Novel Tools for Biophysics Research

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Tanuj Aggarwal

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Murti Salapaka

January, 2012

Acknowledgements

My journey to this point was definitely not a solo ride. Many friends and family members made it possible for me reach here. The inspiration began with my parents who always encouraged me to do my deed without worrying about the end result as it was advocated in the Geeta. That has been my motto throughout.

I am exteremely lucky to have incredible set of friends in my circle that take the gloom away, each one uniquely inspiring and educating. Shourya Otta, Bipin Kumar, Hullas Sehgal, Pranav Agarwal, Vikas Yadav, Tathagata De, Vipul Katyal, Govind Saraswat, Rajat Mittal and Pushkar Modi, Subhrajit Roychowdhury are among my closest friends who were around me for a good portion of my stay as a graduate student making my life merry.

I would like to thank Tom Hays, Rob Davison and Mingang for their invaluable help in performing assay protocols during the later part of my PhD.

Finally, I am grateful to Dr. Murti, my adviser who guided me not only professionally but also at a personal level in brining out my personality and inspiring me to brave any challenge that I face with honesty and hard work.

I am forever indebted to all of you and many other friends not mentioned here.

Dedication

To my parents, P. K. Aggarwal and Rajni Aggarwal

Abstract

This dissertation is aimed toward furthering biophysical studies using novel instrumentation techniques and algorithms. This work describes the development of optical tweezers as an instrumentation platform to study single bio-molecules. Algorithms for signal processing, data analysis are also developed in this thesis within the purview of single molecule experiments but also extensible to a wide array of applications. These tools are tested on experimental data obtained from samples prepared using protocols that are also a part of this work. Key contributions of the thesis includes the following: (i)Construction of optical tweezers with an emphasis on measures taken to reduce the unwanted extrinsic noise in the measurements. (ii)Development of an artificial neural network based technique to increase the measurement range of the instrument by twice the previously reported value. (iii) Application of a recursive least squares based approach to estimate the persistence length of a double stranded DNA molecule in real-time. (iv) Theoretically analyzing a coupled oscillator system as a sensor compared to a single oscillator system. (v) Development of analysis tools and experimental schemes to extract parameters that model kinesin flexibility. (vi) Development of a three-dimensional, multi-motor simulation environment to understand complicated dynamics of multi-motor transport and better interpretation of experimental data. (vii) Development and analysis of an algorithm to fit step signal to a noisy data from single molecule experiments that give better fitting than existing algorithms. The step detection algorithm is further extended to detect events like sudden changes in the parameters of the system in presence of non-linearities.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Biophysics is an interdisciplinary science that borrows ideas from other disciplines in order to explain or predict behavior of biological systems. It is a relatively much younger discipline compared to traditional biology. Due to its interdisciplinary nature, it complements traditional biology by bridging gaps that are left behind by biology. It touches a wide spectrum of application areas in biology, so much so that distinction between the two is gradually fading. Thanks to recent technological advancements, biophysics is enjoying a boom period with many researchers from different disciplinary backgrounds contributing in this area. In particular, novel technologies like Atomic Force Microscopy (AFM) and Optical Tweezers have boosted single molecule research making it an active area of research for the past three decades. Through this dissertation, I hope to contribute my bit of innovation that will help biophysics researchers to take one step further to their path to scientific exploration. In this work, Optical tweezers is developed as the choice instrument for studying DNA and motor proteins like kinesin. Signal processing techniques and novel instrumentation ideas are also developed that forms a major part of the thesis.

Optical tweezers is a laser based instrument capable of trapping microscopic dielectric particles in fluid or vacuum. Particle manipulation capability of laser light was first demonstrated by Arthur Ashkin in 1971 [1] by levitating a microsphere against its weight using a laser beam. The first true optical trap was however invented in 1986 and reported in [2]. Since then, a slew of experiments have been performed on biomolecules by trapping variety of biological subjects. Optical tweezers have found interesting applications in the field of biophysics where physical and dynamical properties of objects as big as cells and microorganisms like bacteria, down to biomolecules like proteins and DNA are studied. Optical tweezers are force and position transducers with high sensitivity. It can balance forces ranging to 200 pN and has sub-pN force measurement resolution. Position measurement resolution of sub-nanometers is achievable with optical tweezers.

The fundamental science behind optical trapping principle can be understood by an approximate simplified theory based on light's momentum property. When it passes through a dielectric particle (henceforth, it is assumed that the particles are microspheres or beads), it reflects and refracts. In both cases there is a change in the incident momentum of light that imparts a force on the bead. In order to stably trap a bead at a particular location, the net change in the momentum of light passing through a bead at that location should be zero, i.e., there should be a balance of forces. Additionally, any small deviations of the bead from the nominal position should result in restoring forces that tend to bring the bead back to its nominal position. It turns out that by strongly focusing a parallel beam of light, a stable trap is created at the focus. A bead at this location experiences two kinds of forces. One tends to push the bead away from the focus (scattering force) and the other attracts the bead toward light field's highest intensity point (gradient force). See Figure 1.1 for illustration. These forces balance each other under equilibrium for a stable trap. The bead also experiences a restoring force if it deviates from the nominal position. In reality, the particle is never at rest because molecules of the fluid medium are constantly colliding with the particle.

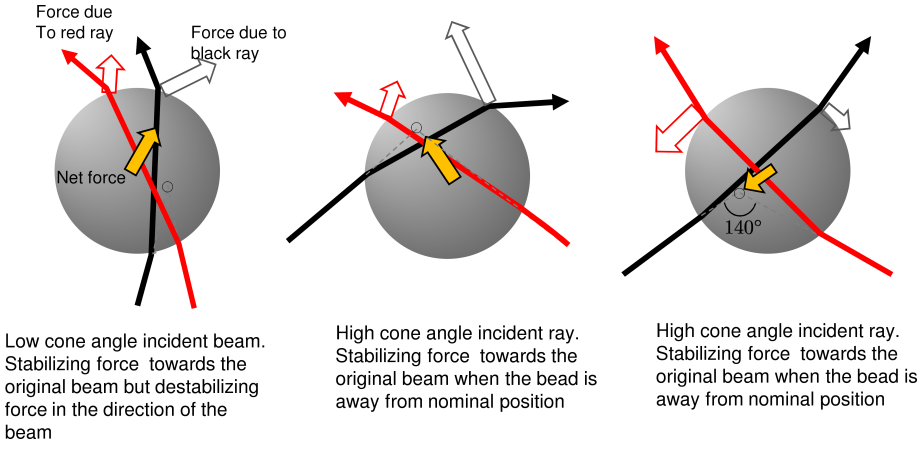


Figure 1.1: Ray diagram model for principle of optical trapping

The random collisions result in Brownian noise or thermal fluctuations of the particle about its nominal position. This forms a source of noise in optical tweezers measurements. System modeling and identification techniques are leveraged for design of better noise filters.

An ideal instrument must be thermally limited as far as noise is concerned that means minimizing measurement noise and instrumental drift as much as possible. To this end, construction of an in-house developed optical tweezers system is described with emphasis to special actions taken to reduce extrinsic noise in the system to obtain a state-of-the-art optical tweezers system. Optical tweezers have been in existence for over two decades and

significant of instrumental breakthroughs over this period have improved the performance and flexibility of the instrument resulting in studies leading to better understanding of subjects under study. In an attempt to further these efforts, as part of the thesis, a neural network based technique is developed that increases the range of a typical optical tweezers measurement system to twice the previously reported value. Complimenting the physics of optical trapping is the automation of the control systems of the instrument which is indispensable for some of the modern assays. The programming architecture that allows convenient experiment-time automation and control is also described. With optical trapping setup in the backdrop, two model bio-systems have been studied - DNA and kinesin. Details of the biochemical protocols used and their modifications are also described.

Persistence length is a parameter that characterizes mechanical stiffness property of chain-like polymers like DNA and proteins. By coupling DNA molecules to microspheres, these molecules are stretched in a thermal bath using optical tweezers, as a result of which, entropic tension is developed within the molecule characterized by its atomic structure. Force vs. extension curves for such molecules are primarily characterized by their persistence length and their contour length. As a part of the thesis, a recursive least squares based approach is developed to estimate the persistence length of double stranded DNA molecule in real-time from force vs. extension behavior. DNA coupled microspheres point toward an interesting paradigm of network based sensing. It is already known that multiple independent sensors always give an improvement in signal to noise ratio. However, the case of using multiple coupled sensors has not been investigated much. As a part of this dissertation, a first attempt is made toward finding the potential benefits, or the lack of it in using two coupled beads sensing a single quantity like force instead of using a single bead.

Kinesin is another interesting model system which is an active subject of investigation. There are several unknowns about the precise functioning mechanism of kinesin. It also provides a rich bed of interesting problems that engage researchers from different areas. Kinesin is a motor molecule that has physical motility within cells. It takes 8 nm steps on microtubules and is capable of moving against a load of about 6 pN. Assays to perform measurements on this molecule using optical tweezers forms a major component of this dissertation. Data from optical tweezers based kinesin assays is severely corrupted with noise from fundamentally limiting sources. An algorithm to find and fit step signal to the kinesin motion data is developed in the thesis that gives better step size detection than existing techniques. Kinesin is also a flexible molecule with interesting structural elements that give it unique functionality like presence of hinge joints and swivel joints within the protein stalk. A preliminary approach to parametrically model the kinesin flexibility is

developed that explains experimental data. The results obtained are also verified through a three dimensional Monte-Carlo simulation of optically trapped bead being transported by multiple motors. Through these simulations some previously unanswered questions on cargo transport by multiple motors and the effect of nonlinear elastic behavior of motors is elucidated.

Chapter 2

Instrumentation

2.1 Construction

The experimental setup (Fig 2.1(a)) consists of a TEM_{00} mode, 1064 nm wavelength trapping laser source (Crystal Laser, 500mW) that passes through a 2-axis acousto-optic-deflector (AOD, IntraAction Corp., DTD-274HA6). The beam is expanded by a pair of lenses LT1, LT2 and steered into the microscope objective (Nikon 100x, 1.4NA, oil immersion) using mirrors and lenses. The focal lengths of LT1 and LT2 are matched so that the expanded beam slightly overfills the back aperture of the objective. Too much expansion results in loss of trapping power due less intensity passing through. On the other hand, under-filling results in a weaker trap due to absence of circumferential rays that are mostly responsible for stable trapping. Central rays have a destabilizing effect as they try to push or scatter the bead away. A telescopic lens assembly, LC1, LC2 is used such that beam rotation at the output aperture of AOD is reconstructed at the back focal plane of the objective. Beam rotation at back focal plane of the objective transforms into beam translations at the front focal plane of the objective. Therefore, the AOD can control the position of the trap in two dimensions. Detection laser (Point Source Inc., iFLEX 2000, 50 mW, 830 nm, p-polarized) is introduced collinear to the trapping laser using a polarizing beam splitter cube (PBS). PBS reflects s-polarized beam and allows transmission of p-polarized beam. Intensity of the detection beam is reduced by placing a neutral density filter (ND) in its path. Intensity of the detection beam is kept low, just enough to make the photodiode sensitive to bead movements. Detection beam is also expanded using a pair of lenses, LD1, LD2. These lenses are arranged such that maximal sensitivity is obtained. In particular, LD1 is translated along the optical axis that changes the vertical position of the beam focus with respect to the trapped bead. When the focus of the beam is slightly below the trapped particle, maximum position measurement sensitivity is obtained. The combined trapping

and detection beams are directed into the back aperture of the objective using a hot mirror, HM1. Hot mirrors reflects light of longer infrared wavelengths and allow transmission of visible light spectrum. The microscope focuses the light into the sample. The sample consists of fluid channel created by sticking two cover glasses using double sided tape. The coverglass is supported by a 3-axis nanopositioning stage (Thorlabs 3DMax, closed loop) that allows nanometer level resolution and control over sample positioning in three dimensions. The laser leaving the sample is collected by a condenser (a 40x microscope objective) that collimates the expanding laser beam. Hot mirror, HM2 then directs the light toward a quadrant photodiode module, QPD (Pacific Silicon Sensors, QP50-6SD2). A conjugation lens, LD3 is placed between HM2 and QPD that images the back focal plane of 40x condenser on the QPD plane. This helps in desensitizing QPD measurements to movements of 40x objective. An 830nm laser line filter (Thorlabs, FL830-10) is placed before the QPD to allow only the detection laser light to fall on the QPD to avoid signal corruption by trapping laser light and fluorescence laser light. The photodiode module provides two voltage signals that are proportional to the asymmetry of light distribution on the photodiode quadrants along the horizontal and the vertical axis. A third signal from QPD is a measure of the total intensity of light falling on all the photodiode quadrants. The signals are amplified using custom designed circuit to fill the data acquisition capture range of $\pm$10V. These signals are captured by a FPGA (Field Programmable Gate Array) based data acquisition card (National Instruments, 7833R). Control logic and voltage to position mapping is programmed on this hardware using custom written code in LabVIEW for FPGA. An LED based white light source is placed over HM2 to illuminate the sample for bright-field imaging. The light travels down the condenser into the sample. The objective images the sample to infinity. A projection lens, LP1 is used to refocus image at infinity on to the camera, CM1.

The entire setup is compacted and mounted on a 2'x3' optical breadboard placed on a vibration isolation platform (BM-8, Minus k Technology). In addition, the setup is housed inside an acoustic hood that isolates the setup from external light, air currents and acoustic disturbances.

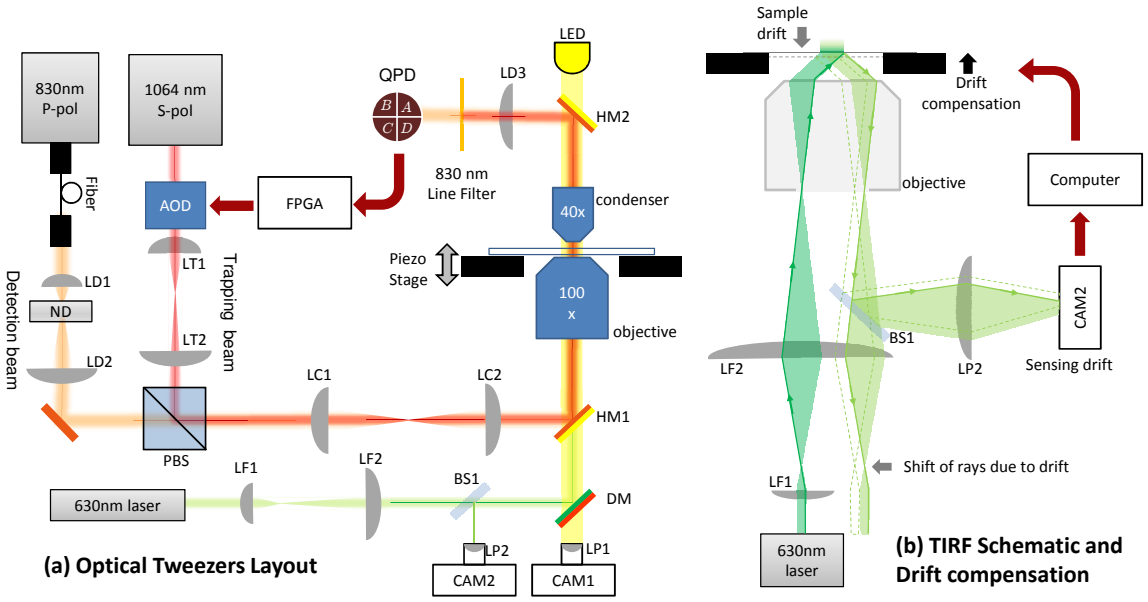


Figure 2.1: Experimental Setup Schematic

2.1.1 TIRF and drift compensation

The setup also houses a fluorescence assembly to enable simultaneous trapping and fluorescence/bright field visualization. A 30mW green (630 nm) diode laser module is used to excite the sample in Total Internal Reflection Fluorescence (TIRF) mode. The TIRF scheme is illustrated in Fig 2.1(b). The excitation laser is expanded using a pair of lenses, LF1 and LF2 and directed into the objective by a dichroic mirror, DM that reflects green laser but allows transmission of red fluorescence emission. The lenses are arranged such that the beam is focused at the periphery of the back aperture of the objective. The laser beam is collimated by the objective and incident upon the sample at a very steep angle. If this angle exceeds the critical angle of refraction at the glass-water interface of the sample, then it results in total internal reflection of the incident beam. Majority of the beam is reflected back into the objective, however, some portion of the light escapes into the sample in a form of evanescent waves. Its intensity diminishes exponentially as measured away from the glass-water interface. Only a few 100nm of the sample on the surface of the cover glass is excited by this light. This results in high contrast or signal to noise ratio in fluorescence image. The reflected light is sampled by a coverglass beam sampler, BS1 and directed into the camera, CAM2 through a projection lens, LP2. From Fig 2.1(b), it can be seen that, if the sample moves in vertical direction relative to the objective then the reflected laser light shifts its path, as a result, the image of the laser spot on CAM2 also shifts. This shift can be detected using image processing techniques in computer. The computer can then send a signal to the piezo stage to compensated for the drift in the sample position to maintain

location of the spot imaged by CAM2. This way drift compensation is achieved. In this setup the sample focus can be maintained to within 5nm in less than 1Hz bandwidth.

2.1.2 Optimizing the setup

In the development of optical tweezers, getting high resolution is a challenge because of precise alignment requirement, various noise sources that can potentially corrupt the measurements and requirement of automated calibration and testing procedures for practical usage. Small deviations of the optical components from their optimal positions can result in poor sensitivity and additional noise coupling from external environment. In this section various sources of noise and alignment issues are discussed with possible resolution techniques. Most diagnosing procedures involve analyzing the power spectrum of the photodiode signals for abnormality. The main sources of noise are as follows,

Quantization Before optimizing noise by analyzing power spectrum, the base level noise must be understood. The base level noise arise due to the finite number of bits used for sampling voltages by the data acquisition. A data acquisition system has limited resolution depending on the number of bits used for storing voltage information. A voltage signal can jump from one bit representation to another randomly even if the change is miniscule due to the rounding off process. This results in Analog to Digital Conversion (ADC) quantization noise. Consider an ADC with minimum measurement unit of Δvolts. Its probability distribution is uniformly distributed between $\pm\Delta/2$, i.e., $p(x) = 1/\Delta$, $x \in [-\Delta/2, \Delta/2]$. Then the variance is given by

$$\int_{-\Delta/2}^{\Delta/2} x^2 p(x) dx = \frac{x^3}{3\Delta}\Big|_{-\Delta/2}^{\Delta/2} = \frac{\Delta^2}{12} volts^2$$

If this is being sampled at F_s then its noise power spectral density is,

$$\frac{\Delta^2}{12}\frac{2}{F_s} = \frac{\Delta^2}{6F_s} volts^2/Hz$$

E.g., 16 bit ADC can measure approximately $\Delta = 0.3mV$. This if sampled at 100kHz will give a noise level of $\frac{(3e-4)^2}{6*1e5} = 1.5e-13\, volts^2/Hz = -128dB$. In terms of ADC counts $\Delta = 1$, noise power density $= 10\log_{10}(1e-5/12) = -60dB$.

Electrical noise The optical table is magnetic and can act as a powerful antenna for capturing extrinsic noise. Strong eddy currents can develop on the surface of the optical table and through radio transmission, it can enter photodiode electronics. This current should be channeled out to the building/earth ground by connecting a

wire to the optical table surface and building neutral grounding pin that may also be available through the power supply. Using breadboard and unshielded wires is also not recommended as they can allow additional ambient electromagnetic noise into the circuit. Custom circuits should be made using surface mounted electrical components in a small sized PCB so that the components are placed closed to each other. Electrical components should have low noise performance specification.

Shot noise Shot noise occurs in electronics due to fluctuations in current determined by the number of electrons passing through. For small currents, the fluctuations in this number may form a larger percentage of the mean current. The signal to noise ratio depends depends on the square root of the magnitude of current. Larger the better. Therefore, to optimize photodiode sensitivity, highest intensity light as allowed by the experiment and sensor limits should be used.

Air currents Air in a room is never completely static. External disturbances create localized 'wind' of small magnitudes not detectable by human sensory organs. Nevertheless, these disturbances change the local refractive index of the medium slightly and affect the path of laser beam. Random fluctuations due to air currents appear as random positional noise in the system. It affects both the detection and trapping beam. The solution is to cover the beam paths with a barrier or house the entire setup inside a sealed hood. In our setup, the effect of air currents can be distinctly observed for frequencies below 10 Hz.

Acoustic noise Acoustic noise is also coupled to air but in the frequency range of 20 Hz to 1 kHz. However transmission of acoustic noise to optical tweezers is through different phenomenon. It excites the mechanical resonances of the optical mounts. Typical frequencies that get affected are in the band of 20-200 Hz. Acoustic noise is easily diagnosed by reproducing a sinusoidal note through computer sound card and measuring the response on the instrument. If a peak appears in the power spectrum of the measured data, at the same frequency as the sound, or if its magnitude changes proportionally to the change in the magnitude of sound then it confirms that acoustic coupling is there. In order to pin point exact optical component most susceptible to acoustic noise, one can do a perturbation test. In this test, each optical mount is perturbed slightly mechanically using a flexible stick. Uniform perturbation must be provided to all the components. The response to this perturbation is measured for every mount. The mount that is most sensitive will give the strongest response. These mounts must be tightened and their mechanical resonant frequencies should be pushed to higher frequencies by putting additional stiffening supports. Reducing

the height of the mount posts also helps. Additionally, placing the entire instrument inside an acoustically sealed hood eliminates the noise.

Mechanical vibrations Another prominent source is the building vibrations. These vibrations shake the entire instrument and consequently this noise enters the measurement system as well. Vibrations in the building has several sources too. Wind gusts are known to shake buildings significantly. Upper floors are particularly very sensitive to the wind fluctuations. Therefore, it is a good idea to choose basement floors to house optical tweezers instrument. The downside of choosing basement floors is that they may house noisy infrastructural machines like HVAC system, ultra low temperature refrigeration systems that generate significant amount of acoustic and mechanical vibrations that couple through walls and floors to the instrument. Typical frequencies affected are in the range of 20-200 Hz. For larger optical setups, air-float optical tables may be used that provide some degree of vibration isolation. For smaller setups, a better and cheaper alternative is to use optimized passive isolation platforms like those by Minus-K Technology. Special considerations are made to ensure stiffness of the overall system is as high as possible. To this end, cantilever mounts are avoided because they amplify the vibrations. Typical microscopes place objective on a cantilever mount so that the objective can be positioned conveniently. However, this makes the setup very sensitive to vibrations as the objective significantly amplifies the effects of vibrations. In our setup, the objective is placed on a metal plate with a custom drilled apertures. The metal plate is firmly mounted by posts at the four corners of the plate so that the cantilever arrangement is avoided. Shorter posts for mounting optics may be used to increase the resonant frequency of the components. A lot of noise can be coupled in due to the cables reaching and coming out of the setups. Thicker and stiffer wires are strongly coupling compared to thinner and lose wires. If possible the cables may be clamped to the table but kept lose as much as possible.

Ambient light It is observed that when fluorescent light sources lit up, noise in frequency and its multiple of 60 Hz and 120Hz is strongly present suggesting that the light is fluctuating and photodiode is sensitive enough to capture it. To cut this noise out, extrinsic light sources need to be turned off and/or the photodiode needs to be covered. Additionally a dichroic filter is placed right in front of the photodiode sensor that allows detection wavelength (830 nm) to pass through while filtering out visible wavelengths of light.

