a histogram of current values for all translocations by summing all data points in events as shown in Fig. 12.

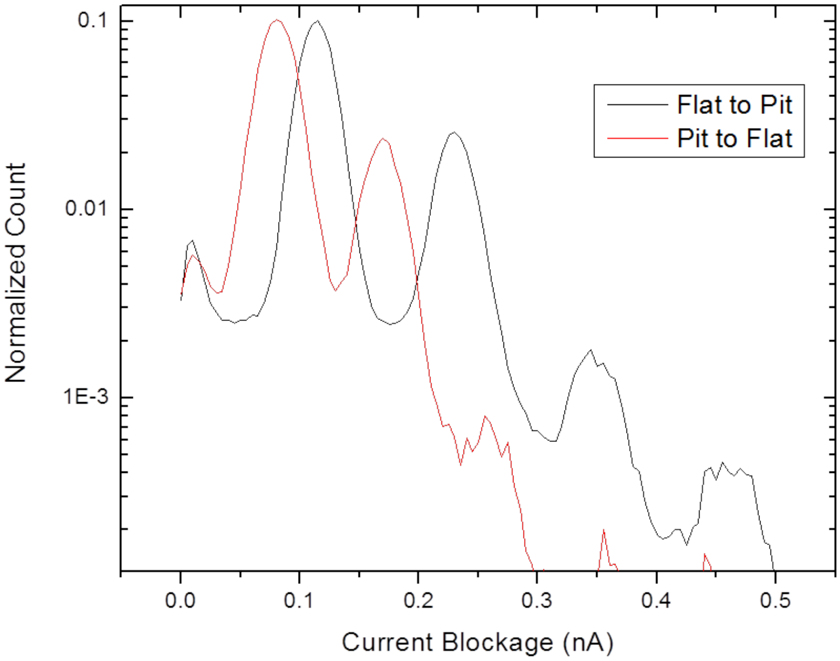


FIG. 12: The histogram presented to the user to determine single-stranded current blockage. The effect of the asymmetry of our structure is reflected in the differing current blockage values when translocations start from one side or the other. The first two peaks are well-defined because many events have at least some portion where the molecule is folded once. Higher numbers of folds are observed more rarely so the farther peaks are more poorly defined.

We fit the histogram using a double-peaked Gaussian fit function and take the value of the first (and typically tallest) peak as the current blockage of a non-folded translocation. We prompt the user to either agree that the peak value was properly fitted or to specify one. We use that value to determine the other current blockage levels (which are linear with respect to number of folds) and run through all events as we ran through baseline, classifying them according to the pattern of levels. The user then verifies that the given classification is correct, supplies one, or rejects the event. The final step in event processing is calculating ECD, which is done by summing all data points in an event. The rest of the pipeline takes analyzed data and creates graphs (making it useful but uninteresting technically).