AOD Proper orientation of a two axis acousto optic deflector is necessary to ensure uniformity in the trap intensity with trap position and for optimal transmission of first order diffracted light from the AOD. AOD operates over radio waves of MHz frequency range. High frequency noise is often noticed in the signal due to the beam passing through the AOD onto the photodiode. This is diagnosed by observing changes in the frequency of the high frequency peaks whenever the AOD input frequency is changed. The observed peaks are likely to be the aliased components of the RF waves not filtered out by the data acquisition system. As these peaks are in very high frequency, they are not expected to affect trap performance. However this beam shouldn't be used for detection purposes. Such peaks could also be due to the aging of AOD because over time the sound absorption element in the AOD degrades and starts reflecting some input waves back that resulting in standing wave formation within the AOD crystal that depends on the input frequency. This results in a nonlinear operation motion of the trap that look like 'wiggles'.

Faulty Equipment It has been observed that various equipments like laser sources and power supplies are potential sources of noise if they are faulty. They are not easily diagnosed because they don't have a characteristic signature and may appear as other kinds of noise sources.

Sensitivity Multiple alignment factors affect the sensitivity of the instrument. The diameter of detection beam and its degree of collimation affect the range and sensitivity of the measurement system. It also depends on the bead size therefore realignment may be necessary whenever beads are changed for optimal operation. A thin beam has larger linear detection range compared to thicker beam. Changing the relative focal point point of the detection laser beam after to the objective or the trapped bead has significant affect on the sensitivity. Relative focal point can be changed by either having a slightly diverging or converging beam instead. Converging beam will bring the focal spot closer to the objective and vice-verse for diverging beam. It is observed that having detection beam focal point slightly lower than the trapping focal points gives best sensitivity. Another important component in obtaining high sensitivity is the condenser lens/objective. Higher numerical aperture condenser allows greater amount of light collection therefore intensity of the light falling on the photodiode is higher and consequently sensitivity, which is dependent on the light intensity, increases. This also means that intensity of detection laser source can be reduced to get a desired sensitivity while not contributing in trapping a particle.

2.2 Programming

Entire programming was done in the LabVIEW graphical programming environment that provides streamlined interface between hardware and software. The programs are designed to acquire data from sensors and send control signals to the actuators. Besides this basic task, a single programming interface allows for offline processing of data for a variety of calibration purposes and for continuous data logging. This is possible because there are two user programs running simultaneously, one in hardware (target) and the other in software (host). These programs can communicate with each and exchange data. LabVIEW streamlines the interfacing by providing blocks for data acquisition and communication, however the architecture of the program needs to be carefully designed for maximum flexibility and increased complexity without compilation errors. Code written for the hardware (FPGA) in the LabVIEW environment has limited resources therefore the code has to be optimized if higher performance and complexity is desired. Speed and easy programmability of NI FPGA however comes with a constraint, it can perform integer computations only. Therefore special efforts are required to ensure accuracy of computations performed within the hardware. An broad overview of tasks shared between host and target is illustrated in Figure 2.2.

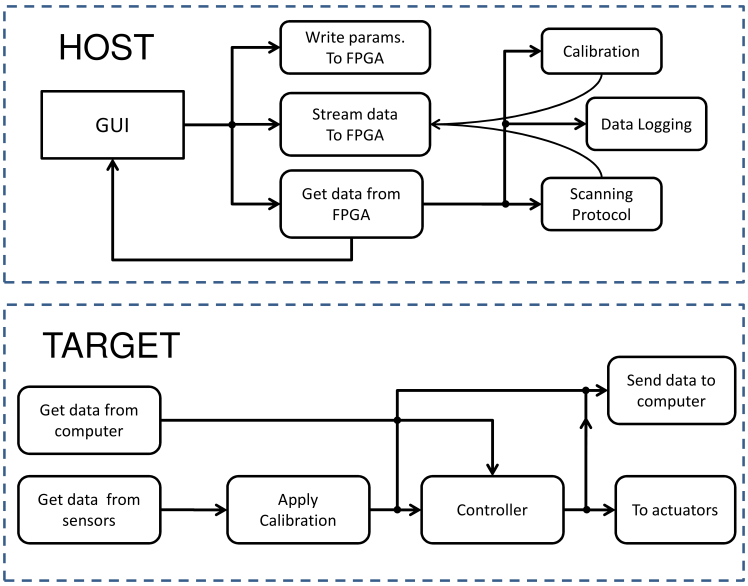


Figure 2.2: Programming overview

Individual blocks in Figure 2.2 are further explained next.

2.2.1 Host

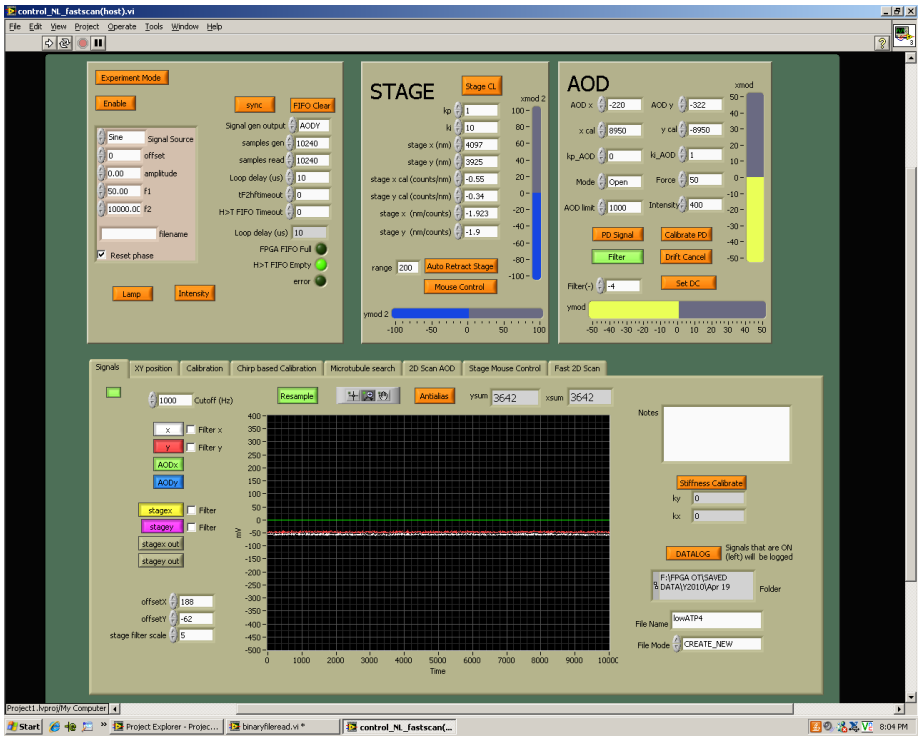


Figure 2.3: Screenshot of Host GUI front panel

GUI is the Graphical User Interface that allows the user to change variety settings pertaining to data acquisition, control and data analysis. Data acquisition parameters like sampling time and number of samples desired can be changed in real-time. Samples captured is conditioned and the signals viewed on an in-built scope. A signal generator interface allows user to select among variety of waveforms like sine, square, sawtooth and triangle with desired amplitude and frequency. Stage calibration factors and PI controller gains can be set. A nominal reference position can also be set in real-time. Likewise, AOD control and calibration factors are user settable. AOD can be run in open loop, closed loop based on the measured bead position or in constant force mode. The GUI also offers setting parameters for various other protocols like AOD scanning (discussed later), sending a chirp signal and capturing the response for transfer function computation and controlling of the stage/AOD using human interface device like a mouse. A screenshot of a sample interface is show in Fig. 2.3

Write parameters to FPGA All the user settable parameters are written to the registers of the FPGA at the beginning of each iteration.

Get data from_FPGA Data captured by the FPGA is sent to an intermediate computer buffer which is read by this block. The host program reads the data when it becomes available. Often, the intermediate buffer is too full and the FPGA cannot put new samples in the buffer. This causes data loss. This may happen when the host computer is slow and/or is performing time consuming operations therefore it is not able to clear the buffer. The program therefore has two modes of operation, in normal mode when no data logging is required, the buffer is forcefully cleared before the beginning of every new data acquisition cycle. This is not desirable when logging data because data is lost every time. Therefore the buffer is not cleared in data logging mode and no other time consuming task is being performed in this mode. However, occasional data loss may still occur because of the computer's deterministic operation or other operating system's background processes taking up resources.

Stream data to FPGA The user generated waveforms are sent to an intermediate buffer on a DMA (direct memory access) channel from where the FPGA can retrieve the data. User can choose where to send this reference data. It can be one of the axis of the piezo stage or one of the axis of the trap. The waveforms form the reference to the controllers inside the FPGA. The user generated waveforms may also be automatically set by protocols for frequency response measurement and AOD scanning.

Data logging Data logging requires continuous acquisition to the computer memory and writing to a hard disk. In order to increase the efficiency of this process, the data is written in binary format instead of text format. This enables continuous data logging at over 50kHz sampling rate. User can easily choose which signals need to be logged.

Calibration Several calibration methods are there as described in more detail in later sections. The most routinely used calibration routine is to fit the trapped bead's position power spectrum with a Lorentzian curve and from there find the trap stiffness and linear position sensitivity. This is implemented in the host program. A novel, chirp based calibration method is also implemented. A sine sweep is given to AOD by selecting chirp signal in the signal generator and the corresponding response is captured. A LabVIEW built in transfer function block is used to get the frequency response and a custom code to fit the response with a discrete time model is written. The degree of the model transfer function is user given. DC gain of the transfer function gives the sensitivity whereas the cutoff frequency of a first order model (numerator, denominator degree of 2 for discrete time model) gives the trap stiffness. Linear sensitivity can also be found by giving a low frequency sine wave and measuring the amplitude of response. The sensitivity is adjusted until the response matches

the input. This routine is implemented as well. A more complicated sensitivity calibration routine, developed in this dissertation, involves scanning of AOD in two dimensions and fitting a neural network model to the response. The scanning process is automated. The 2d calibration data is sent to another computer where a MATLAB program fits a neural network model and the coefficients of the model are stored in a file. The file contents are read by the host program at the click of a button in the GUI.

2.2.2 Target

Target programming is much more challenging than host programming. Host programming is easy because resources are not a real limitation. Any complicated task can be performed at the cost of time. However in FPGA, all tasks must be performed in one hardware clock cycle or else it will lead to errors. Therefore complicated tasks much be split into smaller tasks each of which can be performed in one cycle. While the National Instrument's FPGA module takes care of several of these issues, custom designs tend to become large and complicated and hence care must be taking in the programming architecture to write an optimized code. The blocks are self-explanatory in Figure 2.2 therefore only the optimization techniques will be discussed in this section. A screenshot of the target program block diagram is shown in Figure 2.4 which shows the complexity of coding required for an integrated operating environment. Many of the blocks mask custom written subroutines that are not shown here.

Scale by power of 2 Any computation that needs multiplication or division by a number that is a power of 2 is easily performed by bit shift operations. E.g., 3×16 is same as shifting the bits of the integer representation of 3 by 4 places to the left, i.e., 00000011 becomes 00110000. Likewise division by a number that is a power of two is realized by left bit shift operation.

Fractional multiplication To perform more accurate multiplication of two numbers with fractional decimal representation, the numbers are first scaled by a large number that is a power of 2 using the previous rule and then multiplied in integer representation and the final result is scaled back by the correct amount. This is similar to fixed point calculation except that the fixed point can be changed by the user without reprogramming the hardware. Also, it only improves accuracy of the integer portion of the final answer, the decimal parts are lost. This is easily rectified by working in units where significant number of digits are in integer part and the decimal part can be ignored.

Division General division should be avoided as much as possible. In our implementation there are no division blocks. However one could use integer division block provided by LabVIEW and by utilizing the scaling factor described earlier.

Pipelining A sequence of combinatorial operations where the output of one operation is the input to the next operation becomes increasing costly for hardware level computations due to timing constrains. The compiler may fail to validate the sequence of operations in one hardware clock cycle. In order to resolve this problem, pipelining is often employed. In this technique, each operation runs parallel on hardware however the result of first operation is passed on to the next block in 2nd clock cycle and so on. Therefore as many cycles are required to get a valid final answer as there are operating blocks. Therefore the final answer arrives after a delay. This is acceptable because the clock cycle of the FPGA runs at 40 MHz whereas desired loop rate is about 100 kHz.

Neural Network Neural network is easily implemented in FPGA by using memory elements, addition and multiplication blocks and a saturation block. All the coefficients of the neural networks are stored in memory blocks which are user configurable during run-time by switching the memory blocks to read from an available register. A combinatorial code corresponding to a single neuron is formed and this neuron is reused (muxed) for different signals. Nonlinear block in neurons is chosen to be a symmetric saturating block. For hardware purpose this is plain saturation which is not costly compared to other functions. Other functions can be implemented efficiently by using an interpolating lookup table but accuracy could be an issue in such implementations. The output of the neuron for each iteration is stored and finally summed together (output layer of neural network is basically summing neuron). The floating point coefficients are scaled by a large factor of 2 and the final answer is then scaled back.

PI controller Standard PI control architecture is used in our implementations. However to improve the efficiency of computations each addition or multiplication is done in parallel in a single cycle. Pipelining is utilized to realize a sequence of operations. The controller gains are provided from the host are pre-scaled by a number that is a power of 2. The control signal is scaled back by this number for the final output.

Lowpass filters Several low pass filters are required to be present for signal conditioning or else the controller response may be undesirable resulting in jittery performance and large control actions that may harm the instrument. However low pass filters could be expensive block computationally. A restrictive but computationally efficient version of low pass filters is easily realized by using a structure similar to integral

controller. Integrating action ensures that the steady state error between input and output is zero whereas the high frequency noise is attenuated. The integral gain is chosen to be a power of 2. Therefore using only addition, subtraction and bit-shift operation a low-pass filter is realized, however with limited flexibility.

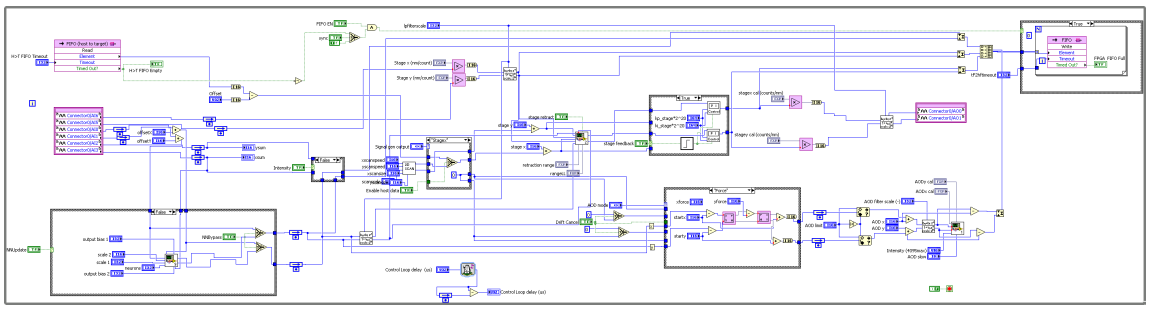


Figure 2.4: Target program in LabVIEW. Many of the blocks are nested subroutines.

2.3 System Calibration

Calibration of the optical tweezers involves finding conversion factors for various equipments within the setup. The factors fall into two categories - sensing calibration and actuation calibration. Sensing calibration numbers convert raw output of sensor signals into physically more relevant meaningful quantities. Actuation calibration numbers convert meaningful numbers into a format understandable to the hardware (actuator). The various instruments that need one-time calibration are, camera, AOD and piezo stage. Detection system (photodiode) typically need calibration before the beginning of every experiment because it depends on the sample itself (bead size in particular). This calibration gives the conversion factors for converting photodiode signals into trapped microsphere's position. For optical tweezers, calibration also refers to obtaining the stiffness of optical trap. The camera is calibrated by imaging a slide-rule that has engraved grid lines at known spacing. Calibration factor with units of nm/pixel is obtained by dividing the known grid distance by the number of pixels that span that distance on the camera. AOD calibration depends on the camera calibration. For AOD calibration, trapping laser spot is focused on a plain coverslip with smallest spot. The image of the spot is observed on the camera and then AOD is given two frequencies alternately at high frequency. On the camera image, two spots are observed. The difference between the frequencies is adjusted so they are spaced apart by a fixed distance (say 50 μm) as measured by the camera calibration. Thus AOD calibration factor is obtained with units, MHz/nm. The control signals are typically in nm therefore another calibration factor is required (with units of nm/MHz) to convert control signal (with units of nm) into AOD specific signal (with units of MHz). This number

is provided by the manufacturer. The two numbers are merged into one by multiplying them so that effectively the units are nm/nm where the numerator refers to the control signal units and denominator refers to image plane units. Piezo calibration numbers is usually provided by the manufacturer or the camera is used again for piezo stage and sensor (LVDT) calibration. A sample is imaged and an arbitrary amount of signal is given to the piezo stage. The amount of movement of piezo stage is measured using the camera and the corresponding change in the LVDT signal is measured. The ratio of the two numbers (nm/Volts) gives LVDT calibration factor. Piezo is inherently a nonlinear device therefore it is always operated in closed loop mode. Therefore the control signals are computed by a controller in which case there is no need for piezo's actuator calibration number.

Detection system calibration is not straightforward and requires some strong assumptions about the system. To this end several techniques exist in the literature that are briefly discussed here.

Thermal response A bead in fluid experiences random forces due to the collision of fluid molecules with the bead. Such forces are well studied and are characterized by a white noise statistics with a power spectral density of $4k_BT\gamma$ where k_B is the Boltzmann's constant, T is the absolute temperature and γ is the damping coefficient. From Stoke's law, $\gamma = 6\pi\eta r$, where η is the viscosity of the fluid and r is radius of the bead. Fluid viscosity is not accurately known especially when assays include a mixture of several reagents. Typically, the viscosity value of water is used. r is set to the mean radius of beads as provided by the manufacturer. The white thermal forces result in position fluctuations that is given by $S_x = \frac{4k_BT\gamma}{4\pi^2\gamma^2(f^2+f_c^2)}$ nm^2/Hz that is a filtered version of the white noise where the filter is a first order low pass filter with a cutoff frequency of $f_c = \frac{k}{2\pi\gamma}$ Hz, where k is the trap stiffness. To obtain stiffness and photodiode calibration numbers, power spectrum of photodiode signals corresponding to the fluctuations of the bead are recorded. A Lorentzian $S_v = \left(\frac{a}{f^2+f_c^2}\right)$ volts^2/ Hz^2 is fit to the spectrum to obtain a and f_c. Hence $k = 2\pi\gamma f_c$. Position calibration factor is given by $\sqrt{\frac{k_BT}{\pi^2\gamma a}}$ nm/volts. This factor is valid for about 150 nm about the center of the trap after which the photodiode voltage varies nonlinearly with the bead position. The advantage of this method is that it gives both stiffness and position calibration factors simultaneously and doesn't need a separate detection beam.

Drag force In this method only the stiffness of the trap is measured and assumes calibrated position measurement to be available, e.g. camera based bead position measurement. In this method, the sample is oscillated in triangular waveform so that the sample's velocity profile is a positive constant (v) during one half of the cycle and negative constant ($-v$) during the other half. During this time the trap is kept

stationary. Moving sample creates a drag force on the stationary bead that is given by γv. The drag force is balanced by the trapping force, kx, where x is the bead displacement from its nominal central position in absence of drag force. x is assumed to be known by other means and hence, $k = \frac{\gamma v}{x}$. This experiment is repeated for several velocities (v) and a (non)linear fit is obtained for k . Several cycles worth data is collected to average out the noise.

Equipartition In equilibrium thermodynamics, equipartition theorem states that every degree of freedom of a particle/system in thermal equilibrium dissipates $\frac{1}{2}k_BT$ of thermal energy through mechanical energy on an average. For optical tweezers, kinetic energy is ignored due to the particle's negligible mass compared to the mean potential energy of the trap which is given by $\frac{1}{2}k\left\langle x^2\right\rangle$. $\left\langle x^2\right\rangle$ is the variance of the bead position assuming calibrated position measurements are available. From equipartition theorem, $\frac{1}{2}k\left\langle x^2\right\rangle = \frac{1}{2}k_BT$. Hence the stiffness parameter is estimated to be $k = \frac{k_BT}{\left\langle x^2\right\rangle}$

Stuck bead scan A bead stuck to the coverslip surface may be used to calibrated photodiode signals and convert it into position signal. The sample is moved by a known distance using a calibrated piezo stage. The response obtained on the photodiode for every position of the bead is recorded and a map is obtained. Typically linear map is assumed for a small range along one axis. In general, for two dimensional applications, nonlinear map must be obtained. This method is particularly sensitive to the axial position of the bead with respect to the detection beam focus. For accurate calibration, care must be taken to make sure that the bead height offset with respect to the detection beam focus is same as when the bead is in trap.

Trapped bead scan[3] Another way of obtaining calibrating factor for photodiode is to scan a trapped bead using AOD. AOD is assumed to be calibrated therefore the bead moves by the amount commanded by the trap position except the disturbance due to thermal fluctuations. Trap stiffness is kept high to reduce thermal fluctuations and increase stiffness. A (non)linear map may be obtained to convert photodiode position signals and corresponding bead position. Such a procedure is explained in detail in the following section that uses novel approach to increasing detection range using neural networks and is implemented in real-time unlike other methods.

RLS Recursive least squares based estimation of trap stiffness parameter. A trapped bead is moved in a profile that has a mixture of sinusoidal frequencies. The response is measured and the trap parameters are estimated in real-time using RLS method. Details of this method can be found in [4]

Higher order corrections Power spectrum of trap bead is assumed to be Lorentzian, however its not completely true because of higher order effects playing a role. These are the effects due to change in damping coefficient of liquid close to the surface, hydrodynamic effects, change in trapping potential itself and finite sampling effect that results in an aliased Lorentzian. A comprehensive analysis of power spectrum for optical tweezers is provided in [5]

Frequency sweep[6] A trapped bead is moved in chirp profile, which is a sinusoidal motion with the frequency increasing with time. By measuring the magnitude and phase for various frequencies the transfer function (magnitude and phase plot with respect to frequency) is measured and fit to a low order model. The cutoff frequency is estimated from the transfer function. The trap parameters are then estimated from the cutoff frequency. The advantage of this method compared to thermal response is that large amplitude sine waves can be given therefore the effects of noise can be reduced significantly. Also the range of linearity can be checked.

2.4 Increasing detection range

In the bead position detection system, forward scattered light from the bead is collected on to a position sensitive or quadrant photodiode that provide signals that depend on the position of the bead relative to the trap center [7]. A variant of this method uses a separate laser beam of very low power that serves as a reference for all measurements. This decouples force actuation from force sensing and enables controlled experiments with position or force feedback [8]. Such a setup has been used to study various motor proteins like kinesin, a protein that walks on microtubules[9]. Many studies of motor proteins use optical tweezers under constant force mode, where the separation between trap center and bead center is regulated at a constant value. One of the limitations in such studies is the short detection range. During experiments, if the bead reaches the limits of detection, then either the entire sample is repositioned to a nominally selected initial position or the bead is forced back, disturbing the experiment midway. An increased linear detection range is therefore desirable. The relationship between photodiode signals and the actual position of the bead is nonlinear [10]. Researchers often use a linear approximation which is typically valid for small deviations about the center. A polynomial fit of intensity normalized photodiode position signals can extend this range by approximately 30% [9]. In this thesis a new method to process photodiode signals will be developed that provides accurate position measurement for a much larger range than the existing schemes. The method is based on neural network mapping of voltage signals to positional signals. Nonlinearity compensation

techniques based on two dimensional polynomial fit [9] and neural networks [11] exist but they are limited by the fact that the map from voltage to position is not one to one for a large range [9]. In this thesis, the domain in which voltage to position mapping is not one-to-many (feasible domain) is estimated and mapped to position signals. An order of increase in detection range was previously demonstrated by [12] where a maximum likelihood estimator was used. However, the method is not practical for real-time implementation that is necessary in feedback based experiments. Also the results were valid for one dimension only. The method developed in this paper uses neural network model instead which is easier to implement on hardware, works in two dimensions and preserves accuracy over a larger range.

2.4.1 Calibration and detection range

Calibration data for voltage to position conversion is obtained by moving the trapped bead in a square grid, $1\,\mu\text{m} \times 1\,\mu\text{m}$ in size with 10 nm spacing between adjacent points, using AOD centered about the detector beam. The scanned locations are denoted by (x, y). Photodiode signals, V_x, V_y and V_z are sensitive to motion of the bead along x, y and z coordinates respectively with strong cross-coupling as the bead moves further away from the center of the detection region. (V_x, V_y, V_z) is captured for every grid point. During scanning, the trap stiffness is kept sufficiently high and the scanning speed sufficiently low to ensure the bead is always equilibrated at the trap center and the effect of thermal noise is low. The calibration data represents a map $f_{xv} : (x, y) \rightarrow (V_x, V_y, V_z)$ from position to voltage. The objective is to obtain the map, $f_{xv}^{-1} : (V_x, V_y, V_z) \rightarrow (x, y)$ to deduce the position from voltage. Traditional schemes do not have V_z as a part of calibration data so their map has the form $f_{xv}^{'} : (x, y) \rightarrow (V_x, V_y)$. Inversion of $f_{xv}^{'}$ over entire domain is not possible as it is not one-to-one. However, if on a restricted domain $f_{xv}^{'}$ is one-to-one then inversion is possible. To find the restricted domain, the following analysis is done based on calibration data. A sample calibration data is shown in the form of contour plots of V_x, V_y (Figure 2.5(b)) and V_z (Figure 2.5(a)). Contour plots for V_x or V_y are overlaid for clarity that shows lines/contours joining points on x-y plane along which the voltage has a fixed value. The difference between the voltages of two adjacent contours is kept uniform. Contour lines of V_x intersects with those of V_y. If a pair of contour lines, one for V_x and the other for V_y intersect at more than a single point then it implies that the voltage pair corresponds to more than one location on the x-y plane. Of all such locations that give the same voltage pair, only the one closest to the center is considered a part of the restricted domain or the detection region. Thereby, estimating the shape of the contour plots, the boundary of detection region is deduced by the set of points where a pair contour plots

are tangent to each other. Similar logic explains that f_{xv} is invertible over a larger domain because now three contour lines must intersect simultaneously at multiple points to be out of the domain. This happens approximately after 500 nm beyond which the intensity signal doesn't change significantly. Therefore, no static map can increase the detection range beyond 500 nm (for a similar setup as in this thesis). Ideally, it would be desirable to have straight and perpendicular contour lines, i.e., f'_{xv} be linear. Although, this is not true, it holds well till approximately 150 nm from the center of the detection region. This limitation can be relaxed if a nonlinear function, e.g., a polynomial is used for f_{xv}^{-1}. This would be close to 250 nm if f'_{xv} is inverted for the kind of data presented here. However, a 5th order polynomial (as implemented in [9]) performs well only for a smaller range less than 200 nm (data not shown). In this thesis, f_{xv} is inverted instead and as explained before it has larger invertible domain. Neural networks are employed to do the inversion resulting in larger detection range. The calibration data is used to train the neural network as explained next.

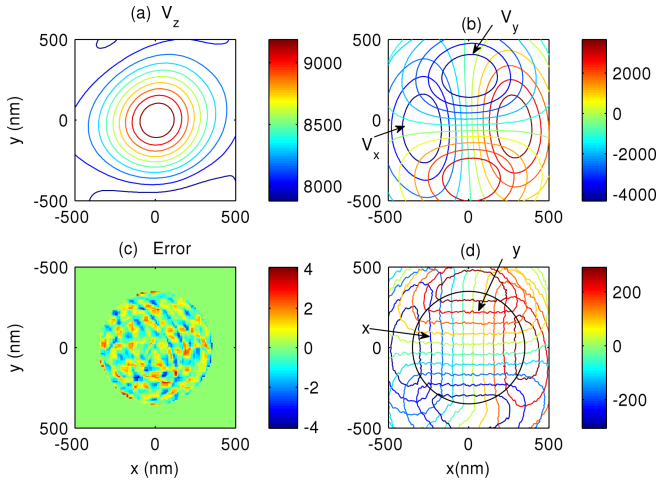


Figure 2.5: (a) Calibration data, Contour plot for intensity signal. (b) Calibration data, Contour plots for V_x and V_y. (c) Error plot between actual x position values and that estimated from voltage signals using a neural network. (d) Contour plots for estimated position values experimentally obtained from voltage signals. Noise in the data is partly due to thermal noise and partly due to AOD nonlinearity which is position dependent. Perpendicular intersections of the contour plots within the indicated circle of radius 350 nm indicates absence of cross-coupling.

2.4.2 Neural network mapping

A neural network consists of layers of interconnected neurons as in Figure 2.6(b). Figure 2.6(a) shows the model of a neuron that has several inputs and one output. The neuron

computes the weighted sum of inputs, add a constant (bias) and passes the result to a non-linear function (also known as activation function). In this thesis, the network consists of 3 input nodes, one input layer with 50 neurons and one output layer with 2 neurons and 2 output nodes. All the input layer neurons have symmetric saturating linear activation function ('satlins ', see Figure 2.6(a)) while the output layer neurons have pure linear activation function ($f(x) = x$). Satlins is chosen for its ease of implementation. From hardware point of view, satlins is simply a saturation filter. Other complex functions can be chosen using a look-up table but they are sensitive to discretization errors. The equations describing the action of neural network is as follows (using notation as in Figure 2.6)

$$\begin{aligned} x &= c_1 + \sum_{k=1}^{n} \text{satlins} \left\{ \sum_{i \in \{x,y,z\}} V_i W_{ik} + b_k \right\} W_{k1} \\ y &= c_2 + \sum_{k=1}^{n} \text{satlins} \left\{ \sum_{i \in \{x,y,z\}} V_i W_{ik} + b_k \right\} W_{k2} \end{aligned}$$

The inputs to the neural network are (V_x, V_y, V_z) and the desired outputs (targets) are the (x, y) obtained from the calibration data. The objective is to find the network weights and biases so that the neural network maps inputs to the targets. A two dimensional polynomial fit (position to voltage) is used to smooth out the noise in the calibration data due to Brownian motion of the bead and nonlinearities in AOD. Smoothed version of voltage values are used for the subsequent steps. Scan locations that are within the circle with radius of about 350 nm from the center of scan are used for training. Mapping a larger requires bigger network that translates into slower computation. Therefore there is a trade-off between bandwidth and range. The training is done in MATLAB using its Neural Network Toolbox. The FPGA hardware does not support floating point operations and a fixed point implementation is undertaken where the weights and biases are scaled by a large number $\left(2^{20}\right)$ and the remaining fractional part is rounded off. After doing integer calculations, the result is scaled back. This process does not significantly affect the accuracy for the chosen architecture. The computational delay is less than 20μs. Remaining operations, like data acquisition, control logic and AOD control are done in parallel to the above operations also take less than 20μs. There is however, a delay of a couple of sampling cycles from sampled data to the computed control signal. The number of neurons, network weights and biases are programmable during run-time and thus recompilation of the code is not required. Total number of multiplications required for 50 input layer neurons and 2 output layer neurons is 300 and memory requirement is of the order of 300 elements, whereas for a 5th order polynomial in three variables, this estimate will

be about 600 for multiplications and 240 elements in memory. Another aspect for not recommending polynomial computation within hardware is evaluating power and successive multiplications which accumulates errors if implemented in fixed point. Method suggested by [12] requires the use of a parametric map, which is basically a form of calibration data. A data with 100 grid points will have 10,000 memory elements, therefore hardware implementation will require storing these many elements which again is quite impractical considering other tasks that the hardware has to perform and a smaller number of grid points will lead to interpolation errors where nonlinearity is severe. Secondly, complex algorithm like maximum likelihood estimator will have huge overheads of its own.

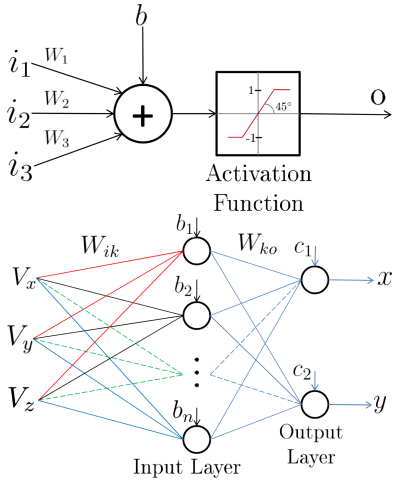


Figure 2.6: (a)Neuron Model. Activation Function shown is called 'satlins' for symmetric saturated linear function. i_k are the inputs to the neuron. W_k are weights associated with the k_{th} input. b is the bias associated with the neuron and O is the output of the neuron.(b)Neural Network Model. Circles represent neurons. Inputs V_x, V_y, V_z are inputs to the n neurons of input layer weighted by W_{ik} from i_{th} input to the k_{th} neuron. Biases for each neuron as shown by b_k. Likewise, output layer weights and biases are shown by W_{ko} and c_k respectively. x and y are the outputs of the network.

2.4.3 Results and discussion

The nonlinear mapping capability of neural networks is demonstrated in Figure 2.5(b). In the region indicated by a circle of radius 350 nm, the output of the network matches with the scanning location with excellent accuracy (approximately 1 nm rms and 4 nm peak-to-peak). Experimental data taken after calibration for a scanning experiment is shown in Figure 2.5(a). Time data sample for similar experiment is shown in Figure 2.7.

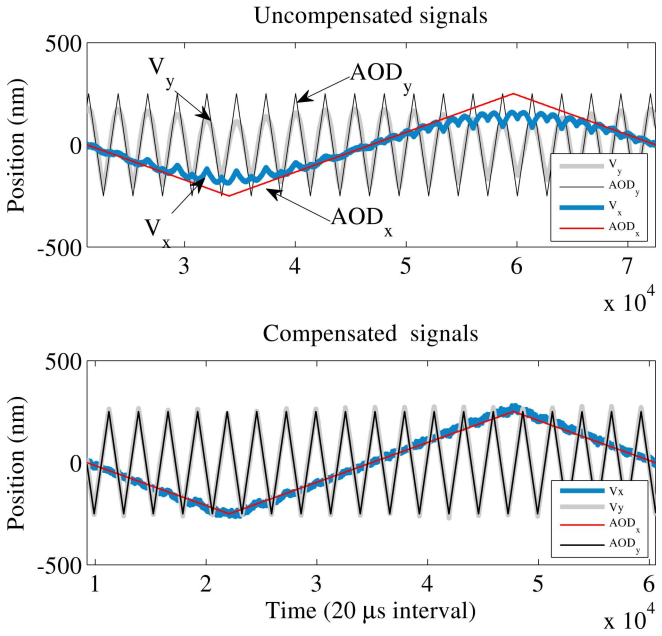


Figure 2.7: Time plot of signals during scanning of size 500 nm peak to peak obtained from hardware. AOD_x and AOD_y are the input signals. V_x and V_y are the corresponding outputs. Bead reaches approximately 350 nm away from the center at certain times. (a) Unfiltered signal with highly nonlinear response (Voltage signals are scaled and shifted for comparison). (b) Position signals obtained after passing voltage signals through neural network with no scaling or shifting Residual nonlinearity in the V_y signal is due to the AOD.

In summary, a convenient way to process photodiode signals is introduced that compensates detection nonlinearity for a larger range and is also suited for real-time implementation that is necessary for feedback based experiments. It is also an effective method to remove cross coupling between x and y position signals. The calibration process is quick enough to be used on a regular basis.

Chapter 3

DNA Techniques

3.1 Literature

A single strand of DNA is a polymer of long chain of four types of Nucleotides or bases commonly abbreviated as A, T, G, C that stand for Adenine, Thymine, Guanine and Cytosine respectively. In nature, there is a complementary strand of DNA that binds to the original single strand through hydrogen bonds to form the well known double helix structure. The complementarity is governed by the A-T, G-C rule i.e., A bonds to T while G bonds to C. Thus if a single strand of DNA goes like AAGTCATGTCGTA, then complementary strand will read TTCAGTACAGCAT. Single molecule manipulation capability of optical tweezers has proved to be a useful tool in understanding the dynamics of individual DNA molecules complementing the data from ensemble averages.

Optical trap was first used for studying DNA in [13, 14, 15] where effect of hydrodynamic drag on the extension of a single fluorescent marker labeled DNA molecule was studied. Much closer study was done by [16] by pulling DNA by its ends using a optical trap and a micropipette. The tether formation involved capturing a streptavidin coated bead and subsequently sucking it onto the tip of a micropipette. The buffer contained biotin-end-labeled double stranded DNA (dsDNA) that got stuck to the bead and extended due to the flow in the chamber. Another similar bead is caught in the optical trap and brought close to the DNA end (the end is not visible, it is 'fished' for tether). Tether formation is confirmed if the micropipette pulls the bead upstream of flow. They showed that under a longitudinal stress of about 65 pN, a dsDNA molecule undergoes a cooperative transition that results in elongation of DNA by 70%. This transition was attributed to unwinding of the DNA into parallel ladder form and subsequently fraying due the presence of nicks. This transition was also affected by the ionic concentration of the buffer. [17] studied the effect of temperature on this transition to find that increase in temperature lowers the force

required to bring the transition.

Article [18] introduced feedback control on the laser power to control the trap stiffness in realtime based on position data sensed by a quadrant photodiode. Ability to control trap stiffness helped to keep the bead in the linear region of position detection as well as study wider range of force. Small DNA fragments or coiled state of DNA have low elasticity and exert small forces on the bead and if the trap stiffness is high then the corresponding bead motion will be smaller leading to low resolution. A low stiffness trap was employed for low force regime of an experiment and as the force magnitude increased the bead got pulled. When the bead reached a set distance from the center of the trap, a position clamping feedback control is activated where the intensity of trap was increased with external force on the bead such that the bead remained stationary. The change in trap stiffness is then a measure of external force. Laser power was controlled using an Acousto-Optic Modulator. This method is applicable for high force regime and thus large range of force profile can be measured in a single experiment.

[19] studied the effect of ionic strength on the elastic properties of a lambda-DNA. They showed that the persistence length, which is a parameter in Worm-Like-Chain(WLC) model of DNA, varies inversely with the ionic strength and so does the stretch modulus(force/strain). Monovalent and multivalent ions had different effect on the persistence length of DNA. Multivalent ions resulted in smaller DNA persistence length. It was known that multivalent cations are capable of DNA condensation (molecule shrinks into compact mass). By stretching DNA, they could prevent condensation but even then there were retractile forces indicative of slight condensation. They proposed 'thermal rachet' mechanism where thermal forcing introduces temporary slack enabling the molecules to form loops and condense. A side-by-side association prevents DNA from reversing the action thus ratcheting the cycle. The DNA condensation behavior was further investigated in 2000 by [20] where they used multivalent cations such as spermidine and hexaammineCobalt(III) to get DNA into toroidal condensed state under no external force. The condensed structure is similar to the DNA packaged inside the nuclei of a bacteriophage. It was shown that elasticities of DNA is different in presence of these ion. Also, there are formation of loops (nulceations) that leads to collapse of of a part of DNA into compact structure.

Another interesting application of optical tweezers on DNA is to estimate its sequence of basepairs. To this direction, work had been done using Atomic Force Microscope (AFM) and microneedles but the first use of optical tweezers for this purpose was done in 2002 by [21] in which they labeled the same extremity of a dsDNA molecule with biotin-labeled-oligonucleotide on one strand and digoxigenin-labeled-oligonucleotide on the the other strand. Biotin binds to streptavidin coated silica beads whereas digoxigenin binds with

anti-digoxigenin coated onto the glass slide. The silica bead was trapped with optical tweezers while the slide was pulled so that the two strands started opening up (unzip). By monitoring the position of the trapped bead, unzipping force was determined. The force profile had several 'force flips' indicative of bond breakage and subsequent relaxation in the tension. The force profile was shown to be sequence dependent and it could be used to estimate the content of A-T or G-C bonds. The reannealing/rezipping process initiated by relaxing the unzipped DNA showed discrete steps in the force signals that seemed to be dependent on the sequence of DNA. This reproducible phenomenon was attributed to formation of transient secondary structure.

3.2 Materials and methods

3.2.1 DNA Labeling

The protocol discussed in the chapter is a unique protocol adapted from [22] applicable to easily available lambda-DNA. It is simple in concept and easy to implement. The aim of the protocol is to attach biotin on one end and digoxigenin on the other end of lambda-DNA. This is a two step process - in the first step lambda DNA is mixed with biotin-11-dUTP, dATP, dGTP and klenow exo^-. This will incorporate biotin on near one end of lambda DNA. In the second step, digoxigenin is incorporated on the other end by mixing the biotin modified DNA with digoxigenin-11-dUTP, dATP, dGTP, dCTP and klenow to label the other end of the DNA with digoxigenin. The step by step protocol is provided here:

Reagents and Instruments required:

1. λ-DNA (New England Biolabs), stock concentration: 500μg/ml
2. Individual sets of dATP, dGTP, dCTP (Bioron GmbH), stock concentration: 10mM, dilute this to 1mM
3. Biotin-11-DUTP (Bioron GmbH), stock concentration 1mM
4. Digoxigenin-11-dUTP (Roche Diagnostics), stock concentration 1mM
5. Klenow exo^- fragment (New England Biolabs), stock concentration 5000 units/ml, 10X buffer supplied.
6. Reagents and buffers: 1M Tris-HCl(pH 8), 0.5M EDTA(pH 8), PBS (1M), 5M NaCl, Nuclease free / dionized water, BSA (All from Ambion)
7. Streptavidin coated microspheres, Antidigoxigenin coated microspheres (Spherotech)

8. Microcon-10 spin filters for DNA (Millipore)

Protocol:

1. Add the following in order:

Item	Quantity (μL)	Target
Water	9	
DNA (500μg/ml)	4	2μg
dATP (1mM)	2	100μM
dGTP (1mM)	2	100μM
Bio-11-dUTP (1mM)	2	100μM
Klenow-exo^-	1	
Total	20	

2. Incubate the above reaction mix for 1 hour at room temperature. Note, the enzyme (Klenow fragment) should be kept on ice all the time or in freezer unless required.

3. After incubation heat the mix at 75° for 10 minutes (to stop enzyme activity)

4. Dilute the mix in a total of 500μL TE buffer (10mM Tris-HCL, 1mM EDTA - pH 8) and spin filter in Microcon-10 as per manufacturer's protocol

5. Repeat the above wash times 3-4 times to get rid of unincorporated nucleotides and enzyme. At the end of filtering step, you should be left with about 5-15 μL of DNA solution. Adjust the water in the following step accordingly.

6. Add the following

Item	Quantity (μL)	Target
Water	3	
DNA (from prev. step)	10	
dATP (1mM)	2	100μM
dGTP (1mM)	2	100μM
Digoxigen-11-dUTP (1mM)	2	100μM
Klenow-exo^-	1	
Total	20	

7. Incubate the above reaction mix for 1 hour at room temperature. Note, the enzyme (Klenow fragment) should be kept on ice all the time or in freezer unless required.

8. After incubation heat the mix at 75° for 10 minutes (to stop enzyme activity)

9. Dilute the mix in a total of 500 μL TE buffer (10mM Tris-HCl, 1mM EDTA - pH 8) and spin filter in Microcon-10 as per manufacturer's protocol

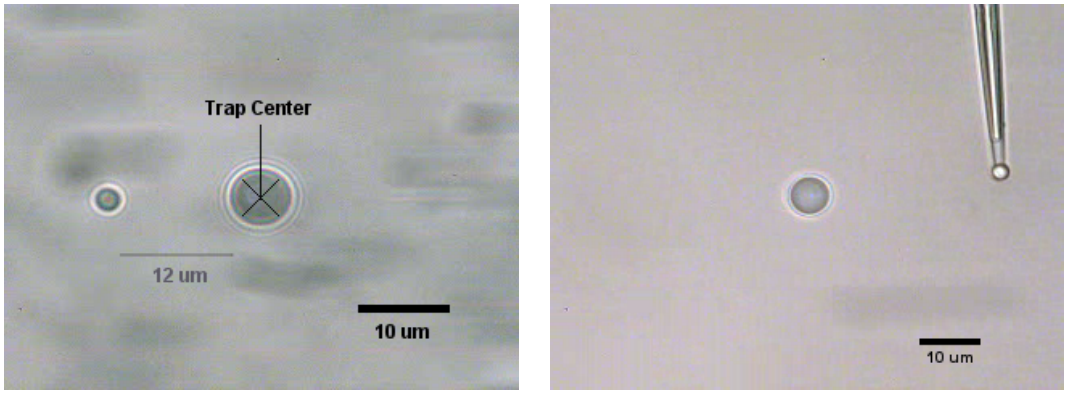
10. Repeat the above wash times 3-4 times to get rid of unincorporated nucleotides and enzyme. At the end of filtering step, you should be left with about 5-15 μL of DNA solution.

11. Dilute the DNA in a total of about 100 μL of TE Buffer.

Preparation of Microspheres: The stock solution of coated microspheres need to be washed to removed any free labels floating in the solution otherwise they will bind to DNA leading to inefficient tethering with the bead. To do that take 200 μLof stock solution of microsphere solution (0.5% w/v, 2.1μ m diameter) and centrifuge it at 10,000g for 5 minutes. Decant the supernatant and resuspend the pellet in 200 μL of PBS buffer with 0.1mg/ml of BSA (to avoid nonspecific adhesion) and 1M NaCl.

***In Situ* Tethering:**

Once the DNA is labeled the next step is to obtain a tether such that one DNA molecule is attached to two differently coated beads at its ends. First, streptavidin coated bead solution and DNA solution is mixed such that the stoichiometry is about 30 DNA molecules per bead. Let the mixture incubate for about 30 minutes. As an example, mix 10 μL of 0.5% w/v of streptavidin coated beads with 4 μL of 20μg/mL of lambda-DNA solution to obtain 30:1 ratio. After incubation, add anti-dig coated beads and load this solution into syringe tube for introducing into a flow cell. Inside flow cell, introduce a flow of this solution and first optically trap a streptavidin coated bead and transfer it onto a micropipette tip by sucking through the tip. Next, optically trap a antidig coated bead and relocate the streptavidin coated bead(on the micropipette tip) to 10-16 μm downstream of antidig coated bead and 'fish' for the free end of DNA stretching out from the streptavidin coated bead. If a successful tether is made then the antidig coated bead will be pulled out of the optical trap (provided the laser power is low enough) against the flow direction that confirms a successful tether. More information on stoiciometric ratios and buffer conditions can be referred from [23] that can be a useful guide in selecting appropriate salt concentrations and pH of the solution. Figure 3.1 shows tether formation by introducing a flow and trapping antidig coated bead. The streptavidin coated bead is at a fixed distance from the trapped bead while rest of the beads in the solution are moving with the flow.