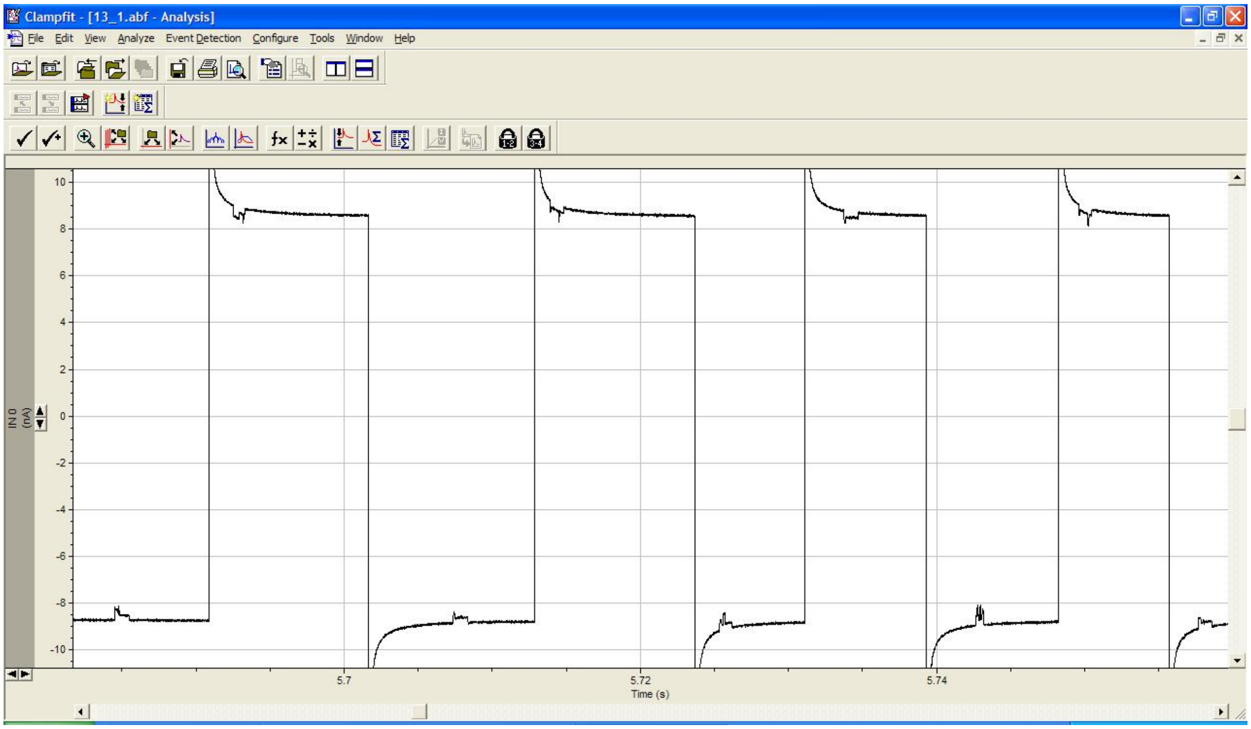

Clampfit - [13_1.abf - Analysis]
File Edit View Analyze Event Detection Configure Tools Window Help
10
8
6
4
2
0
-2
-4
-6
-8
-10
IN 0 (nA)
5.7
5.72
5.74
Time (s)

FIG. 13: A chunk of ping-pong data used to build our analysis algorithms. Important features are translocations on moving baselines and the events' structures.

Ping-Pong Translocations

For molecular ping-pong experiments we sought not only to measure ECDs of translocations, but also to group translocations together into single-molecule groups to understand how relaxation affects ECD by varying recapture time. The data analysis pipeline for ping-pong experiments proceeds largely as described above, but with more involved event boundary detection. The added complication is due to the nastiness of the data for which Fig. 13 is a good representation. Before we wrote code to detect and process events all analysis was hand-done, using voltage flips triggered by translocations to find events. This analysis became increasingly arduous as noise increased, triggering false flips; flips were no longer a reliable indicator of a translocation and hours were wasted looking at empty baseline hunting for a single event. Problematic features include, but are not limited to: increased noise in the data sets and translocations on moving baselines.

Moving baselines posed a problem because our only working automatic detection scheme relied upon a flat baseline and deviation from an average current value to detect an event.

This event trigger was always firing because current was always moving, necessitating a brand new detection scheme. Our first thought was to fit the exponential-seeming decay of the data and subtract it to allow us to use the same event detection method as for monodirectional translocations. Fitting seemed ideal as it would best preserve the exact shape of translocations relative to the baseline. None of MatLab's fit functions worked when fed our data; either they over-fit and removed the event or they did not return a function at all. We hypothesized that the sheer number of data points overwhelmed the fitting toolbox, so we tried sampling the data by averaging over various numbers of milliseconds. Fitting still did not work. Multiple problems arose while sampling, the greatest of which was that we would sample points within the event, pulling the fit function (if one was returned) into the event, destroying ECD measurements. So we began trying other possible heuristics for identifying events.