(a) (b)

Figure 3.1: (a) Larger bead (4.26μm in diameter, anti-digoxigenin coated) is optically trapped and tethered through DNA(16μm long) to smaller bead(2.1μm in diameter, streptavidin coated) inside a flow chamber (b) Micropipette held bead and optically trapped bead shown together inside a flow chamber(there is no tether here)

In another setup, DNA stretching can be done without the need of micropipette. In this case the two beads are trapped in two independent traps. Alternatively, two traps may be generated from a single beam by time multiplexing them, i.e., by switching between two trap positions at much faster rate compared to the diffusion rates of either bead. This essentially creates two virtually independent traps. This method doesn't require formation of special flow channels and the trapped beads lie in the same plane. With micropipettes, one is restricted to work in the plane of micropipette which may not be the optimal location for optical trap. This setup is used in the quantitative experiments described in later sections.

3.3 Persistence length estimation

A typical DNA molecule is composed of two long chains of molecules. They are coiled around each other to form a double helical structure. λ-DNA is obtained from a virus that attacks bacteria. It has a contour length, L of about 16.4μm long. In an aqueous environment a free molecule of DNA assumes arbitrary conformations driven by thermal forces such that on an average the end to end distance of the molecule is zero. However if the molecule is stretched, the number of conformations that DNA can assume is limited to those whose end to end distance is equal to the extension, z of the molecule. This results in reduced entropy of the DNA-system and it tries to go back to its random free state giving rise to an entropic force, F. This force is balanced by the external force that is stretching the DNA. Following equation describes the force-extension behavior of such polymeric molecules and is commonly known as worm-like chain model. This equation holds for an average force and extension in a thermal bath with temperature T.

$$F = \frac{K_B T}{P}\left[\frac{1}{4}\left(1-\frac{z}{L}\right)^{-2} - \frac{1}{4} + \frac{z}{L}\right] \tag{3.1}$$

Where F is the average force on DNA, K_B is the Boltzmann's constant, T is temperature in Kelvin, z is extension of DNA and L is the contour length of DNA and the quantity P is known as the persistence length. This parameter is a property of the polymer that represents its rigidity. If persistence length is small compared to contour length then it means the polymer is very flexible. On the other hand if the persistence length is comparable to the contour length then it means that the polymer is rigid like a rod. Another interpretation of P is that it is the average projection of end to end vector of the polymer in a thermal bath along the initial segment of the polymer where it is held fixed.

Persistence length is an important physical parameter for biological macromolecules like DNA and other proteins. This helps understanding in part the physical aspects of functioning of these molecules, e.g. how an extremely long DNA is packaged inside a small nucleus of a cell or how protein molecules interact with each other to carry out life processes. In this regard Optical Tweezers is a novel instrument that is capable of studying such molecules individually in isolation. Optical tweezers is a setup that can trap micron sized beads and measure its position accurately (sub-nm resolution). It can also be calibrated to measure forces based on the displacement of the bead from the center of the trap. To study a biomolecule, it is tethered between two beads, one of which is held stationary with respect to the microscope slide and other is held in the optical trap. The stationary bead is controlled by a nanopositioning stage or another trap. To stretch the biomolecule, the stage is moved away from the trapped bead and the force on the molecule is measured from the displacement of the bead from the center of the trap. Figure describes the process.

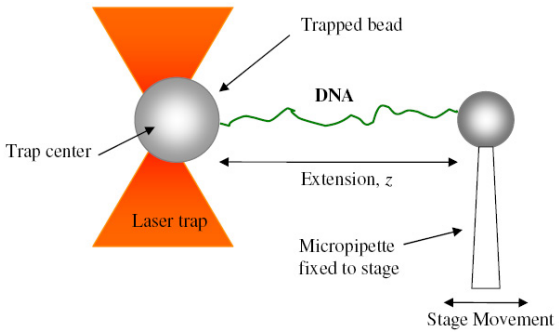


Figure 3.2: Double Trap setup

To estimate the persistence length, the given static model will be cast into least squares

parameter estimation form,

$$y = \theta * \phi$$

where

$$\begin{aligned} y &= F \\ \theta &= \frac{1}{P} \\ \phi &= K_B T \left[\frac{1}{4} \left(1 - \frac{z}{L} \right)^{-2} - \frac{1}{4} + \frac{z}{L} \right] \end{aligned}$$

F is measured from optical trap's spring constant, k and the bead displacement x_b. L is a known quantity (for λ-DNA, $L = 16\mu m$). z is known from the stage movement. T is assumed to be the room temperature (295K) and $K_B = 1.38x10^{-23}$. It should be mentioned here that bead position is also affected by the thermal forcing of the surrounding medium coupled with approximately first order dynamics of the trapped bead. More on this will be discussed later. The discounted least-squares estimation algorithm for scalar signals in discrete time is then given by the following equations:

$$\theta_{n+1} = \theta_n + P_n \phi_n \left(\frac{y_n - \theta_n \phi_n}{\alpha + P_n \phi_n^2} \right); \theta_0 = 0$$

$$P_{n+1} = \frac{1}{\alpha} \left[P - \frac{P^2 \phi^2}{\alpha + P \phi^2} \right]; P_0 = 100$$

When P is scalar then iteration on P is equivalent to:

$$P_{n+1} = \frac{P_n}{\alpha + P_n \phi_n^2}$$

Where, $\alpha \leq 1$ is the discounting factor to track slowly varying parameter, θ.

3.3.1 Simulation Results

Parameter fitting by the above method was tested in simulations assuming force and extensions are measurable quantities following the model 3.1. DNA model incorporates a time varying persistence length. In Figure 3.3(a), persistence length is varied with time while the extension is kept constant at 15500. Discounting factor is fixed at 0.98 chosen as a compromise between accuracy and speed of convergence. In Figure 3.3(b), a constant persistence length is assumed so more accurate and less noisy result is achieved by making discounting factor equal to 1 (i.e., no discounting). In Figure 3.3(c), speed of

convergence can further be improved if a varying extension is given to the DNA. This is related to the persistency of excitation. The results are plotted in Figure 3.3(c) are for $z = [13500 + 2000sin(\pi t)]\, u(8-t) + 40u(t-8)$

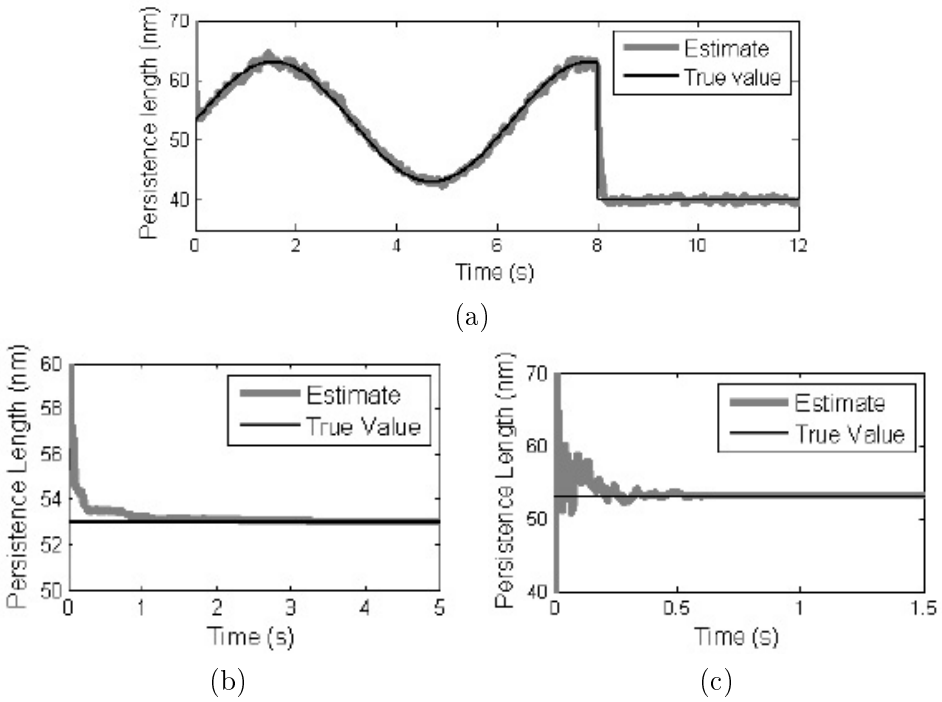


Figure 3.3

3.3.2 Experimental Results

A double trap setup was used to stretch DNA and obtain force v.s extension data. The data was collected and offline fitting was done using RLS method. A hardware version was also used but it did not perform well because of lack of desired precision in integer computation constraint of FPGA. Figure 3.4 shows the result of the fitting procedure. The experimental results do not converge to the true physical parameters of the chosen DNA molecule. This is not the result of poor estimation but due to errors in calibration and probably due to biochemical reasons.

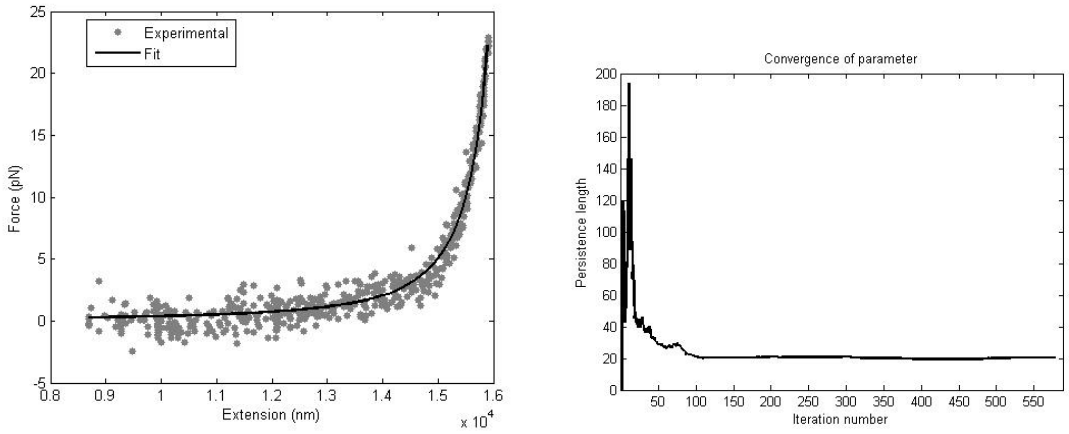


Figure 3.4: Experimental Data (a) Force Extension data, L=16.7μm (b) Convergence to $P = 20 \pm 2nm$

3.3.3 Conclusion

Least squares estimation of parameters offers an online method of estimating system parameters if the unknown parameter is linearly related to the input-output signals or their modified versions. In this project it was revealed that even in presence of noise (thermal and measurement noise) it has an averaging effect on the noise as against the gradient based algorithm which will have same amount of noise in the estimation that the system is excited with. This was verified by simulations but is not reported here. Secondly least squares method finds the optimal fit for parameters when the states of the systems scan the configuration space. This is equivalent to finding a least squares curve fit for a noisy input output graph. The advantage with LS estimation is that it can be implemented in an online fashion.

In this particular project, the output (y_k) was also estimated rather than using the exact value. y_k denotes the tension in the DNA molecule which is estimated from the displacement of the bead and the spring constant of the optical trap. The displacement of the bead in turn changes the extension of the DNA so the actual extension is 'applied extension' minus the bead displacement. The fluctuations in the bead position are however quite small (10 nm) compared to the applied extension (15500 nm) but simulations show that the estimate is quite sensitive to errors in the extension value. Another aspect to consider is that the force vs. extension map holds for average values of force and extension so validity of the results at higher speeds needs to be studied further.

3.4 Coupled sensor system

3.4.1 Effect of Thermal bath on coupled damped oscillators

Let the state space description of two oscillators coupled by a linear spring be given by

$$\begin{aligned} C\dot{X} + KX &= F(t) \\ \dot{X} &= -C^{-1}KX + C^{-1}F(t) \end{aligned} \tag{3.2}$$

Where X is the state vector comprising position of the two oscillator systems, C is the symmetric and positive definite damping matrix, K, the positive definite stiffness matrix and $F(t)$ is the Langevin forcing vector. Matrices, C and K are positive definite matrices therefore invertible. $C^{-1}K$ is also positive definite and therefore diagonalizable by a matrix Q. i.e.,

$$\begin{aligned} & -Q^{-1}C^{-1}KQ &=: \Lambda & \quad \text{(diagonal)} \\ \Rightarrow\ & Q^*KC^{-1}Q^{*-}1 &= -\Lambda & \quad (\because\ K, C^{-1}, \Lambda \text{ are symmetric}) \\ \Rightarrow\ & -C^{-1} &= A\Lambda Q^{-1}K^{-1} & \\ \text{or}\ & -C^{-1} &= K^{-1}Q^{*-1}\Lambda Q^* & \\ \therefore\ & A\Lambda Q^{-1}K^{-1} &= K^{-1}Q^{*-1}\Lambda Q^* & \\ \Rightarrow\ & Q^*KQ\Lambda &= \Lambda Q^*KQ & \\ \Rightarrow\ & K_D\Lambda &= \Lambda K_D & \end{aligned} \tag{3.3}$$

$K_D := Q^*KQ$ is diagonal if Λ has distinct eigenvalues, that is shown below. Note from Eq. 3.3,

$$\begin{aligned} & (K_D\Lambda)_{ij} &= (\Lambda K_D)_{ij} \\ \Rightarrow\ & \textstyle\sum_k K_{Dik}\Lambda_{kj} &= \textstyle\sum_k \Lambda_{ik}K_{Dkj} \\ \Rightarrow\ & K_{Dij}\Lambda_{jj} &= \Lambda_{ii}K_{Dij} \\ \Rightarrow\ & K_{Dij}(\Lambda_{jj} - \Lambda_{ii}) &= 0 \end{aligned} \tag{3.4}$$

Therefore, for $i \neq j$, $\Lambda_{ii} \neq \Lambda_{jj}$, $\therefore K_{Dij} = 0$. Introducing new coordinate system $P = Q^{-1}X$, the potential energy of the system is given by $\frac{1}{2}X^*KX = \frac{1}{2}P^*Q^*KQP = \frac{1}{2}P^*K_DP$. The system is decoupled into two oscillators in the new coordinates that should have K_D as its stiffness matrix. Let the damping matrix in the new coordinates be C_D. Then,

$$\begin{aligned} C_D\dot{P} + K_DP &= \eta(t) \\ \text{or } \dot{P} &= -C_D^{-1}K_DP + C_D^{-1}\eta(t) \end{aligned} \tag{3.5}$$

As the state matrix is decoupled in the new coordinates,

$$-C_D^{-1}K_D = \Lambda \Rightarrow C_D = -K_D\Lambda^{-1} \tag{3.6}$$

and from fluctuation dissipation theorem for single oscillator model,

$$\langle \eta\eta^* \rangle = 4K_BTC_D \tag{3.7}$$

Now applying the transformation to the original coordinates,

$$\dot{P} = -Q^{-1}C^{-1}KQP + Q^{-1}C^{-1}F(t) \tag{3.8}$$

Comparing the last terms of 3.5and 3.8,

$$Q^{-1}C^{-1}F(t) = C_D^{-1}\eta(t) \tag{3.9}$$

Therefore

$$\begin{aligned}
& & F(\omega) &= CQC_D^{-1}\eta(\omega) \\
\Rightarrow & & \langle F(\omega)F^*(\omega)\rangle &= 4K_BTCQC_D^{-1}C_D{C_D^*}^{-1}Q^*C^* \\
& & &= 4K_BTCQC_D^{-1}Q^*C \\
\text{Now} & & QC_D^{-1}Q^* &= -Q\Lambda K_D^{-1}Q^* \\
& & &= -Q\Lambda Q^{-1}K^{-1}{Q^*}^{-1}Q* \\
& & &= -Q\Lambda Q^{-1}K^{-1} \\
& & &= C^{-1}KK^{-1} \\
\therefore & & \langle F(\omega)F^*(\omega)\rangle &= 4K_BTCC^{-1}C \\
& & &= 4K_BTC
\end{aligned} \tag{3.10}$$

3.4.2 Sensing using coupled oscillators

This section analysis the possible advantages of measuring a signal through a known networked system whose individual nodes can be measured against a single measurement node. This analysis focuses on an optical tweezers setup that can trap and detect position of multiple beads tethered through a biomolecule with setup shown in Figure 3.5. Often the signals measured are corrupted by process noise and measurement noise, thus it is imperative that one minimizes the effect of noise in the final estimates. The intuition behind this lies in signal processing techniques that can fuse common information from multiple signal sources (bead position in our case) to get less noisy estimates of the information. The analysis described here attempts at finding conditions under which a chosen detection approach will show improvement in signal to noise ratio (SNR).

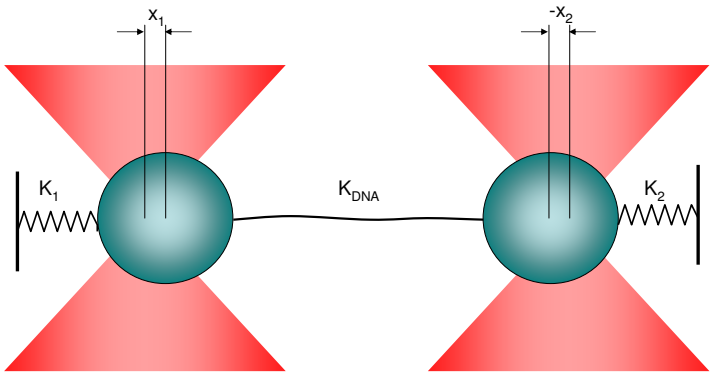


Figure 3.5: Double trap setup

The state space description of the tethered double trap system is given below.

$$\begin{pmatrix} \dot{x}_1 \\ \\ \dot{x}_2 \end{pmatrix} = \underbrace{\begin{bmatrix} \frac{-k_1-k_d}{\gamma_1} & \frac{k_d}{\gamma_1} \\ \\ \frac{k_d}{\gamma_2} & \frac{-k_2-k_d}{\gamma_2} \end{bmatrix}}_{A} \begin{pmatrix} x_1 \\ \\ x_2 \end{pmatrix} + \underbrace{\begin{bmatrix} \frac{1}{\gamma_1} & 0 \\ \\ 0 & \frac{1}{\gamma_2} \end{bmatrix}}_{B} \underbrace{\begin{pmatrix} u_1+\eta_1 \\ \\ u_2+\eta_2 \end{pmatrix}}_{u+\eta} \tag{3.11}$$

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \underbrace{\begin{pmatrix} n_1 \\ n_2 \end{pmatrix}}_{n}$$

where u_1 and u_2 are the force inputs to beads 1 and 2 respectively and are also the signals that are to be estimated but corrupted by the process noise η_1, η_2 and measurement noise n_1, n_2. Solution to 3.11 in Laplace variable s is given by the following equation.

$$\begin{aligned} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} &= (sI - A)^{-1} B(u+\eta) \\ &= \frac{1}{\Delta} \begin{bmatrix} \gamma_2 s + k_2 + k_d & k_d \\ k_d & \gamma_1 s + k_1 + k_d \end{bmatrix} (u+\eta) \end{aligned} \tag{3.12}$$

$$where, \quad \Delta = \gamma_1\gamma_2 s^2 + \{\gamma_1(k_2+k_d) + \gamma_2(k_1+k_d)\}\, s + k_d(k_1+k_2) + k_1k_2$$

Hence the objective is to best estimate u_1 and u_2 in steady state i.e., find $\hat{u}_i$ such that $\langle u_i \rangle = \langle \hat{u}_i \rangle$ such that $\left\langle (\Delta\hat{u}_i)^2 \right\rangle$ is minimized for $i = 1, 2$ and $\Delta u_i = \hat{u}_i - \langle \hat{u}_i \rangle$, where $\langle . \rangle$ denotes time average over a finite bandwidth. The noise signals are assumed to be white and uncorrelated so that $\langle nn' \rangle = R$ is a diagonal matrix. Also it can be shown that for

a coupled oscillator in a thermal bath with temperature T, the thermal noise variance $\langle \eta\eta' \rangle = 4K_BTC\omega_B$, where C is the symmetric damping matrix (proof in Appendix) and ω_B is the bandwidth over which noise is integrated.

Method I (Inversion)

$$\begin{aligned} \underbrace{\begin{pmatrix} \hat{u}_1 \\ \hat{u}_2 \end{pmatrix}}_{\hat{u}} &= \left(\frac{1}{\Delta} \begin{bmatrix} \gamma_2 s + k_2 + k_d & k_d \\ k_d & \gamma_1 s + k_1 + k_d \end{bmatrix} \right)^{-1} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \\ &= \begin{bmatrix} \gamma_1 s + k_1 + k_d & -k_d \\ -k_d & \gamma_2 s + k_2 + k_d \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \end{aligned} \tag{3.13}$$

Filter, $Q = \dfrac{c}{s+c}$ is introduced to make inversion realizable with unity gain up to a bandwidth of c rad/s to obtain the following relation.

$$\begin{pmatrix} \hat{u}_1 \\ \hat{u}_2 \end{pmatrix} = \begin{bmatrix} (\gamma_1 s + k_1 + k_d)Q & -k_d \\ -k_d & (\gamma_2 s + k_2 + k_d)Q \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \tag{3.14}$$

Substituting expressions for y_1 and y_2 from 3.11 into 3.14

$$\begin{pmatrix} \hat{u}_1 \\ \hat{u}_2 \end{pmatrix} = \begin{pmatrix} u_1 + \eta_1 \\ u_2 + \eta_2 \end{pmatrix} + \begin{bmatrix} (\gamma_1 s + k_1 + k_d)Q & -k_d \\ -k_d & (\gamma_2 s + k_2 + k_d)Q \end{bmatrix} \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} \tag{3.15}$$

Clearly, the mean value of the estimates match the true mean by construction. Next, find noise variance over a bandwidth of ω_B small enough that the noise shaping transfer functions remain flat in the band to simplify calculations.

$$\begin{aligned} \left\langle (\hat{u} - \langle \hat{u} \rangle)(\hat{u} - \langle \hat{u} \rangle)' \right\rangle &= \langle \hat{u}\hat{u}' \rangle - \langle \hat{u} \rangle \langle \hat{u}' \rangle \\ \hat{u}\hat{u}' &= uu' + \eta\eta' + Knn'K' \\ \langle \hat{u}\hat{u}' \rangle &= \langle uu' \rangle + \langle \eta\eta' \rangle + K \langle nn' \rangle K' \\ \langle \hat{u} \rangle \langle \hat{u}' \rangle &= \langle uu' \rangle \\ \therefore \quad \left\langle (\hat{u} - \langle \hat{u} \rangle)(\hat{u} - \langle \hat{u} \rangle)' \right\rangle &= \langle \eta\eta' \rangle + K \langle nn' \rangle K' \\ &= 4K_BTB^{-1}\omega_B + \underbrace{KRK}_{>0} \end{aligned} \tag{3.16}$$

Therefore, the noise in the estimates due to Langevin forcing is same as that for a single bead detection however the measurement noise is scaled by the matrix K. Higher the stiffness of tether, k_d, higher will be be noise variance.

Method II (Partial Inversion, assume $u_2 = 0$)

From 3.12, estimates for u_1 from y_1 and y_2 are obtained as follows,

$$\begin{aligned}\begin{pmatrix} \hat{u}_{1a} \\ \hat{u}_{1b} \end{pmatrix} &= \left(\frac{1}{\Delta}\begin{bmatrix} \gamma_2 s + k_2 + k_d & 0 \\ 0 & k_d \end{bmatrix}\right)^{-1} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \\ &= \begin{pmatrix} \dfrac{\Delta}{\gamma_2 s + k_2 + k_d} Q & 0 \\ 0 & \dfrac{\Delta}{k_d} Q^2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}\end{aligned} \tag{3.17}$$

By construction, the mean value of estimates match the true inputs. Next, find the noise in the individual estimates.

$$\begin{aligned}\left\langle (\hat{u}_1 - \langle \hat{u} \rangle)^2 \right\rangle &= \langle \hat{u}_1^2 \rangle - \langle \hat{u}_1^2 \rangle \\ \langle \hat{u}_{1a}^2 \rangle &= \langle u_1^2 \rangle + \langle \eta_1^2 \rangle + \left| \frac{k_d}{\gamma_2 s + k_2 + k_d} \right|_{s=0}^2 \langle \eta_2^2 \rangle \\ &\quad + \left| \frac{\Delta}{\gamma_2 s + k_2 + k_d} \right|_{s=0}^2 \langle n_1^2 \rangle \\ \therefore \left\langle (\hat{u}_{1a} - \langle \hat{u}_{1a} \rangle)^2 \right\rangle &= 4K_B T \gamma_1 \omega_B + \left(\frac{k_d}{k_2 + k_d} \right)^2 4K_B T \gamma_2 \omega_B \\ &\quad + \left(\frac{k_d(k_1 + k_2) + k_1 k_2}{k_2 + k_d} \right)^2 R_1\end{aligned} \tag{3.18}$$

Likewise,

$$\begin{aligned}\left\langle (\hat{u}_{1b} - \langle \hat{u}_{1b} \rangle^2 \right\rangle &= 4K_B T \gamma_1 \omega_B + \left(\frac{k_1 + k_d}{k_d} \right)^2 4K_B T \gamma_2 \omega_B \\ &\quad + \left(\frac{k_d(k_1 + k_2) + k_1 k_2}{k_d} \right)^2 R_2\end{aligned} \tag{3.19}$$

Method III (Convex fusion of two estimates)

Improvement over previous method can be achieved by fusing the two estimates. An intuitive approach would be to take a linear combination of the two estimates such that the

noise variance is minimized. Let $0 \leq \alpha \leq 1$, $\Delta = k_d(k_1 + k_2) + k_1 k_2$ and propose,

$$\begin{aligned} \hat{u}_1 &= \alpha \hat{u}_{1a} + (1-\alpha)\hat{u}_{1b} \\ &= u_1 + \eta_1 + \alpha\left(\frac{k_d}{k_2+k_d}\right)\eta_2 + \alpha\left(\frac{\Delta}{k_2+k_d}\right)n_1 + \\ &\quad +(1-\alpha)\left(\frac{k_1+k_d}{k_d}\right)\eta_2 + (1-\alpha)\left(\frac{\Delta}{k_d}\right)n_2 \end{aligned} \tag{3.20}$$

Then the expression for noise variance is given by,

$$\begin{aligned} \left\langle \Delta\hat{u}_1^2 \right\rangle &= \langle \eta_1^2 \rangle + \left\{ \frac{\alpha^2 k_d^2}{(k_2+k_d)^2} + \frac{(1-\alpha)^2(k_1+k_d)^2}{k_d^2} \right\} \langle \eta_2^2 \rangle + \\ &\quad + \left(\frac{\alpha\Delta}{k_2+k_d}\right)^2 \langle n_1^2 \rangle + \left(\frac{(1-\alpha)\Delta}{k_d}\right)^2 \langle n_2^2 \rangle \end{aligned} \tag{3.21}$$

Note that for all values of α, the estimate has accumulated more thermal noise as against single bead case. Nevertheless one can pick $\alpha = \arg\min_\alpha(\Delta\hat{u}_1^2)$ that can be found by differentiating $\Delta\hat{u}_1^2$ w.r.t. α.

$$\alpha = \frac{\left(\frac{k_1+k_d}{k_d}\right)^2 \langle \eta_2^2 \rangle + \left(\frac{\Delta}{k_d}\right)^2 \langle n_2^2 \rangle}{\left(\frac{k_d}{k_2+k_d}\right)^2 \langle \eta_2^2 \rangle + \left(\frac{k_1+k_d}{k_d}\right)^2 \langle \eta_2^2 \rangle + \left(\frac{\Delta}{k_2+k_d}\right)^2 \langle n_1^2 \rangle + \left(\frac{\Delta}{k_d}\right)^2 \langle n_2^2 \rangle} \tag{3.22}$$

Method IV (Equal and opposite forces, $u_1 = u$, $u_2 = -u$)

This setup involves applying equal and opposite forces on the two beads. This situation arises when changes in DNA tether length pull the beads together. Substituting $u_1 = u$ and $u_2 = -u$ in 3.12,

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \frac{1}{\Delta}\begin{bmatrix} \gamma_2 s + k_2 + k_d & k_d \\ k_d & \gamma_1 s + k_1 + k_d \end{bmatrix}\begin{pmatrix} u + \eta_1 \\ -u + \eta_2 \end{pmatrix}$$

$$= \frac{1}{\Delta}\begin{bmatrix} \gamma_2 s + k_2 \\ -\gamma_1 s - k_1 \end{bmatrix} u + \frac{1}{\Delta}\begin{bmatrix} \gamma_2 s + k_2 + k_d & k_d \\ k_d & \gamma_1 s + k_1 + k_d \end{bmatrix}\begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix}$$

From the above expressions, propose estimating u from measurements as follows:

$$\begin{pmatrix} \hat{u}_a \\ \hat{u}_b \end{pmatrix} = \Delta \begin{bmatrix} \frac{1}{\gamma_2 s + k_2} & 0 \\ 0 & \frac{-1}{\gamma_1 s + k_1} \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \tag{3.23}$$

Substituting expressions for y_1 and y_2 from 3.11 into 3.23,

$$\begin{pmatrix} \hat{u}_a \\ \hat{u}_b \end{pmatrix} = \Delta \begin{bmatrix} \frac{1}{\gamma_2 s + k_2} & 0 \\ 0 & \frac{-1}{\gamma_1 s + k_1} \end{bmatrix} \left\{ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} \right\}$$

$$= \Delta \begin{bmatrix} \frac{1}{\gamma_2 s + k_2} & 0 \\ 0 & \frac{-1}{\gamma_1 s + k_1} \end{bmatrix} \left\{ \frac{1}{\Delta} \begin{bmatrix} \gamma_2 s + k_2 \\ -\gamma_1 s - k_1 \end{bmatrix} u \right.$$

$$\left. + \frac{1}{\Delta} \begin{bmatrix} \gamma_2 s + k_2 + k_d & k_d \\ k_d & \gamma_1 s + k_1 + k_d \end{bmatrix} \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} + \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} \right\}$$

$$= \begin{bmatrix} u \\ u \end{bmatrix} + \begin{bmatrix} 1 + \frac{k_d}{\gamma_2 s + k_2} & \frac{k_d}{\gamma_2 s + k_2} \\ \frac{-k_d}{\gamma_1 s + k_1} & -1 - \frac{k_d}{\gamma_1 s + k_1} \end{bmatrix} \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} + \begin{bmatrix} \frac{n_1}{\Delta(\gamma_2 s + k_2)} \\ \frac{n_2}{\Delta(\gamma_1 s + k_1)} \end{bmatrix}$$

Note that each of the estimates has more thermal noise due to coupling than would have been without coupling. For simplicity, measurement noise will be assumed to be zero for further analysis. The two estimates will now be fused as before using convex combination such that noise is minimized.

$$\begin{aligned} \hat{u} &= \alpha \hat{u}_a + (1 - \alpha) \hat{u}_b \\ &= u + \left[\alpha \left(1 + \frac{k_d}{\gamma_2 s + k_2} \right) + (\alpha - 1) \frac{k_d}{\gamma_1 s + k_1} \right] \eta_1 \\ &\quad + \left[\alpha \frac{k_d}{\gamma_2 s + k_2} + (\alpha - 1) \left(1 + \frac{k_d}{\gamma_1 s + k_1} \right) \right] \eta_2 \\ &= (\alpha p - q)\, \eta_1 + (\alpha p - q - 1)\, \eta_2 \end{aligned}$$

where $p = 1 + \dfrac{k_d}{\gamma_2 s + k_2} + \dfrac{k_d}{\gamma_1 s + k_1}$ and $q = \dfrac{k_d}{\gamma_1 s + k_1}$. Hence,

$$\left\langle \Delta\hat{u}^2 \right\rangle\big|_{\omega_B} = (\alpha p - q)^2 \left\langle \eta_1^2 \right\rangle + (\alpha p - q - 1)^2 \left\langle \eta_1^2 \right\rangle$$

ω_B is chosen such that the transfer functions seen by $\eta_1$and $\eta_2$is flat till this band. This implies that ω_B has to be the smaller of $\frac{k_1}{\gamma_1}$and $\frac{k_2}{\gamma_2}$. For optimal combination, minimize w.r.t. α,

$$\begin{aligned}
& & \frac{d}{d\alpha}\left\langle \Delta\hat{u}^2 \right\rangle\big|_{\omega_B} &= 0 \\
\Rightarrow & & 2(\alpha p - q)\, p \left\langle \eta_1^2 \right\rangle + 2(\alpha p - q - 1)\, p \left\langle \eta_2^2 \right\rangle &= 0 \\
\Rightarrow & & (\alpha p - q)\left(\left\langle \eta_1^2 \right\rangle + \left\langle \eta_2^2 \right\rangle\right) &= \left\langle \eta_2^2 \right\rangle \\
\Rightarrow & & \alpha &= \frac{1}{p}\left(\frac{\left\langle \eta_2^2 \right\rangle}{\left\langle \eta_1^2 \right\rangle + \left\langle \eta_2^2 \right\rangle} + q\right)
\end{aligned}$$

If the two beads are identical then $\alpha = 0.5$ corresponding to error variance of $\frac{1}{2}\left\langle \eta^2 \right\rangle$,i.e., half the error variance obtained with single bead.

Method V (Wiener Filtering)

In this section an optimal estimate will be obtained utilizing following assumptions

1. $\langle u \rangle = 0$, u is a widesense stationary signal, i.e., ϕ_{uu} is time independent.
2. $\phi_{u\eta} = 0$
3. $\phi_{n\eta} = 0$
4. $\phi_{un} = 0$

Then,

$$\begin{aligned}
\phi_y &= \phi_x + \phi_n \\
\phi_x &= (sI - A)^{-1} B(\phi_u + \phi_\eta) B^*(-sI - A)^{-1} \\
\phi_{uy} &= E\left\{u(s)y^*(s)\right\} \\
&= E\left\{u(s)u^*(s)\right\} B^*(sI - A)^{-1*} \\
&= \phi_u B^*(sI - A)^{-1*} \\
&= \phi_u B^*(-sI - A)^{-1}
\end{aligned}$$

Under these assumptions, a multi-input multi-output (MIMO) Wiener filter can be derived to obtained an optimal filter in the least squares sense. Denote the filter by W_{uy}. A noncausal Wiener filter is then given by

$$\begin{aligned}
\therefore\ W_{uy} &= \phi_{uy}\phi_y^{-1} \\
&= \phi_u B^*(sI-A)^{-1}\left[(sI-A)^{-1}B(\phi_u+\phi_\eta)B^*(-sI-A)^{-1}+\phi_n\right]^{-1} \\
&= \phi_u B^*(sI-A)^{-1}\left[(sI-A)^{-1}B(\phi_u+\phi_\eta)B^*(-sI-A)^{-1}+\phi_n\right]^{-1} \\
&= \phi_u B^*\left[(sI-A)^{-1}B(\phi_u+\phi_\eta)B^*(-sI-A)^{-1}(-sI-A)+\phi_n(-sI-A)\right]^{-1} \\
&= \phi_u B^*\left[(sI-A)^{-1}B(\phi_u+\phi_\eta)B^*+\phi_n(-sI-A)\right]^{-1} \qquad (3.24)
\end{aligned}$$

Note that the Wiener filter is by construction such that if r is a root of the filter then so is $-r$. Hence this filter is unstable in forward time direction. However if we assume bilateral Laplace transformation then we can decompose the filter into stable and unstable components. The stable component is the causal part of the wiener filter and the unstable component is the noncausal part of the filter which when implemented in negative time direction is stable. With the above filter, find the error variance of the signal $u-\hat{u}$.

$$\begin{aligned}
u-\hat{u} &= u - W_{uy}y \\
&= \left[u - W_{uy}\left\{G(u+\eta)+n\right\}\right] \\
\therefore\ \langle(u-\hat{u})^2\rangle &= \int\left\{(I-W_{uy}G)\,\phi_u\,(I-W_{uy}G)^* + W_{uy}G\phi_\eta G^*W_{uy}^* + W_{uy}\phi_n W_{uy}^*\right\} \qquad (3.25)
\end{aligned}$$

This was evaluated numerically for various simulation test cases.

3.4.2.1 Simulations

Simulations were performed using the model described by Eq. 3.11. Input force u_2 is set to 0 and u_1 is generated as a sequence of random, zero mean Gaussian noise with variance of 20. Estimation of this force was done by the Wiener filter described by 3.24. The theoretical error variance was computed by numerically integrating the expression in 3.25. This was compared with error variance obtained from simulated results. Figure 3.6 shows the results. In plots of Figure 3.6(a)-(e), only measurement noise was added. In plots of Figure 3.6(f)-(i), thermal noise was also added along with measurement noise. Thermal noise was also simulated as zero mean Gaussian noise with variance of $4K_BT\gamma$ to each bead using γ values for the respective beads.

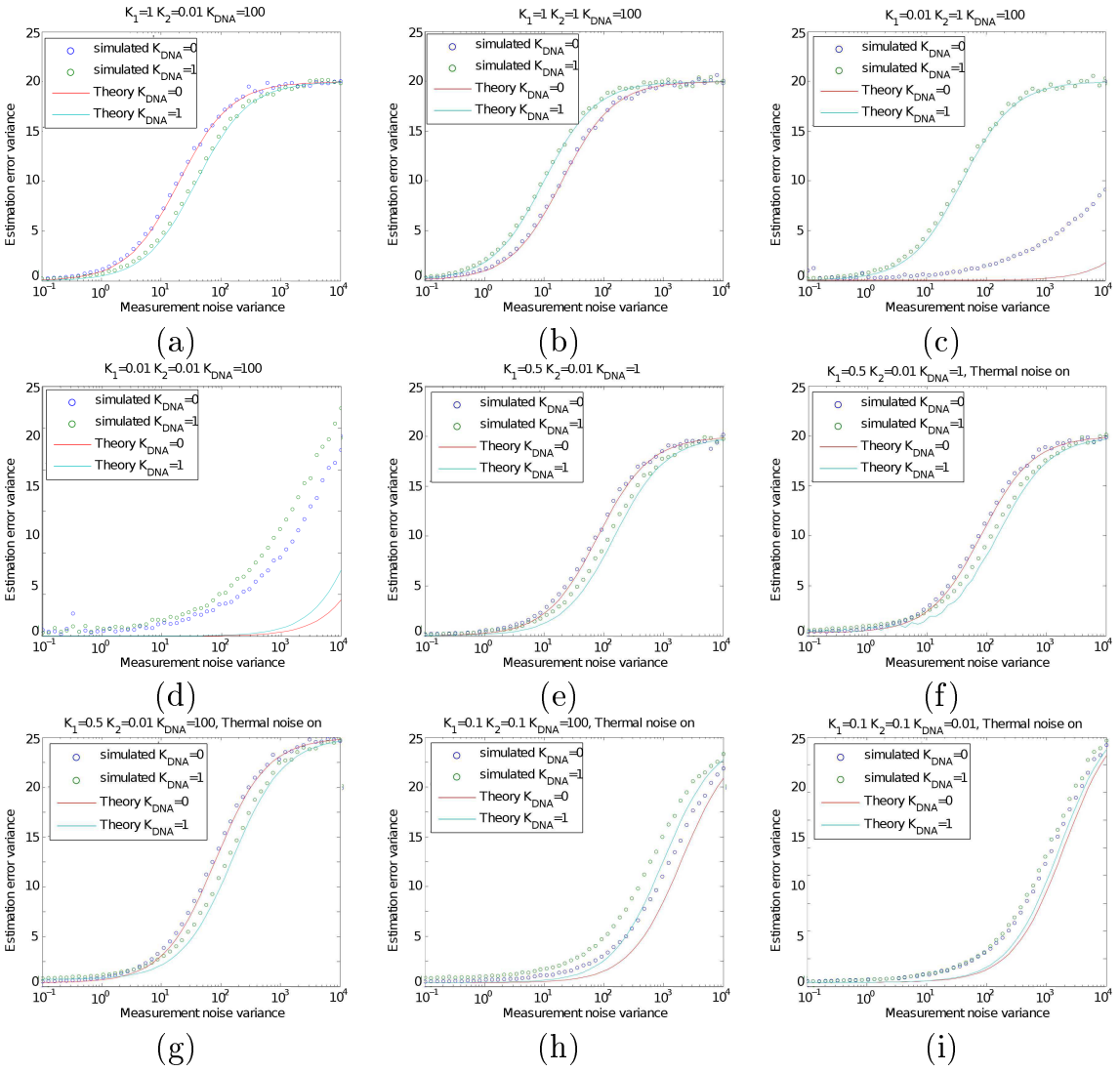


Figure 3.6: Estimation error by Wiener filter for various setups.

3.4.3 Conclusion

From the analysis it is clear that the force estimates using two measurements from a coupled system has more noise compared to single bead measurements in general. There is slight improvement in cases where stiffness of the bead 2 is kept low and DNA stiffness is high. In optical tweezers setups thermal noise dominates over measurement noise so practical advantage is not there in detection of external force through such network of beads. Interestingly, if one focuses on a different signal, e.g. equal and opposite force acting on the two tethered beads, then one can deduce the magnitude of this force with better resolution compared to a single bead.

Chapter 4

Kinesin Techniques

Kinesin is a motor protein that carries large intracellular components like vesicles by moving along cylindrical protein structures called microtubules. Kinesin's monomeric structure has a large 'head' that is connected to a long coiled chain(stalk) through a short flexible chain called 'neck'. The other end of the stalk is connected to the 'tail' portion. Two such monomeric units coil around each other to form kinesin dimer (see Figure 4.1). The two kinesin heads attach to microtubules one by one in a 'walking' motion while the tail is attached to the 'cargo' that needs to be transported.

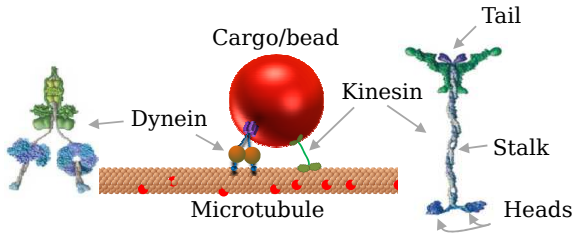


Figure 4.1: Cartoon illustrating cellular cargo transport by kinesin and dynein

One of the earliest single molecule experiments performed using optical tweezers was on kinesin by [24] in 1990. Their method involved coating of $0.2\mu m$-diameter beads with a carrier protein (Bovine Serum Albumin, BSA or mixture of casein and cytochrome) and subsequently incubating it with kinesin with varying concentrations. Silane treated microscope slide was incubated with Taxol-stabilized microtubules which bound to the glass surface. Using optical tweezers the beads were captured and brought close to the microtubules. Upon attachment of myosin to microtubule, the trap was shut off and bead movement was monitored through video camera. It was observed that under high concentration of kinesin corresponding to about 17 or more kinesin molecules per bead, nearly all

beads traveled to the end of the microtubules (about $4\mu m$). Under lower concentrations (less than 1 kinesin molecule per bead), the beads moved on an average of 1.4 μm and then got released. Several possibilities were suggested in the paper describing the walking mechanism where kinesin heads detach from microtubule for a short duration (less than 10ms) before reattaching and completing the stroke. Three years later in 1993, Block's group published a paper on kinesin stepping with relatively high resolution data [25] and showed that kinesin takes steps of about 8nm while moving over microtubules. The experimental setup involved interferometer based position detection. Silica beads were treated with protein(BSA) and incubated with kinesin. Microtubules were introduced into a flow cell that had been incubated with a chemical (4-aminobutyldimethylmethoxysilane) to which the microtubules bind. Beads coated with kinesin were trapped and brought close to the microtubules. The position data was captured for different laser power corresponding to high and low load regime.

4.1 Materials and methods

4.1.1 Kinesin assays

Kinesin is expressed in animal cells by introducing DNA vectors in the host cell. Due to limited resources, these protocols couldn't be followed in our lab. Instead, kinesin was gifted by Thomas Hays Lab.

4.1.2 Microtubule preparation

Microtubules are formed by polymerizing α and β tubulin monomers in presence of Taxol. Following protocol gives high density fluorescent labeled microtubules with average length greater than 50 microns. Prepare labeled tubes as follows

BRB80 This is 80mM PIPES pH 6.9, 1mM EGTA, 1mM $MgCl_2$.

DTT100 Dilute 1M DTT 10-fold in BRB80 to make 100μM DTT.

Tax100 Thaw Taxol (Sigma, T7402-1mg/117μL, 10mM) stock stored at -80 degrees. Make 100μM Taxol solution by diluting 100-fold in BRB80.

MTB Microtubule buffer. This is the buffer in which microtubules remain stable. This buffer is 10μM Taxol, 1mM GTP, 1mM DTT in BRB80.

BRB80+Glycerol Mix BRB80 and glycerol in the ratio of 60:40 v/v. Pipette up and down several times to mix or use vortex shaker. This stock can be stored in 4 degrees

refrigerator for later use. Mix large amounts (>100 μL) because pipetting small amounts of glycerol is not easy.

Cushion This buffer provides viscous environment for free tubulin during centrifugation in order to separate polymerized and unpolymerized tubulin. Cushion buffer is 10μM Taxol, 1mM GTP, 1mM DTT in BRB80+Glycerol mix. Pipette up and down several times.

Assay

Take 1 aliquot of Rh-tubulin (1 μL of 4 μg/mL) and mix with thawed 1 aliquot of tubulin (50 μL of 1 μg/mL). Mix well by pipetting. Centrifuge the mix at 15,000 rpm for 10 minutes at 4°C. In my case, the centrifuge was prechilled using ice packs stored in -80°C freezer. Transfer the solution to a new 0.5mL tube. Add 0.5 μL of 0.1M GTP stock (1mM final conc.), 0.5 μL of DTT100 and 6 μL of Tax100. Pipette up and down several times. Incubate 5' at 37 degrees for 30 mins. Next, add 35 μL of Microtubule Buffer (MTB) with no pipetting up and down (the total volume should now be 100 μL). Now introduce 40 μL of Cushion buffer. This is to be done using long narrow tipped flexible pipettes. The pipette tip should be touching the bottom of the microtube and slowly eject Cushion into microtube solution. At this stage one can see a meniscus between Cushion and previous contents of microtube solution. The idea is to provide a cushion to previous contents. While ejecting CB into microtube, do not press the micro-pipette beyond the stopping point, else it can introduce bubbles and may lose the meniscus. Gently pull out micropipette out of the microtube without disturbing the meniscus.

Then spin at 15K in the centrifuge for 35 mins, keeping a note of which side of microtube is on the outer side of centrifuge, as pellets will be formed there. The pellets may be visible as faint pink blob on the wall of the tube that faced the outer side of the centrifuge. Remove the supernatant by pipetting out 140 μL. Resuspend the pellets in 50 μL of MTB carefully without mixing. This will dilute free tubulin further. It is important to remove out as much free tubulin as possible in order to ensure that microtubules stick properly to the glass slides in the kinesin bead assay. If required the microtubule solution may be centrifuged through the cushion buffer again. Pipette out 50 μL of the solution and then add fresh 50 μL of MTB buffer. This time mix the solution slowly by pipetting up and down several times (50 times) to disperse the solution well. Wrap final tube in aluminum foil to protect from light. The microtubules are stable for months. Dilution of 1:50 gives sufficient density for visualization under fluorescence microscope.