We eventually arrived at a robust heuristic to add to our detection arsenal by visual inspection of the data: use changes in the slope of the moving baseline. Because the absolute value of slope is always decreasing on moving baselines, we used an increase in slope as our final test for event boundaries. To find events, we first locate "zaps" in the data: places where the voltage flipped, causing a capacitive discharge that overloaded our recording equipment. We run once through the data file, recording locations of all zaps. We then search forward from the zap positions when searching for beginnings of events. An event must pass a multi-step detection procedure to be classified as such. First, we use our tried-and-true deviation from baseline current as a first flag. However, we greatly reduce the number of data points used to define baseline current to 20 from 250 in the monodirectional case. If lookahead current deviates far enough from the baseline (defined on a per-dataset basis, typically around .15nA), we calculate baseline slope by finding the inter-data-point slopes (that is, the difference in current between adjacent data points) and averaging them. We make the same averages for the lookahead data points. If the lookahead average is greater than baseline average by .05nA/point (an arbitrary cutoff determined by lots of fiddling) we know we have found the beginning of an event. Detection of event ends proceeds largely the same way except it looks backward from the next recorded zap.

Translocations made by the same molecule were grouped according to the zap that proceeded it. Zaps were assigned "zap groups" which were determined by their separation. We recorded the elapsed time between each zap, binned the values and took the mode to find the most frequent inter-zap time. If two zaps were separated by more than five times this mode we deemed them in a separate group. If two translocations were in the same group we calculated the recapture time in addition to ECD.

RESULTS AND DISCUSSION

My major result is the method of automatic detection of translocations on a moving baseline. While I have no results in the classical experimental sense, my work at worst improved grad students' quality of life and at best will inform planning of future experiments. Two test cases for the effectiveness of my script were 1) a minute of data without any falsely triggered voltage flips and 2) 30 seconds of data where most voltage flips were falsely triggered interspersed with legitimate molecular ping-pong. I compared event data acquired by running through the files by hand and with my script and found that neither data set proved problematic. For set 1 all events were found and start and end times were within .04ms (10 data points) of the exact value. For set 2 all events were found but one false positive was also recorded. When put through the wringer with true experimental data, the script found all events, but did not find their beginnings as precisely as wanted; the script is still useful for FPGA programming, but needs more tweaking to completely automate data analysis.

One class of event, an example of which is shown in Fig. 10, proved a particular problem for the ping-pong script. Most events are on less extreme slopes of baseline and do not have such unusual event structures. Events like these forced me to consider how to deal with odd foldings of recaptured molecules. Such edge cases were handled by nesting another loop in the event-end detection to ensure that the value found is at the event's end.