4.1.3 Silanizing glass

Silanization of glass provides amine groups -NH_2 hanging form the glass surface. The carboxyl groups on the proteins bonds with the hanging amine groups to form a stable attachment point. Hence proteins stick to the silanized glass. To silanize glass coverslips (24mmx60mm), they must be acid washed first. To acid wash coverslips, soak glass slides in 1M NaOH in presence of 2% detergent. Rinse with water then immerse in 50% H_2SO_4 for 1 hour. Rinse with absolute ethanol. If absolute ethanol is unavailable rinse in ddH_2O. Dry in desiccator. Perform silanization on the day of the experiment. Add 1 mL of APTES to 20 mL acetone (5% solution). Place acid washed coverslips in the APTES solution for 30-45 mins. Rinse with acetone. Rinse in running water for several seconds. Dry in desiccator.

4.1.4 Polylysine coating

Poly-l-Lysine is a positively charged polymer that gets adsorbed to the negatively charged glass surface. Therefore Polylysine coating provides positively charged surface for attracting negatively charged microtubules or other proteins. For Polylysine coating of glass coverslips, they must be cleaning using KOH solution as follows. Prepare saturated KOH solution in ethanol. Add about 20g of KOH pellets in 200 mL ethanol. Let KOH dissolve completely in ethanol. It may take couple of hours for dissolution. Place coverslips in dispersed fashion to the solution. Let stay for at least 30 min. May be left overnight as well. Rinse the coverslips by dipping individual coverslips in deionized water until the water runs smoothly off the glass surface. Rinse the coverslip in running stream of clean water for 30 seconds for both sides of the coverslip. Dry in a desiccator if possible or using air stream.

4.1.5 DEAE polymer coating

DEAE (Diethylaminoethyl-Dextran hydrochloride, Sigma D9885) is also a positively charged polymer. It is most convenient and robust to use. A 10μg/mL solution of DEAE in water can coat clean cover glass to sufficient density in 5 minutes.

4.1.6 Bead Assay

Prepare the following

Beads Wash 10 μL of beads (carboxylate latex beads, 0.5 μm diameter, 2.5% w/v) in 100 μL BRB80 by mixing the two and vortexing followed by centrifugation at 10,000g for 2 min. Remove the supernatant and resuspend the pellet in 100 μL BRB80. Repeat 3-4 times. The last resuspension should be in 10 μL BRB80 and 10 μL casein (10 mg/mL). Sonicate the beads in cold water for 30 mins.

OS Oxygen Scavenging System. 100X Stock solutions are 20 μL aliquots, -80°C storage, Glucose (450 mg/ml), Glucose Oxidase (20 mg/ml, Sigma G-2133), Catalase (3.5 mg/ml, Sigma C-40), 50% 2-mercaptoethanol (make fresh – on ice). Prepare on ice, 10X stock: 30 μL buffer 5 μL of each reagent (4) above – add glucose last, keep on ice in tightly capped tube (important!). Solutions are good for above 2 hrs, then need to be remade.

Block Blocking solution. This solution will coat the assay chamber with BSA to prevent beads from sticking to the glass surface. Blocking solution consists of 2 mg/mL BSA and 1x OS in in MTB.

BRB80CA 0.5 mg/mL Casein and 10 μM ATP in BRB80. keep on ice

Bead-Motor Add 2 μL of beads to 16 μL of BRB80CA. Add 2 μL of diluted kinesin. Kinesin dilution is adjusted according to experimental needs. For single molecule experiments, kinesin concentration is kept low enough so that about than half of the kinesin coated beads attach to microtubules and show motility.

Motility 10 μM Taxol, 1mg/mL casein, ATP (desired concentration), 1x OS in BRB80.

Flow Chamber Flow chamber is created by taking a 24mm x 60mm silanized or polylysine coated coverslip and placing two strip of double sided tape parallel to each other and the long side of the coverslip such that a channel is formed roughly 3mm wide along the center of the coverslip. The length of the tape should be larger than 22 mm. Take a 22mm x 22mm and place it on top of the channel. Press the coverslip from top over the tape region to seal the chamber formed. Cut out the extra tape outside the top coverslip. The volume of the chamber formed is in the range of 5-10 μL

To perform the assay, first dilute the microtubules 50 folds and flow in 10 μL of this solution into the flow chamber. Allow 5 minutes for the microtubules to stick to the glass surface. Then flow Blocking solution using perfusion technique, i.e., buffer is flowed in from one end of the chamber and is sucked out of the other end using a filter paper or kimwipe. Let stay for 10 minutes. Repeat once more. Finally dilute 1 μL of bead-motor in 100 μL of Motility solution and perfuse about 15 μL of this solution in the flow chamber. Suck out excess fluid outside the chambers using kimwipe. Seal the chamber using Vaseline or grease. Observe under the microscope. Focus the chamber so that the microtubules are visible. A bead is trapped from the surrounding and brought in the vicinity of the microtubules using piezo stage. Calibration of the trap stiffness and photodiode sensitivity is performed. The stage position and the objective height is adjusted to allow kinesin on the bead to attach to the microtubule. If motility is observed then the adjustment is stopped and data recorded. If

required, constant force mode is activated and the data recorded. Characterizing kinesin flexibility

4.2 Characterizing Kinesin flexibility

In-vivo, multiple kinesin are known to transport a single cargo[26] and this requires coordination among different kinesin molecules. In order not to hinder each others activity, they must be flexible[27]. α-helical coiled coils are expected to be very rigid with an effective persistence length of 150 nm.[28] This implies a very rigid structure for a molecule like kinesin which has a length scale of 150nm and approximate rest length of 65nm within the solution[29, 30]. Then, what imparts flexibility to kinesin? Flexibility of full length kinesin-1 is attributed to the presence of two 'kinks' within the kinesin stalk. The kink proximal to heads behaves as a swivel joint whereas the other kink, in the middle of the stalk, behaves as a hinge joint. The kinks allow free rotation of segments of stalk and thus aid in stress-free procesive motion kinesin heads even in presence of partner motors. These flexible joints also play a role in repressing motility of free soluble kinesin not bound to a cargo[31]. Furthermore, the speed of motor transport is suggested to be an increasing function of the elasticity (flexibility) of the kinesin tether under Brownian ratchet model of kinesin motility[32]. These observations make motor tether flexibility an interesting research area.

Estimation of persistence length of kinesin has not received much attention in the literature, probably because of lack of a method to stretch kinesin by a known extension. Many researchers assume a linear spring assumption for kinesin for simplification of analysis[29] or lack of a nonlinear model[27]. A nonlinear force extension profile was reconstructed from optical tweezers data by Atzberger et al[30]. However, the reconstruction was critical based on geometrical consideration of the optical trap and kinesin attachment, integration of noisy velocity data and several assumptions on kinesin rest length, bead radius etc. No connections were made to the popular worm-like chain (WLC) model. These difficulties potentially lead to significant loss in accuracy. In this report a convenient method that circumvents several of these problems is presented. The scheme requiring fewer assumptions and is robust against noise. A derivative of the WLC model is utilized that relates force to stiffness. Force and stiffness are experimentally determined by analyzing the response of a trapped bead tethered to microtubule-bound kinesin molecule when the trap position is modulated in a sinusoidal fashion. The bead response depends on the stiffness of the kinesin. Therefore, the stiffness is determined from the relation that relates input modulation of trap, response of the bead, trap stiffness and WLC stiffness.

4.2.1 Experimental relation between force and stiffness

First, experimentally determine a relation between tension within and stiffness of kinesin. A kinesin tethered bead is optically trapped and brought near a microtubule attached to glass surface. The trap is dragged along the microtubule until the kinesin heads attach to the microtubules in presence of AMP-PNP. Due to the non-hydrolyzability of AMP-PNP, kinesin heads remain stationary on the microtubule and other end of kinesin is controlled by the bead. Therefore kinesin stretches but the amount of stretch is not measurable because the point of attachment of kinesin heads to the microtubule and that of the tail to the bead is not known. However, if there is a tension to the amount of F_m in the motor then the equilibrium bead position (x_b) and the trap position (x_T) are related as

$$F_m = k_T(x_b - x_T). \tag{4.1}$$

In general force in the kinesin molecule increases with stretching. The relation between force and extension is a nonlinear one and in this section we intend to find that relation. The extension of the molecule cannot be experimentally measured because the position of the motor heads (x_m) is not observable. To overcome this limitation, an experimental setup is realized as shown in Fig. 4.2.

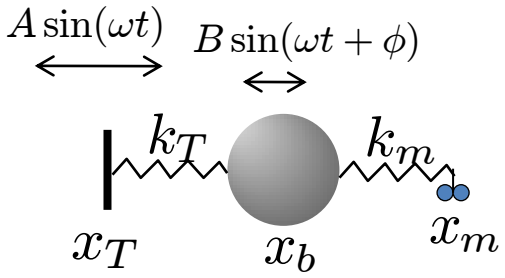


Figure 4.2: Setup for kinesin stretching experiment. x_m is unknown and varied by moving the piezo stage. For small oscialltions in x_T the system is expected to behave like a linear system with linear spring stiffnesses for the trap and the kinesin molecule.

Now consider a linearized version of Eq. 4.1:

$$\frac{\mathrm{d}F_m}{\mathrm{d}x_b} = k_T\left(1 - \frac{\mathrm{d}x_T}{\mathrm{d}x_b}\right).$$

Then define

$$\begin{aligned} z :=& x_m - x_b \\ k_m :=& \frac{\mathrm{d}F_m}{\mathrm{d}z} \end{aligned} \tag{4.2}$$

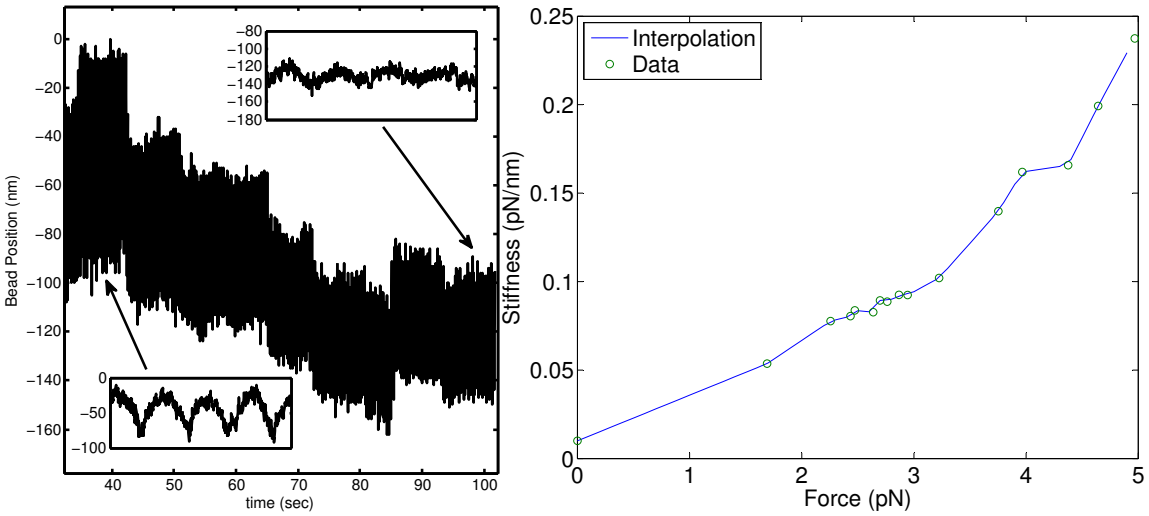
k_m denotes the instantaneous stiffness of the kinesin molecule which depends on the kinesin extension z. x_m is assumed to be constant for small time intervals. Therefore, $k_m = -\frac{\mathrm{d}F_m}{\mathrm{d}x_b}$. The term $\frac{\mathrm{d}x_T}{\mathrm{d}x_b}$ is not known but can be found experimentally by giving sinusoidal oscillations to x_T and measuring the corresponding sinusoidal bead response x_b for a given stretched condition. The response will be sinusoidal because for small enough oscillations, the system behaves like a linear system. This is shown in Fig. 4.3. Let $x_T = A\sin(\omega t)$ and $x_b = B\sin(\omega t)$, then$\frac{\mathrm{d}x_T}{\mathrm{d}x_b} = \frac{A}{B}$. Therefore, stiffness of a kinesin molecule can be measured experimentally for a given equilibrium situation as

$$k_m = k_T \left(\frac{A}{B} - 1 \right). \tag{4.3}$$

FigAfter obtaining k_m, Eq. 4.2 is used to obtain the extension of the molecule, i.e.,

$$z = \int_0^z \mathrm{dz} = \int_0^{F_m} \frac{\mathrm{d}F_m}{k_m}. \tag{4.4}$$

Note that, k_m depends on the extension, z of the molecule, which further depends on the force, F_m on it. Therefore, the integration in Eq. 4.4 needs to be performed numerically.



(a) (b)

Figure 4.3: (a)Bead data showing change in amplitude of oscillations as kinesin is stretched by pulling the bead using optical tweezers. The amplitude is high (top inset) when bead is near the center of the trap indicating low force regime and therefore tension on kinesin is low. On the other hand amplitude is reduced when bead is farther away from trap center (bottom inset) indicating high force regime and therefore tension on kinesin is high which makes it stiffer. (b) Experimentally obtained relation between force and stiffness of kinesin as it is stretched.

4.2.2 Theoretical relation and model fitting

In this section, we explore the goodness of a wormlike chain(WLC) model in explaining the force vs. stretching behavior. The worm-like chain model is given by[33]

$$\hat{F} = \underbrace{\frac{K_B T}{L_p}\left[\frac{1}{4}\left(1-\frac{z}{L_c}\right)^{-2}+\frac{z}{L_c}-\frac{1}{4}\right]}_{WLC(L_c,L_p,z)}. \tag{4.5}$$

This model consists of two unknown parameters: L_p, which is the persistence length and L_c, which is the contour length. The parameters that best fit the experimentally obtained force vs. extension graphs are obtained. The results are shown in Fig. 4.4. The results show that kinesin stiffness can be well desribed by wormlike chain model and the fitted parameters hint toward structural properties of kinesin. A small persistence length of about 2nm and a contour length of about 180nm indicate that indeed kinesin is very flexible. The advantage of this entire scheme is that minimum assumptions need to be made and higher fidelity data can be obtained by observing longer data records for a given stretched condition.

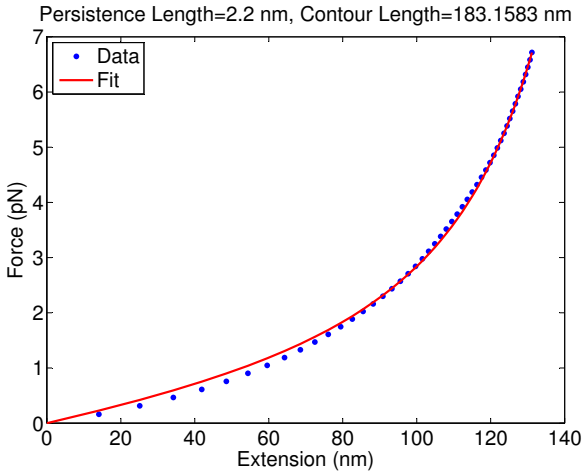


Figure 4.4: Force vs. extension curve for kinesin stretching. A good fit is obtained with the WLC model for the experimental data.

4.3 3D Monte-Carlo simulations of experiments

A popular research methodology is to hypothesize mechanisms through which a system functions and simulate this behavior on computer. By comparing outcomes of simulated system versus measurements of actual systems, the hypothesis can be validated. Another advantage of simulations is that they can be made arbitrarily complex at the cost of computation time. In the context of single kinesin molecule based transport, the problem is conceptually tractable and well understood. However, in real systems, a cargo is transported by multiple motors of possibly different kinds, e.g., kinesin and dyneins are both known to simultaneously transport a cargo where the two kinds of motors pull the cargo in opposite direction, so there is a complex interaction taking place. Unfortunately, the measurement system provides data corresponding to only a few variables in this vastly multivariate scenario. Much of the insight is to be inferred indirectly. There are various geometrical effects and nonlinearities which may affect transport dynamics but not directly observable. These could be,

1. The shape of the cargo
2. Effect of number of motors that are involved in transport.
3. Effect of location of motor attachment on the cargo.
4. Effect of elasticity of the motor
5. Effect of cargo rotation and rotational diffusion

These are some of the questions that can be explored through a three-dimensional simulation and by observing changes in the measurable quantity like cargo position.

4.3.1 Algorithm and Flowchart

The geometry of simulation consists of a microtubule aligned along the x-axis of the Cartesian coordinate system. A spherical bead is assumed to be nominally located at the location of the trap. A variable number of motors are assumed to be attached to the bead. The motors tails are constrained to be on the surface of the bead whereas motor heads point are directed in the direction joining bead center to the motor tail position, when the motor is not attached to the microtubules. If the motor is attached to the microtubule then the motor head position is constrained to walk along the x-axis. Thus translation of bead corresponds to updating the position of the bead center and the same offset is applied to the motor tail position. Rotation of bead corresponds to updating the motor tail positions. Same rotation is applied to all the motor tails in order to maintain relative position of the motors on the bead. The following set of equations described the dynamics of forces coming into play and the simulation framework for a robust simulation.

Symbol	Description	Symbol	Description
$\mathbf{x_b}$	Bead position vector	γ	Translational drag coefficient
$\mathbf{x_t}$	Trap position vector	γ_R	Rotational drag coefficient
$\mathbf{x_{mh}}$	motor head position	$\mathbf{\Gamma_m}$	Net torque due to motors
$\mathbf{x_{mt}}$	Motor tail position	$\mathbf{\Gamma}_\eta$	Torque due to thermal process
$\mathbf{F_t}$	Force by the trap	ROT(p,u,θ)	Rotation of vector p about
$\mathbf{F_m}$	Force by the motors		u by an angle θ
$\mathbf{F}_\eta$	Translational thermal force	$\mathcal{N}_{1\times3}(0,1)$	$\mathbb{R}^3$ random normal vector
$k_m()$	Motor spring force	r	Radius of bead
t_k	Time vector of simulation	η	viscosity of medium
T_s	Nominal sampling time	x_0	Motor rest length
dt	Adaptive sampling time		

Table 4.1

Translational dynamics

$$\begin{aligned}
\mathbf{F_t} &= k_t.(\mathbf{x_b} - \mathbf{x_t}) \\
\mathbf{F_m} &= k_m(\mathbf{x_{mh}} - \mathbf{x_{mt}}) \\
\gamma &= 6\pi\eta r \\
\langle \mathbf{F}_\eta \times \mathbf{F}_\eta \rangle &= 4k_b T\gamma \mathbf{I_{3\times 3}} \\
\mathbf{F} &= \mathbf{F_t} + \mathbf{F_m} + \mathbf{F}_\eta \\
\dot{\mathbf{x}}_\mathbf{b} &= \frac{\mathbf{F}}{\gamma}
\end{aligned}$$

Rotational dynamics

$$\begin{aligned}
\mathbf{\Gamma_m} &= (\mathbf{x_{mt}} - \mathbf{x_b}) \times \mathbf{F_m} \\
\gamma_R &= 8\pi\eta r^3 \\
\dot{\theta} &= \frac{\|\mathbf{\Gamma}\|}{\gamma_R}
\end{aligned}$$

Simulation of translational dynamics with adaptive sampling time dt

$$\begin{aligned}
\tilde{\mathbf{F}} &= \mathbf{F_t} + \mathbf{F_m} \\
d\mathbf{x} &= T_s \frac{\tilde{\mathbf{F}}}{\gamma} \\
dt &= \begin{cases} \frac{T_s}{\|d\mathbf{x}\|} 5, & \|d\mathbf{x}\| > 5 \\ T_s & \text{otherwise} \end{cases} \\
\mathbf{F}_\eta &= \sqrt{2k_b T\gamma \frac{1}{2dt}} \mathcal{N}_{1\times 3}(0,1) \\
\mathbf{F} &= \tilde{\mathbf{F}} + \mathbf{F}_\eta \\
d\mathbf{x} &= dt \frac{\mathbf{F}}{\gamma} \\
\mathbf{x_b} &= \mathbf{x_b} + d\mathbf{x} \\
\mathbf{x_b}(2) &= \max(\mathbf{x_b}(2), 0)
\end{aligned}$$

Simulation of rotational dynamics with previously computed value of dt. Bead is rotationally symmetric so the only thing that needs to be updated is the location of motor tails on

the bead.

$$\begin{aligned}
\mathbf{\Gamma}_\eta &= \sqrt{2k_bT\gamma_R\frac{1}{2dt}}\mathcal{N}_{1\times 3}(0,1) \\
\mathbf{\Gamma} &= \mathbf{\Gamma_m} + \mathbf{\Gamma}_\eta \\
d\theta &= \frac{\|\mathbf{\Gamma}\|}{\gamma_R}dt \\
\mathbf{x_{mt}} &= \mathbf{x_b} + \mathrm{ROT}(\mathbf{x_{mt}} - \mathbf{x_b}, \mathbf{\Gamma}, d\theta)
\end{aligned}$$

Simulating motor head attachment and movement

$$\begin{aligned}
\lambda &= 800\,|F_{stall} - \|\mathbf{F_m}\|| \\
p_{step} &= \lambda dt e^{-\lambda dt} \\
\text{attached?} &= \begin{cases} \text{Yes} & \text{if } \|\mathbf{x_{mt}} \times \mathbf{e_1}\| < 30 \\ \text{No} & \text{if } \|\mathbf{F_m}\| > F_{stall} \\ \text{No change} & \text{otherwise} \end{cases} \\
\mathbf{x_{mh}} &= \begin{cases} \mathbf{x_{mh}} + 8\mathbf{u}(p_{step})\mathbf{e_1} & \text{if attached} \\ \frac{\mathbf{x_{mt}} - \mathbf{x_b}}{r}(r + x_0) & \text{otherwise} \end{cases}
\end{aligned}$$

4.3.2 Validation

The simulation program is validated by running test cases where the results are known or can be matched with experiments.

Test 1: Power spectrum of trapped bead

As shown in Fig. 4.5, theoretical power spectrum of x-coordinate position of a trapped bead matches well with that obtained from 3D simulations.

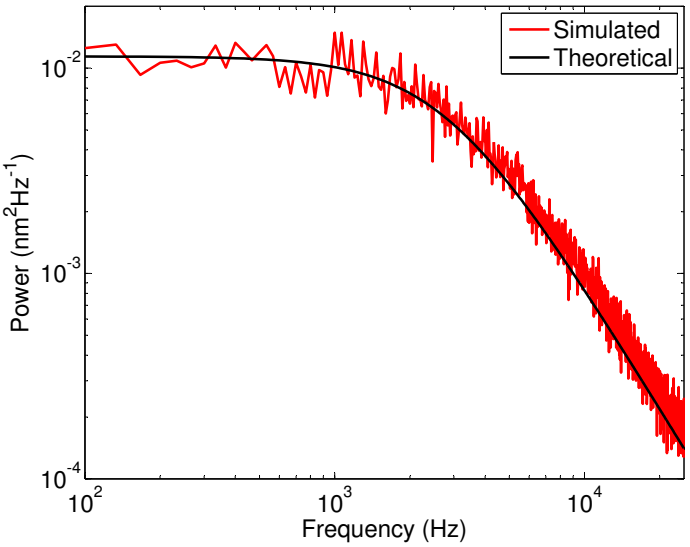


Figure 4.5: Trapped bead spectrum

Test 2: Bead variance change with motor stretch

In this test, the idea is to reproduce the experimental observation of reduced variance when the bead is being pulled by a kinesin molecule. The variance reduces with the force registered by the optical tweezers. Fig. 4.6 shows that the effect is reproduced along with other artifiacts like reattachment of kinesin after detachment and apparent bead velocities.