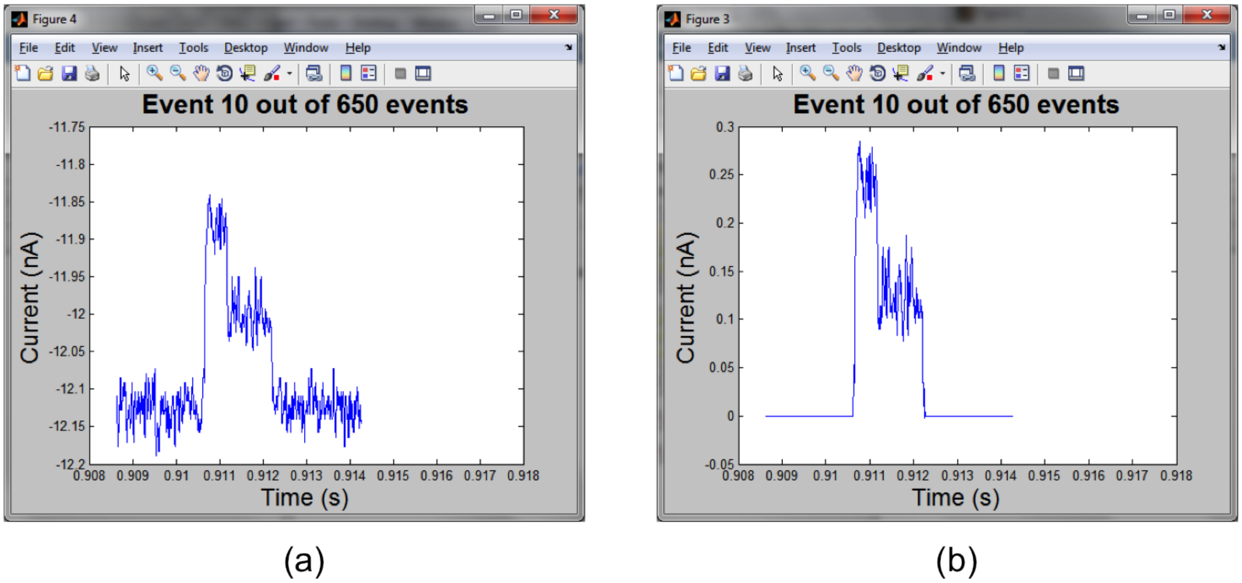


FIG. 14: How an event is processed by the script. (a) shows a portion of the untouched recorded data before any processing has been applied. (b) shows the same event after some processing which flattens the baseline outside of the event and shifts the event to have a baseline of zero to ease ECD calculation.

For monodirectional translocations all events are also found with a few false positives. The processing that goes into events is visible in Fig. 14. Notice that the baseline is shifted and that there is a small buffer of data points outside of the event that is left to ensure that no points within the event are zeroed because that would invalidate ECD measurements.

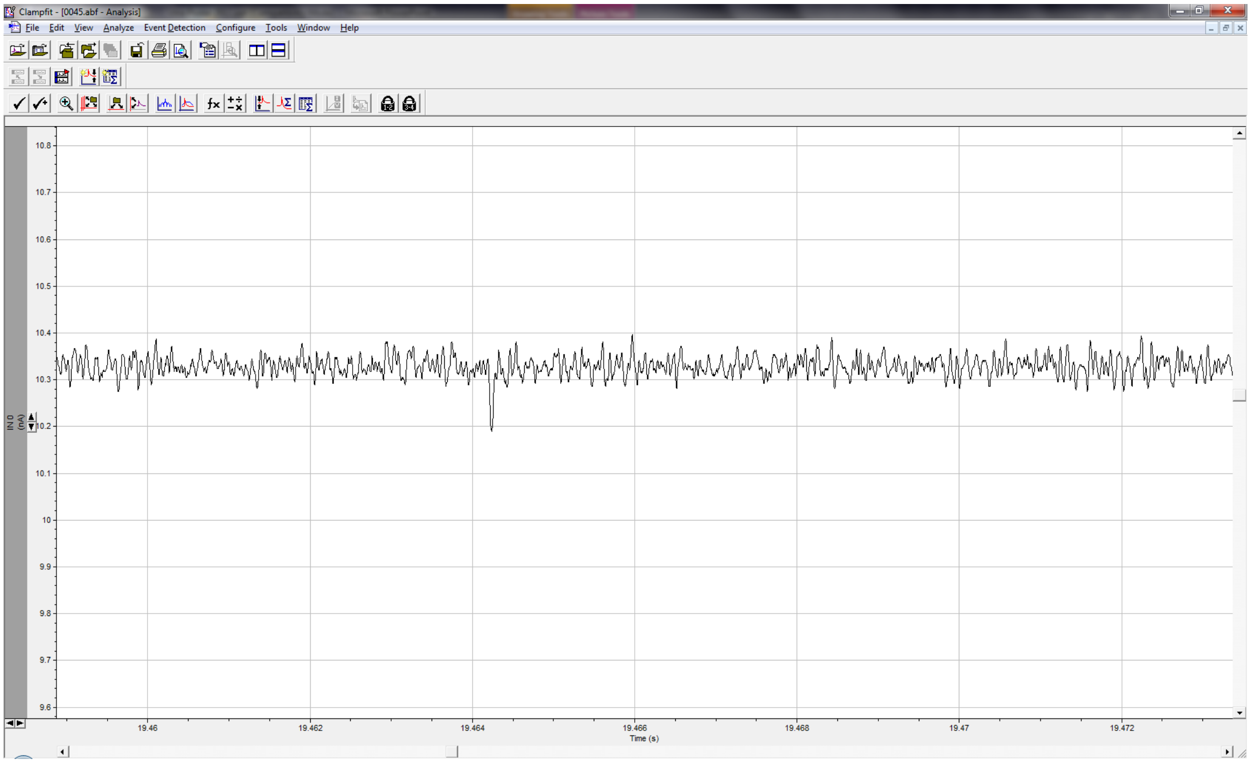


FIG. 15: A small current spike that typically gets falsely flagged as an event. The peak is approximately .11nA from the baseline, which is a reasonable cutoff for events.