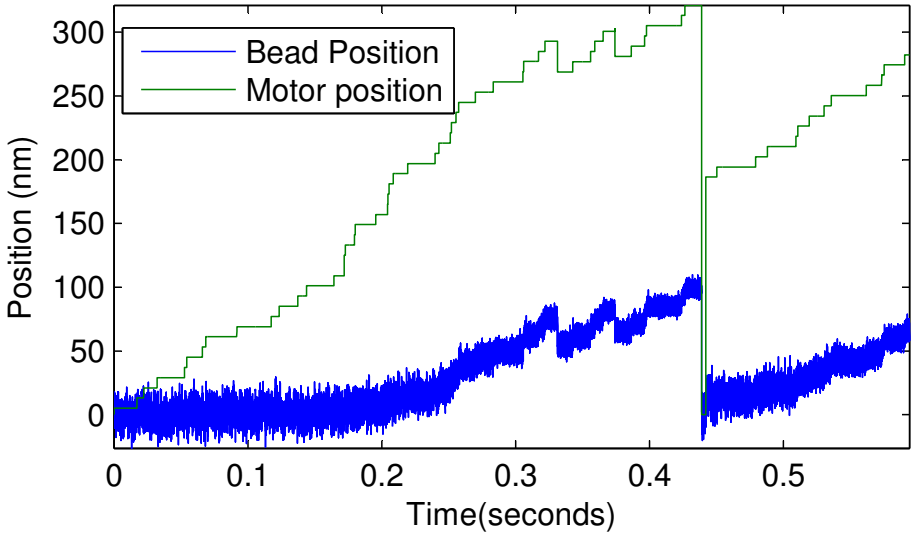
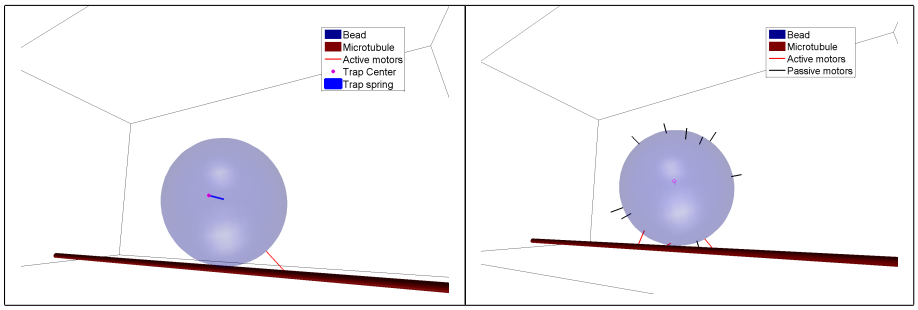


Figure 4.6: Single kinein motor pulling the bead in 3D simulation. The variance in the bead position reduces as the bead moves away from the trap center. This is because the motor stiffness increases with stretching and constrains the bead movement. Also note that the bead velocity is smaller when near the trap center. This is because initially the motor movement is primarily responsible for rotating the bead and only after further rotation is restricted, does the bead get pulled by the motor in a significant manner. Steps in the bead position are apparent only in high stiffness region due to two reasons.

Test 3: Visual confirmation of geometric validity

Visual confirmation of the simulation process is essential when attempting to replicate three dimensional processes. This is because humans have natural 3D processing capability and can detect abnormalities with ease when compared to data dependent tests. In a test case shown in Fig. 4.7, it is verfied that when a motor is stretched it is pointed toward the center of the bead due to geometrical constraints like the bead cannot go below the surface and that the net torque should be zero in equilibrium. In case of multiple motors as well (Fig. , the net effect is sensible at visual level.



(a) (b)

Figure 4.7: Screenshot of 3D geometry

4.3.3 Observations

Some of the interesting observations made through the simulations are as follows:

1. Steps taken by cargo is not uniform if the load on kinesin is varying or if multiple motors are involved in transport. Only, under constant force experiments on single kinesin molecules, can 8 nm steps be expected to be seen in the cargo position.

2. Simulations suggest that kinesin may be experiencing much larger forces than that measured by optical tweezers assuming linear 1-D geometry. Therefore, actual stall force of 6 pN will correspond to a stall force of about 3.5 pN as measured by optical tweezers (this depends on the geometry and thereby on the kinesin flexibility model). Knowledge of actual stall force and that measured by optical tweezers can help compute the extension of the kinesin molecule.

3. The power spectrum of an optically trapped bead position tethered to surface through kinesin molecules exhibits distinct deviation from the power spectrum of an untethered bead. Such a deviation is observed experimentally as well as shown in Fig. 4.8. A good match between experimental and theoretical power spectrums indicated that kinesin indeed behaves like a nonlinear worm-like chain with rotational diffusion of the bead playing a significant role.

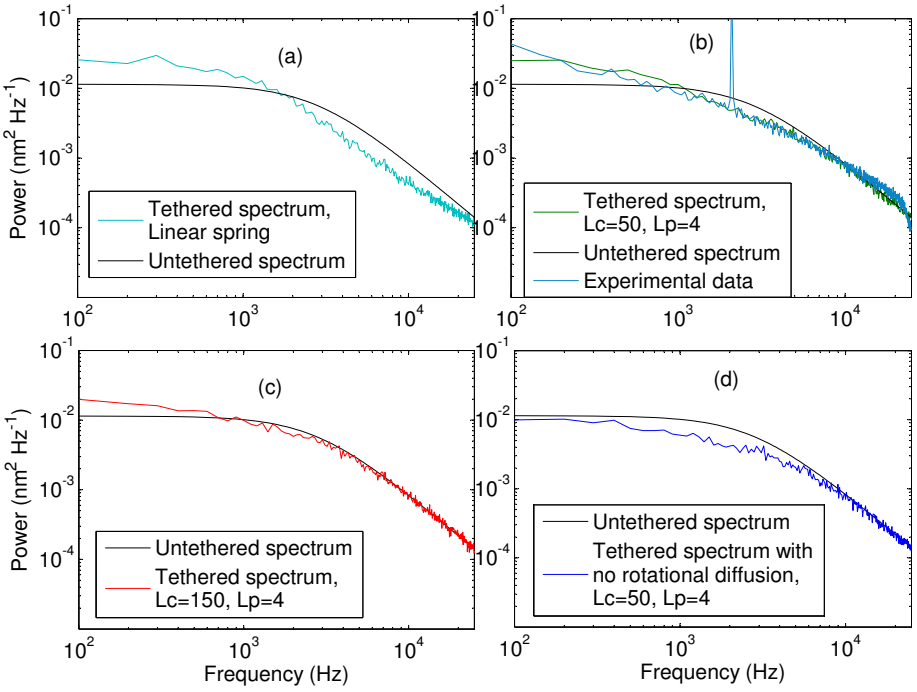


Figure 4.8: Power spectrum of bead position obtained for different simulation settings. In (b) an experimentally obtained power spectrum of tethered bead shows good match with tethered spectrum with worm-like chain model for kinesin. A peak in the experimental plot is due to the sinusoidal actuation of the trap.

Chapter 5

Step Detection Algorithm

Many biological machines move in a stepwise fashion. Such machines include the microtubule motors, kinesin and dynein, which take 8 nm steps along the microtubules lattice[25], as well as the actin based myosin motors, for example myosin V, which takes 32 nm steps[34]. These motor proteins are responsible for the transport of numerous cellular cargoes and malfunctions in their behavior are linked to multiple human diseases[35]. Similarly, many biological processes depend on the ability of proteins to fold and unfold; events that can occur quite rapidly [36]. Here too malfunctions in the folding/unfolding mechanisms can lead to severe consequences. Other examples of important discrete events in biological processes can be found in the conformation changes that underlie ion channel activity [37, 38] or translation of mRNA by RNA polymerase[39].

Given that many biological processes depend on the ability of molecules to change their state in discrete steps, statistics on the sizes and dwell times of these steps can provide useful insights into the dynamics of the processes [40]. For instance, intracellular cargoes are often transported by multiple motors[41] and information on the cargo step size can provide important information on the behavior of these motors. Indeed, in the case of multiple kinesin motors acting to transport a cargo, an 8 nm step by the cargo indicates synchronized action of the motors. In contrast, cargo displacement that is a fraction of the standard 8 nm kinesin step suggests that multiple motors step along the substrate in an asynchronous manner [41]. Thus, by monitoring the step size of the cargo, important information on how the motors coordinate their motion can be obtained.

Recent advances in instrument technology have renewed interest in the methods used to analyze the statistics of stepping motion in single molecule data. Instruments like atomic force microscopes (AFM) [42] and optical tweezers [43] can be used to study the behavior of single molecules under controlled forces. Optical-tweezers have enabled observation of the stepping behavior of motor proteins like kinesin, dynein, myosin and RNA-polymerase [44],

while, micro-cantilever probes have facilitated the study of how macro protein molecules fold and unfold [45]. Even though these instruments provide insight into biological processes at the single molecule scale, additional analytical methods are important for revealing mechanisms. The need for higher temporal resolution is evident, for example, in kinesin based transport studies. Kinesin molecules can carry a cargo along a microtubule at speeds of over 800 nm/sec. However, with existing capabilities, it is hard to discern steps at these speeds. As a consequence, speeds are artificially lowered to less than 50 nm/sec by limiting ATP concentration or by exerting large opposing forces [9]. Similarly, as described above, higher resolution is needed to analyze the action of multiple motors on a single cargo and to resolve variation in the step size in the cargo displacement data[41].

Given the need for methods that are faster and have higher resolution what are the associated challenges? A significant challenge in step detection is to distinguish signal from the noise. This challenge is difficult, since in many studies the step-size to be discerned is often smaller than the standard deviation of the noise. Noise can be categorized into two classes based on its origins. One fundamental source of noise has its origins in the thermal fluctuations of the biological system or the mechanical probe. Noise also arises in the electronics of the measurement devices. Apart from the noise, another key challenge to overcome is the dynamics of the probe being used to investigate the state of the molecule. The probe-dynamics can directly affect the shape of the measured output. For instance, the force caused by a step taken by a motor attached to a bead leads to the bead assuming a new position. However, the bead reaches its new equilibrium position asymptotically, with a lag determined by the time-constant of the dynamics of the bead which is governed by the viscosity of the surrounding medium, the stiffness of the trap and the stiffness of the motor. If the motor stepping rate is faster than the time-constant of the bead dynamics, then the sequential steps interfere with each other. As the bead reaches a new equilibrium position resulting from a first step, it is subjected to new forces due to the subsequent step. Thus, if the response of the probe is slow, the discreet steps will be missed. Apart from distorting step signals, probe dynamics also affects the noise spectrum. For instance, thermal forces that are constantly perturbing a bead, is a white process (has a flat spectrum). However, the effect of thermal noise on the bead position is filtered (appears as colored noise). This is caused by the bead dynamics. If not addressed, such an affect will lead to incorrect conclusions on the stepping behavior of the molecule. In this regard, the step detection methodology reported here represents a significant advance over existing methods.

A step-detection method has to strike a compromise in detection sensitivity - detecting true steps but not counting false steps. If the goal is to capture a higher percentage of true-steps then, for a given step-detection method, the number of false steps detected will

also increase. Apart from the relative size of the steps in comparison to the noise standard deviation, the rate at which steps in molecular state occur also determines the percentage of true positives and percentage of false positives. Indeed, effective step-detection methods implicitly utilize the randomness of the noise process to reduce the effects of noise. The data is averaged over a time window, and the longer the time-window the lower the average level of noise and the easier it is to detect smaller step sizes. However, the size of the time-window is dictated by how often steps occur. The size of the time-window is reduced when the stepping rate is higher; otherwise even the steps in true signal will be averaged leading to incorrect conclusions on the step-size, the number of true steps detected and the timing of the steps. In summary, relative performance of step detection methods is determined by the percentage of true positives versus percentage of false positives scored which depends on the standard deviation of the noise levels and the dwell times between true steps. Unfortunately, there are few performance measures by which to compare step-detection methods.

In the present report, we describe a new step-detection method that addresses the affects of probe dynamics and readily detects small step sizes with short dwell times. We develop a comprehensive set of performance criteria and show that our method outperforms other step detection methods. To validate the method, we characterize the statistics of step changes for the molecular motor, kinesin and unfolding of skeletal-muscle protein, titin.

5.1 Step Detection Methodology

Our step-detection methodology seeks to increase the detection of true steps, while decreasing the detection of false steps. To do so, we use an optimization based strategy, where, the quality of a fit is assessed via a cost function that exploits the nature of typical single-molecule data. Single-molecule data is obtained by sampling at significantly higher rates than inter-arrival times of steps. As a result, true steps in the data occur at few sample points of the data. Step detection methods face the challenge to find these step locations among numerous possible choices.

What are the metrics that aid in making these choices? A candidate $\hat{x}$ is considered as a good estimate of the true stepping signal x, if the percentage of true positives (% TP) is high and the percentage of false positives (% FP) is low. How do we quantify % TP and % FP of a candidate $\hat{x}$ without the knowledge of the true signal? A means of assessing % TP of a candidate $\hat{x}$ is the χ^2 error given by $\sum_k (y_k - \hat{x}_k)^2$, where y_k denotes the k^{th} sample of data y and $\hat{x}_k$ denotes the k^{th} sample of candidate $\hat{x}$. % FP of a candidate $\hat{x}$ can be assessed by the total number of steps in the candidate. Note that if the candidate $\hat{x}$

underfits the data (that is it has fewer steps than the true signal) then it is expected that the number of steps in $\hat{x}$ is also small leading to low % FP and also low % TP, whereas if it overfits the data (where the noise is interpolated by the candidate), then the χ^2 error incurred by the candidate is small, leading to high % TP but also high % FP.

To address the issue of simultaneously increasing % TP and lowering % FP (or, to avoid underfitting and overfitting), we first present a composite measure, $J(\hat{x})$, given by

$$J(\hat{x}) := \underbrace{\sum_{k=1}^{N}(y_k - \hat{x}_k)^2}_{\chi^2\ error} + \underbrace{W\sum_{k=1}^{N}\bar{\delta}(\hat{x}_k - \hat{x}_{k-1})}_{penalty}. \tag{5.1}$$

where N is the total number of samples and $\bar{\delta}(u) = 0$ if $u = 0$ and $\bar{\delta}(u) = 1$ if $u \neq 0$. Thus $\sum_{k=1}^{N}\bar{\delta}(\hat{x}_k - \hat{x}_{k-1})$ equals the number of steps in $\hat{x}$. We also call $J(\hat{x})$ the cost of the candidate $\hat{x}$ where, the parameter W characterizes the relative importance of % FP over % TP. A larger W places higher importance on reducing % FP. W can also be thought of as a penalty for introducing a step in the candidate.

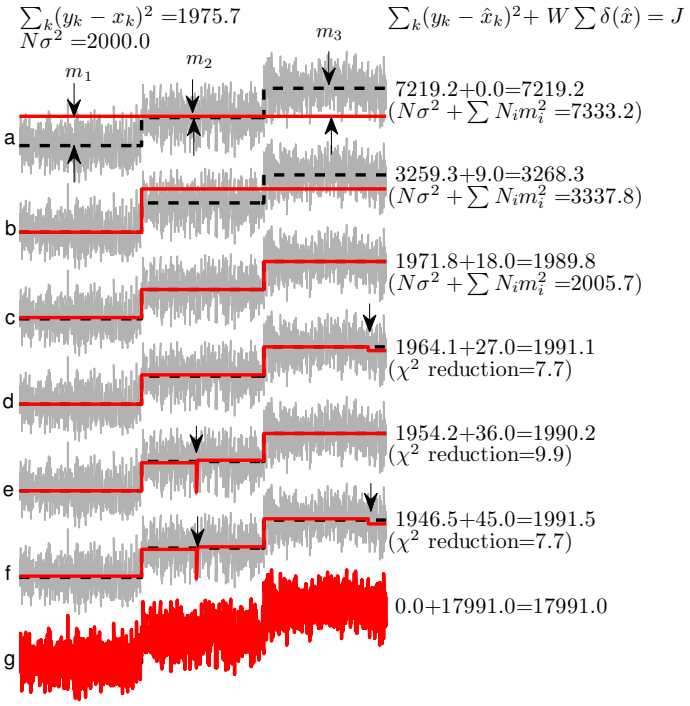


Figure 5.1: Comparison of cost, J of various step estimates (red). True step signal is shown in dashed black line that has two steps of unit magnitude. Simulated noise with $\sigma = 1$ is added to true signal to obtain noisy data (gray). The estimates were chosen to have 0,1,..,5,N steps (plots a to g) such that χ^2 error was minimized in each case. Cost of these estimates (shown to the right of the plots) is computed as the sum of χ^2 error and penalty, $W = 9$, on the number of steps in the estimate. The optimal number of steps is 2 and we see that (a,b) underfit the data and therefore have large χ^2 error leading to large cost. Theoretical estimate of the χ^2 error is shown within parentheses, which is the sum of χ^2 error of noise ($N\sigma^2$) and error between estimate and true signal(estimated as $\sum N_i m_i^2$). On the other hand, d,e,f,g overfit the data such that χ^2 error reduces by less than 9 (per step, on an average) when compared to c but accumulate a penalty cost of $W = 9$ with each additional step, thus increasing the total cost. As a result, c, which has optimal number of steps has the least cost and our optimal fit.

Our strategy is to choose W that favors the cost $J(x)$ of the true signal x to be the smallest amongst the costs, $J(\hat{x})$, of all candidates $\hat{x}$. If W can be so determined then the best candidate can be determined by choosing $\hat{x}$ that has the minimum cost, $J(\hat{x})$. A choice of $W = 9\sigma^2$ where σ is the standard deviation of the noise n_k is shown to avoid under-fitting and over-fitting. Thus with the appropriate choice of W, our step-detection methodology determines the candidate that achieves the minimum cost in (5.1). For example, Fig. 5.1 (a-g) shows seven candidates of the true stepping signal that has two steps. We note that (c) has the least cost of all the candidates. The costs are computed using only the data and the candidate function itself. As the candidate in (c) has the minimum cost, our step-detection methodology will yield candidate in (c) as the estimate of the true stepping-signal.

Indeed, the candidate in (c) does provide the best estimate (amongst (a-g)), when assessed by % TP and % FP, and step-sizes. We note that the candidate in (a) has no steps (and therefore under-fits the data with % FP=0) and the candidate in (g) (that over-fits the data) has the χ^2 error zero (with % TP =100).

In the example above, $W = 9\sigma^2$ did result in the candidate $\hat{x}$ that minimizes the cost $J(\hat{x})$ to be a good estimate of the true stepping signal. Does this choice of W lead to good estimates for typical single-molecule data?

To test this, we first estimate the cost $J(x)$ of the true signal, x, as follows: let us assume that there are d steps in the true signal. Also, $y_k - x_k = n_k$, where we assume n_k is zero mean Gaussian noise with variance σ^2. When the number of samples, N, is large $\sum_k (y_k - x_k)^2 = \sum_k n_k^2 \approx N\sigma^2$. Using this fact and Eq. 5.1, $J(x) = \sum_k (y_k - x_k)^2 + Wd \approx N\sigma^2 + Wd$. For the data presented in Fig. 5.1, the χ^2 error for the true signal x is 1975.7, whereas its estimate, $N\sigma^2$, as determined above, is 2000. Also with $W = 9\sigma^2 = 9$, $J(x) = 1984$.

Given a candidate, $\hat{x}$, that under-fits the data, how can we choose a W that makes the cost $J(\hat{x})$ larger than the cost of the true signal $J(x)$? In the example shown in Fig. 5.1, the top two candidates under-fit the data. The number of steps, $\hat{d}$, in those candidates is less than the number of steps, d, in the true signal. Given the measured data, y, and a candidate, $\hat{x}$, we can always partition the time axis into segments, where, in each segment, both the candidate, $\hat{x}$, and the true stepping signal, x, are constant. We suppose that there are r such segments with the i^{th} segment having N_i samples. The difference in the values of the candidate and the true stepping signal in the i^{th} is denoted by m_i. As the number of steps in the candidate and the stepping signal are small in comparison to the total number of samples, it is expected that N_i is large. For the sample index k in the i^{th} segment, $y_k - \hat{x}_k = y_k - x_k + x_k - \hat{x}_k = n_k + (x_k - \hat{x}_k) = n_k + m_i$ and thus $y_k - \hat{x}_k$ has the same statistics as the noise n_k with the mean shifted by m_i. Thus an estimate for the χ^2 error, $\sum_k (y_k - \hat{x}_k)^2$, incurred in the i^{th} segment with N_i samples is $N_i(\sigma^2 + m_i^2)$. Here also we have utilized the fact that N_i is large. Thus we can estimate that $J(\hat{x}) \approx \sum_{i=1}^{r} N_i(\sigma^2 + m_i^2) + \hat{d}W = N\sigma^2 + \sum_{i=1}^{r} N_i m_i^2 + \hat{d}W$ or $J(\hat{x}) - J(x) \approx \sum_{i=1}^{r} N_i m_i^2 - (d - \hat{d})W$. To ensure $J(\hat{x}) > J(x)$, we need to choose W such that $W < \frac{\sum_{i=1}^{r} N_i m_i^2}{d - \hat{d}}$. As the number of samples, N_i, in each segment is large (for candidates in Fig. 5.1(a,b) the smallest segment has greater than 650 samples), even a small m_i leads to a large $\sum_{i=1}^{r} N_i m_i^2$ and thus a value of $W = 9\sigma^2 < \frac{\sum_{i=1}^{r} N_i m_i^2}{d - \hat{d}}$ will be typically satisfied. The smaller the value of W, the easier it is to assign a larger cost $J(\hat{x})$ to the under-fitting candidate. However, even with a choice of $W = 9\sigma^2$, we can expect that underfitting candidate will have a larger cost than the true signal.

Given an estimate $\hat{x}$ that over-fits the data, how can we choose W that makes the cost

$J(\hat{x})$ larger than cost of the true signal $J(x)$? In this case, the number of steps in the candidate is larger than the true number of steps. For every pair of steps added to the true number of steps n, there is the potential to reduce the χ^2 error by interpolating another data point y_k. Noise, n_k, can take a value beyond 3σ with a probability less than 0.3 %. Thus an estimate on the reduction of the χ^2 error will be smaller than $9\sigma^2$ (as $y_k = x_k + n_k$). However, the cost of adding each extra step is W. Thus if $W \geq 9\sigma^2$, then the cost, $J(\hat{x})$, of an over-fitting $\hat{x}$ will be greater than $J(x)$. This choice of W also addresses the concerns of outliers, where the cost of interpolating an outlier (caused by $n_k \approx 3\sigma$) will be prohibitive.

Thus, with a choice of $W \approx 9\sigma^2$ there is high likelihood that the cost of the true stepping signal will be the smallest and the concerns of over-fitting, under-fitting and outliers are addressed.