One problem that still plagues monodirectional translocation analyses are "spikes:" small, unexplainable peaks that get flagged as events if the threshold is set low enough. An example can be seen in Fig. 15. Such false positives arise because the threshold for an event is set low enough that they get flagged. Ultimately, the user decides this threshold and it is obvious that setting the threshold too low and getting false positives is better than setting it too high and missing events.

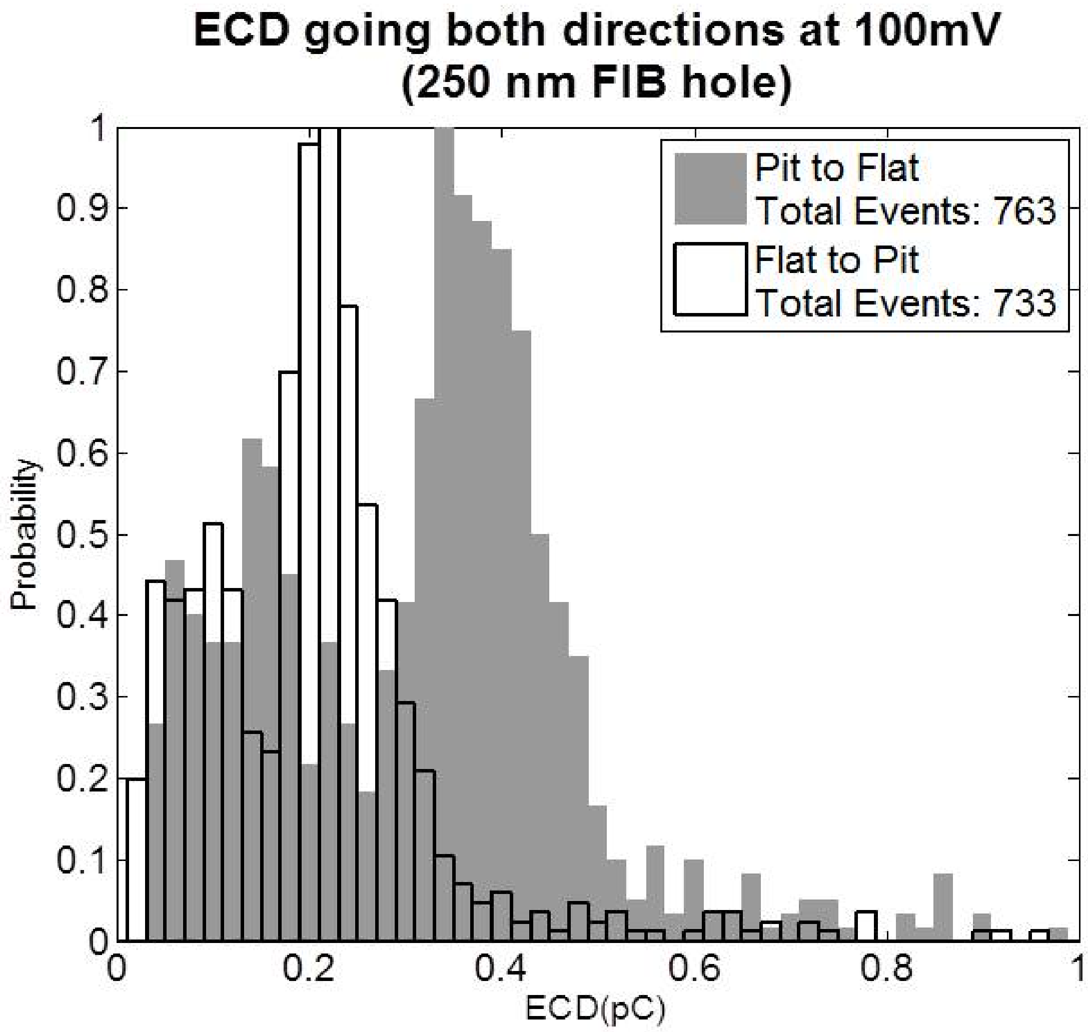


FIG. 16: One of the graphs produced by the data analysis pipeline.

The best and most straightforward way to avoid flagging spikes as events is to set a lower bound on event length. We decided not to do so because that type of analysis treads dangerously close to falsely biasing our data. Another method to combat such false positives is to offer some sort of suggested cutoff based on an analysis of the file. The major problem with such an analysis is that very few things are constant over even a minute of data: the baseline can shift by a nanoamp or more; pores can get clogged, utterly destroying any sense of a baseline in the data; and spikes could still be considered events, defeating the purpose of such an analysis.

Of of the graphs from the final product of the analysis pipeline is shown in Fig. 16. The smaller peak at roughly half of the ECD value of the larger peak is due to the splitting of

DNA molecules somewhere during the experiment preparation.

An as-of-yet unresolved issue with ping-pong event detection is dealing with a "double event," where two molecules translocate when only one is supposed to be ping-ponged. The current workaround is to discard any event ends found more than 5ms after the event start.

CONCLUSION

My work will improve ping-pong experiments by better detecting when a molecule has been recaptured. Presently most ping-pong sequences end with a molecule being recaptured and no voltage flip because the detection code on our FPGA uses a different method to detect a molecule's passage. Sadly, I did not have time to update the FPGA's code during my time in the lab. We chose to attack the problem with MatLab first due to the ease of rapid iteration it afforded. Now that we have arrived an a robust solution, implementing the derivative-focused translocation detection needs to be done to drastically improve data collection for ping-pong experiments by ping-ponging a molecule more and (hopefully) eliminating false voltage flips.

My work will also enable for detection of a trapped molecule's escape from the cavity. The experiment I envision proceeds as the preliminary efforts did except that, after the first translocation, voltage is turned off, hopefully trapping the molecule. We would wait at least 200ms to allow the molecule to equilibrate but likely longer to allow the molecule to diffuse away from the structure in case it was not trapped. When a driving voltage is applied with the opposite polarity to remove the molecule from the chamber the equilibration of ionic flow takes a couple of milliseconds as seen in the ping-pong data. (Note that the capacitive discharge may disappear but some sort of shifting baseline will definitely be present.) Those two milliseconds are the most likely time for a trapped molecule to escape. Our previous data analysis tools would not have been able to identify the molecule's exit. Now ours can.

[1] *Double helix serves double duty* (2013), URL `http://www.nytimes.com/2013/01/29/science/using-dna-to-store-digital-information.html`.

[2] W. R. JT Del Bonis-O'Donnell and D. Stein, New Journal of Physics **11** (2009).

[3] P. D. L. R. A. R. Fancesco Valle, Melanie Favre and G. Deitler, arXiv soft condensed matter (2008), URL `http://arxiv.org/pdf/cond-mat/0503577.pdf`.

[4] K. D. Dorfman, Reviews of Modern Physics **82**, 2903 (2010), URL `http://rmp.aps.org/pdf/RMP/v82/i4/p2903_1`.

[5] S. F. E. M. Doi, *The Theory of Polymer Dynamics* (Oxford University Press, 1988).

[6] R. P. Stellwagen NC, Gelfi C, Biopo **42**, 687 (1997).

[7] W. M. Haynes, ed., *CRC Handbook of Chemistry and Physics* (CRC Press, 2012-2013), 93rd ed., URL `www.hbcpnetbase.com`.

[8] M. Z. U. Z. D. K. P. E. B. N. H. D. C. D. Meng-Yue Wu, Ralph M. M. Smeets and H. W. Zandbergen, Nano Letters **9**, 479 (2009), URL `http://pubs.acs.org/doi/abs/10.1021/nl803613s`.