5.1.1 Incorporating probe-dynamics

The preceding analysis provides and intuitive approach for fitting steps to a noisy signal where underlying signal is also assumed to be taking ideal steps. However, all real systems are band-limited, i.e., their response to a step input is not a step but a distorted and smoother version of it. Fitting a data that is the output of a smoothing filter is a more difficult task because it requires inverting the filter characteristics. In this section, a general framework for step fitting is developed that considers the filtering effect of a sensor probe. The mathematical formalism borrows ideas from statistical literature, in particular, maximum likelihood sequence estimation. Toward the end of this section, strong analogy between the preceding section's intuitive ideas and this section's mathematically rigorous problem development is established.

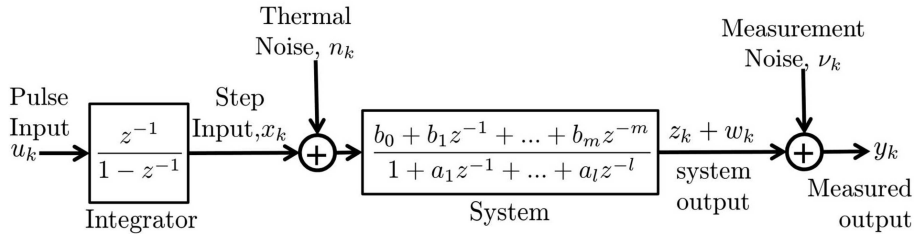


Figure 5.2: Discrete time input output model.

In general, the input output response is described by a dynamical model. A generic discrete time auto regressive moving average (ARMA) process is described by

$$p_k + a_1 p_{k-1} + \cdots + a_l p_{k-1} = b_0 q_k + b_1 q_{k-1} + \cdots + b_m q_{k-m} \tag{5.2}$$

where $p = (p_0, p_1, \ldots)$ represents the output signal of the measurement system due to an

input signal $q = (q_0, q_1, \cdots)$. k is the time/sample index. Thus, the measurement system processes or "filters" the input signal q and provides the output signal p. Fig. **5.2** shows a block diagram depicting various input sources and their filtering by the system dynamics. Represent the time shift operator by D, with $(D^r x)_k = x_{k-r}$, then the above dynamical model can be symbolically represented as

$$p = H(D)q, \text{ where } H(D) = \frac{b_0 + b_1 D + \cdots + b_m D^m}{1 + a_1 D + \cdots + a_l D^l}, \tag{5.3}$$

which is a representation analogous to the $\mathcal{Z}$-transform. The above description includes all systems that are linear, time invariant, causal and finite dimensional which is a large class of systems that include, for example, optical tweezers and AFMs. As an example, a model for optical tweezers in the above form is developed here, where, the measurement system dynamics in continuous-time is described by

$$\gamma \dot{x_b} + k\left(x_T - x_b\right) = f$$

where γ is the drag coefficient of the trapped bead, k is the trap stiffness, x_b is the bead position, x_T is the trap position and f is the external force on the bead. In studies of motor proteins using optical tweezers, external force, f has two components. One is the force applied by the motor, f_m and the other is random force, n' due to the thermal bath. f_m is generated when the motor is stretched and is assumed to admit the following description:

$$f_m = k_m(x_m - x_b)$$

where k_m is the motor stiffness and x_m is the position of the motor beyond its rest length. In controlled force studies, f_m is regulated at a constant value, f_0, by modulating x_T. The popular choice of control is to have $x_T = x_b - \frac{f_0}{k}$. This leads to the continuous-time dynamics

$$\gamma \dot{x_b} + k_m x_b = k_m x_m + f_0 + n'.$$

As f_0 is a constant, it only affects the steady state solution of x_b. Therefore, without loss of generality, the dynamics, once discretized in time, leads to the difference equation

$$\underbrace{x_{b,k}}_{p_k} = \underbrace{\frac{k_m T_s}{(\gamma + k_m T_s) - \gamma D}}_{H(D)} \underbrace{\left(x_{m,k} + \frac{n'}{k_m}\right)}_{q_k} \tag{5.4}$$

where T_s is the sampling time of the measurement system and discretization method used is described by the transformation $\dot{x} \rightarrow \frac{x_k - x_{k-1}}{T_s} = \frac{1-D}{T_s} x_k$. Thermal noise n' is a white

process with a power spectral density $4K_BT\gamma$ where K_B is the Boltzmann's constant, and T is the absolute temperature. Note that Eq. **5.4** has the same form as Eq. **5.3**. Note that q in Eq. **5.4** is the sum of two signals: signal of interest, $x := x_{m,k}$ and noise, $n := \frac{n'}{k_m}$. Using the linear dynamics of the system, p can be decomposed into two signals as

$$p_k = z_k + w_k \text{ where, } z = Hx \text{ and } w = Hn, \tag{5.5}$$

where z is the output of the measurement system when input is only the signal x, whereas w is the output of the measurement system when input is the noise, n. In addition, typically p is corrupted during measurements and thus the measured data is given by

$$y_k = p_k + \nu_k \tag{5.6}$$

where ν_k is also often modeled as a white noise process with known power that can be experimentally determined. As an application to step detection, additional modeling assumptions are made about the input signal, x, that it is a staircase function generated by a sequence of steps. Thus

$$x_{k+1} = x_k + u_k. \tag{5.7}$$

Note that without any further assumptions on u_k, Eq. **5.7** is merely stating that $x_{k+1} - x_k := u_k$ and thus poses no further constraints on the applicability of the model. The complete model of the measurement system is then described by Eq. **5.3**, **5.5**, **5.6** and **5.7** that is depicted in Fig. **5.2**.

5.1.2 Cost function derivation: MAP framework

Step detection methodology, as outlined int the main article, needs a cost function formulation for a candidate step function. The step function that minimizes the chosen cost function is the optimal estimate for a particle iteration of the algorithm. The formulation of cost function here is for a more general setting than what is presented in the main article. For the remainder of the article, stochastic variables will be represented by bold characters and the normal character will represent a particular realization of the corresponding stochastic variable. Summarizing the model description (Eq. **5.5-5.7**), it follows that

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \boldsymbol{u}_k \\ \boldsymbol{z}_k &= \sum_{i=0}^{m} b_i \boldsymbol{x}_{k-i} - \sum_{j=1}^{l} a_j \boldsymbol{z}_{k-j} \ , \ \ \boldsymbol{w}_k = \sum_{i=0}^{m} b_i \boldsymbol{n}_{k-i} - \sum_{j=1}^{l} a_j \boldsymbol{w}_{k-j} \\ \boldsymbol{y}_k &= \boldsymbol{z}_k + \boldsymbol{w}_k + \boldsymbol{\nu}_k. \end{aligned} \tag{5.8}$$

with the assumption that the noise process, $\boldsymbol{n}$ and $\boldsymbol{\nu}$ are zero mean Gaussian white noise with variance σ_n^2 and σ_ν^2 respectively. An intuitive strategy to identify the location and size of steps that occur in x, by analyzing measured data y, involves solving the following optimization problem

$$\begin{aligned}
&\min_{(u_0,u_1,\ldots,u_{N-1})} \sum_{k=0}^{N-1} \underbrace{(y_{k+1}-z_{k+1})^2}_{\text{Quadratic Error}} + \underbrace{W(u_k)}_{\text{Penalty}} \\
&\text{subject to :} \\
&x_{k+1} = x_k + u_k \ ; \ x_0 = 0 \\
&z_k = \sum_{i=0}^{m} b_i x_{k-i} - \sum_{j=1}^{l} a_j z_{k-j} \ ; \ z_0 = 0
\end{aligned} \tag{5.9}$$

where the data record $(y_1, \ldots, y_N)$ is available and $W(u_k)$ is a penalty term, that penalizes every nonzero choice of u_k. Evidently, the quadratic term strives to choose u such that the data y is interpolated well while the penalty term provides a means to isolate the effect of noise terms w and ν from the eventual fit, $(u_0, \cdots, u_{N-1})$ that also determines $(x_1, \ldots, x_N)$. $W(u_k)$ also provides a means to incorporate a relative preference of one size of the step over another. Taking as an example, the detection of steps in kinesin motor data, $W(u_k)$ can be shaped such that it is small when u_k is close to 8 nm and thus, these steps will be favored over other step sizes. In the algorithm developed here, such shaping is tuned automatically as an integral part of the detection methodology.

It is shown in this article that the heuristic approach above is similar to maximum likelihood sequence estimation, $\hat{x}_1^N$ of sequence x_1^N from given measurement sequence y_1^N, that is obtained by determining

$$\hat{x}_1^N = \arg\max_{x_1^N} p_{\boldsymbol{x}_1^N|\boldsymbol{y}_1^N}(x_1^N|y_1^N) \tag{5.10}$$

where, $x_1^N := \{x_1, x_2, ..., x_N\}$, $p_{\boldsymbol{x}}(x)$ denotes the probability density function (p.d.f.) of random variable $\boldsymbol{x}$, $p_{\boldsymbol{x}|\boldsymbol{y}}(x|y)$ denotes the p.d.f. of $\boldsymbol{x}$ given $\boldsymbol{y} = y$. Thus in Eq. **5.10**, the optimal choice $\hat{x}_1^N$ is sought that is the most probable input sequence given that the measured sequence is $y_1^N = (y_1, \ldots, y_N)$. Note that by applying Bayes' rule [46] it follows that

$$\hat{x}_1^N = \arg\max_{x_1^N} \frac{p_{\boldsymbol{x}_1^N,\boldsymbol{y}_1^N}(x_1^N, y_1^N)}{p_{\boldsymbol{y}_1^N}(y_1^N)} = \arg\max_{x_1^N} p_{\boldsymbol{y}_1^N|\boldsymbol{x}_1^N}(y_1^N|x_1^N) p_{\boldsymbol{x}_1^N}(x_1^N). \tag{5.11}$$

By further application of Bayes' rule, it can be shown that

$$\hat{x}_1^N = \arg\max_{x_1^N} \left\{ \prod_{k=1}^{N} p_{\boldsymbol{y}_k|\boldsymbol{y}_1^{k-1},\boldsymbol{x}_1^N}(y_k|y_1^{k-1}, x_1^N) p_{\boldsymbol{x}_k|\boldsymbol{x}_1^{k-1}}(x_k|x_1^{k-1}) \right\} \tag{5.12}$$

where

$$p_{\boldsymbol{y}_1|\boldsymbol{y}_1^0,\boldsymbol{x}_1^N}(y_1|y_1^0, x_1^N) := p_{\boldsymbol{y}_1|\boldsymbol{x}_1^N}(y_1|x_1^N)$$

$$p_{\boldsymbol{x}_1|\boldsymbol{x}_1^0}(x_1|x_1^0) := p_{\boldsymbol{x}_1}(x_1).$$

From the input-output model in Eq. **5.8** it follows that,

$$\boldsymbol{y}_k = \boldsymbol{z}_k + \boldsymbol{w}_k + \boldsymbol{\nu}_k$$

$$= \underbrace{\sum_{i=0}^{m} b_i \boldsymbol{x}_{k-i}}_{\bar{\boldsymbol{x}}_k} + \underbrace{\sum_{i=0}^{m} b_i \boldsymbol{n}_{k-i}}_{\bar{\boldsymbol{n}}_k} - \underbrace{\sum_{j=1}^{l} a_j \boldsymbol{y}_{k-j}}_{\bar{\boldsymbol{y}}_k} + \underbrace{\sum_{j=1}^{l} a_j \boldsymbol{\nu}_{k-j}}_{\bar{\boldsymbol{\nu}}_k} + \boldsymbol{\nu}_k.$$

From the last equation, it can be seen that if $\boldsymbol{y}_1^{k-1} = y_1^{k-1}$ and $\boldsymbol{x}_1^{k-1} = x_1^{k-1}$ then $\boldsymbol{y}_k = \bar{x}_k + \bar{\boldsymbol{n}}_k - \bar{y}_k + \bar{\boldsymbol{\nu}}_k + \boldsymbol{\nu}_k$. Therefore, the distribution of $\boldsymbol{y}_k$ is dictated by the random variable, $\bar{\boldsymbol{n}}_k + \bar{\boldsymbol{\nu}}_k + \boldsymbol{\nu}_k$. When $\boldsymbol{y}_k = y_k$, then $\bar{\boldsymbol{n}}_k + \bar{\boldsymbol{\nu}}_k + \boldsymbol{\nu}_k = y_k - \bar{x}_k + \bar{y}_k$. It follows that,

$$p_{\boldsymbol{y}_k|\boldsymbol{y}_1^{k-1},\boldsymbol{x}_1^N}(y_k|y_1^{k-1}, x_1^N) = p_{\bar{\boldsymbol{n}}_k+\bar{\boldsymbol{\nu}}_k+\boldsymbol{\nu}_k}(y_k - \bar{x}_k + \bar{y}_k) = \frac{1}{\sqrt{2\pi\bar{\sigma}_n^2}} \exp\left(-\frac{(y_k - \bar{x}_k + \bar{y}_k)^2}{2\bar{\sigma}_n^2} \right), \text{ where}$$

$$\bar{\sigma}_n^2 := \sum_{i=0}^{m} b_i^2 \sigma_n^2 + \sum_{j=1}^{l} a_j^2 \sigma_\nu^2$$

which follows from the fact that $\boldsymbol{n}_i$, $\boldsymbol{\nu}_j$ are Gaussian, i.i.d. (independent and identically distributed) noise sources. Note that in Eq. **5.12**, $p_{\boldsymbol{x}_k|\boldsymbol{x}_1^{k-1}}(x_k|x_1^{k-1}) = p_{\boldsymbol{x}_k|\boldsymbol{x}_{k-1}}(x_k|x_{k-1})$ and as $\boldsymbol{u}_{k-1} = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$, it follows that $p_{\boldsymbol{x}_k|\boldsymbol{x}_{k-1}}(x_k|x_{k-1}) = p_{\boldsymbol{u}_{k-1}}(x_k - x_{k-1})$. In the absence of a-priori knowledge and for the purpose of initializing the algorithm, a uniform distribution for $p_{\boldsymbol{x}_1}(x_1)$ is assumed. Likewise, it is assumed that $x_0 := x_1$ and $p_{\boldsymbol{u}_0}(x_1 - x_0) := 1$, from which it follows that,

$$\hat{x}_1^N = \arg\max_{x_1^N} \left\{ \prod_{k=1}^{N} p_{\boldsymbol{y}_k|\boldsymbol{y}_1^{k-1},\boldsymbol{x}_1^N}(y_k|y_1^{k-1}, x_1^N) p_{\boldsymbol{x}_k|\boldsymbol{x}_1^{k-1}}(x_k|x_1^{k-1}) \right\}$$

$$= \arg \max_{\substack{x_1^N \\ u_k = x_{k+1} - x_k}} \prod_{k=1}^{N} \frac{\exp\left(-(y_k - \bar{x}_k + \bar{y}_k)^2 / 2\bar{\sigma}_n^2\right)}{\sqrt{2\pi\bar{\sigma}_n^2}} p_{\boldsymbol{u}_{k-1}}(u_{k-1}) \tag{5.13}$$

$$= \arg \min_{\substack{x_1^N \\ u_k = x_{k+1} - x_k}} \sum_{k=1}^{N} \frac{(y_k - \bar{x}_k + \bar{y}_k)^2}{2\bar{\sigma}_n^2} - \log p_{\boldsymbol{u}_{k-1}}(u_{k-1}) \tag{5.14}$$

$$= \arg \min_{\substack{x_1^N \\ u_k = x_{k+1} - x_k}} \sum_{k=1}^{N} (y_k - \bar{x}_k + \bar{y}_k)^2 - 2\bar{\sigma}_n^2 \log p_{\boldsymbol{u}_{k-1}}(u_{k-1})$$

$$= \arg \min_{\substack{x_1^N \\ u_k = x_{k+1} - x_k}} \sum_{k=0}^{N-1} (y_{k+1} - \bar{x}_{k+1} + \bar{y}_{k+1})^2 + W(u_k)$$

$$= \arg \min_{\substack{x_1^N \\ u_k = x_{k+1} - x_k}} \sum_{k=0}^{N} h_k(x_{k-m+1}^k, u_k) + W(u_k) \tag{5.15}$$

where

$$W(u_k) := -2\bar{\sigma}_n^2 \log p(u_k), \;\; h_k := (y_{k+1} - \bar{x}_{k+1} + \bar{y}_{k+1})^2 .$$

h_k does not have explicit dependence on x_{k+1} because x_{k+1} is expressed in terms of x_k and u_k. The structure of cost function in Eq. 5.15 is similar to one in Eq. **5.9** for appropriate choice of weighting function W. Thus, the intuition driven optimization problem where the compromise between the accuracy of fitting data and fitting noise as posed in Eq. **5.9** is the same as the probabilistic setup of problem in Eq. **5.15**.

The objective now is to find a staircase function that minimizes the cost function in Eq. **5.15**. A straightforward approach is to compare the cost of all possible staircase functions. Computationally this is not feasible. Consider a measured signal that has N samples with each sample allowed to take any of the possible M values. Evidently, the total number of candidate sequences is M^N. As N can be a large number (for example $\sim 10^4$ in this article), even if $M = 2$, the total number of possible comparisons is intractably large (10^{3010}). Fortunately, the structure of the cost function lends itself to a form where a solution approach, based on the dynamic programming, can be applied which greatly reduces the number of computations (less than 10^8). As shown later, computational complexity of this approach is NM^{m+1}, where m is the same as in Eq. **5.8**. This is a significant reduction in the number of computations. Subsequently, other techniques are developed to further

reduce the computational complexity that render the step detection method applicable to single molecule studies based on optical tweezers or AFM.

The next question is how can one obtain a model for the probe. It is not always possible to obtain a dynamical model from physical principles as utilized above. Nonetheless, there are several methods available that provide a model to predict probe response for a given input. A frequently employed method is to obtain the power spectrum of the data sensor/probe when it is being excited by thermal noise with known statistics. By fitting a model to the power spectrum, the dynamics can be estimated. For example, in the case of optical tweezers, thermal noise is a white noise source with a known spectral density $S_\eta = \frac{1}{k}\sqrt{4k_BT\gamma}$ (k_B is Boltzmann constant, T: absolute temperature, γ: is the damping constant and k: trap stiffness). The corresponding noise deviation is $\sigma = \frac{1}{2}S_\eta F_s$ where F_s is the sampling frequency. Assuming a first order model for optical tweezers of the form $S_y^2(f) = S_\eta^2\frac{1}{\tau^2 f^2 + 1}$, where f is the frequency in Hz, the fitting procedure will provide τ. In Fig. 5.9, dynamics and noise values are estimated by fitting these parameters such that simulated noise spectrum matches the experimental noise spectrum. τ is the time constant determined by the time taken for the sensor to reach approximately 63% of the input step amplitude (rise time). Therefore, τ can also be estimated by experimentally giving a known step and measuring the rise time. A general approach to model identification entails giving known input and observing its output and then fitting models to the observed input-output behavior. A commonly employed input waveform is a chirp or a frequency sweep wherein a sinusoidal wave is generated whose frequency increases steadily with time. If the system is linear, its output will also be sinusoidal, but with an amplitude and phase delay that varies with frequency. By measuring the amplitude amplification and phase delay for every frequency, linear models (that describe some differential equation) can be fitted using tools available in software like MATLAB. The fitted model can then be used to predict the output for an arbitrary input and possibly to estimate input from output. Such methods can be used to obtain and validate the model purely experimentally and the resulting models will include the effects of instrumentation dynamics as well[47].

5.2 Evaluation of step detection algorithm

5.2.1 Evaluation with simulated data

Unimodal stepping:simulated transport by a single kinesin motor

Cargo transported by a single kinesin motor changes position by 8 nm with every step of the motor along the microtubule lattice. The resultant step-size distribution is unimodal. We simulated such a distribution by generating 8 nm steps with arrival times that follow

Poisson distribution. White noise with a standard deviation, σ, of 5 nm was added to model thermal noise. Our step-detection methodology was employed to detect the timing and magnitude of steps in the simulated data. Fig. 5.3 shows the steps detected with initial penalty on steps being $W = 9\sigma^2$. This yielded an estimate which was used to obtain a step-size histogram. This new step-size distribution yielded another estimate and the process was repeated to obtain new step-size distributions and stepping signal estimates. The convergence of histograms is quick and typically takes less than 10 iterations to converge. At the end of iterations, the final histogram shows a sharp peak at 8 nm step size.

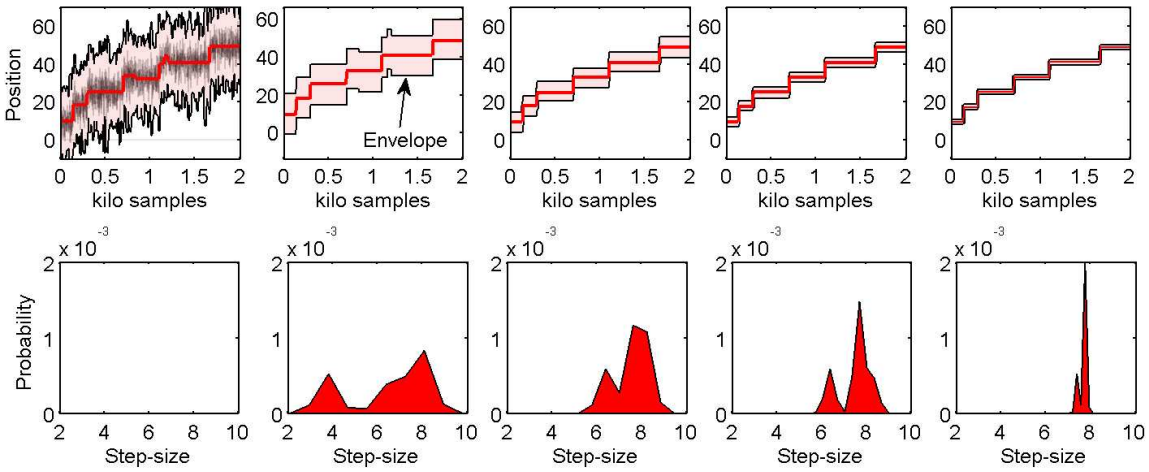


Figure 5.3: Iterations of step fits (left to right). Top row shows the step fits (red) and the envelope (shaded region) within which the fit was constrained. As iterations progress, the envelope is made narrower for better accuracy. Bottom row shows the smoothed step-size probability distribution obtained from the step size histogram of previous iterations fits. A Gaussian FIR filter is used for smoothing that reduces bias in the estimation of distribution. In the first iteration a constant weight on step sizes is used instead of a probability description hence it is shown blank. The smoothing level is reduced over iterations to get sharp histograms.

Fast and slow step combinations

Whether the kinesin motor takes sub-steps is still being debated (see [48]). Coppin and coworkers[39] suggested that kinesin takes a 3 nm step rapidly followed by another 5 nm step. Similarly, Nishiyama eta al.[49] suggested that kinesin takes two 4 nm steps in rapid succession. As the dwell time of the first step is small, it is especially challenging for current step detection methods to distinguish two 4 nm steps from a single 8 nm step. We asked whether our step-detection methodology can resolve such sub-steps. Fig. 5.4 shows a simulation for a 3+5 nm step combination, where the dwell time of the 3 nm step is 0.6 ms (the data was sampled at 20kHz). Over 80% of the 3 nm steps were correctly identified. The step size histogram converged closely to the true step sizes. In contrast, step-size

histograms of the χ^2 method did not show a distinct bi-modal distribution, even when information about the total number of true steps was provided. Our algorithm shows a rapid decrease in the detection rate as the dwell time of the initial step is reduced, indicating that a minimum dwell time is required for reliable detection of substeps.

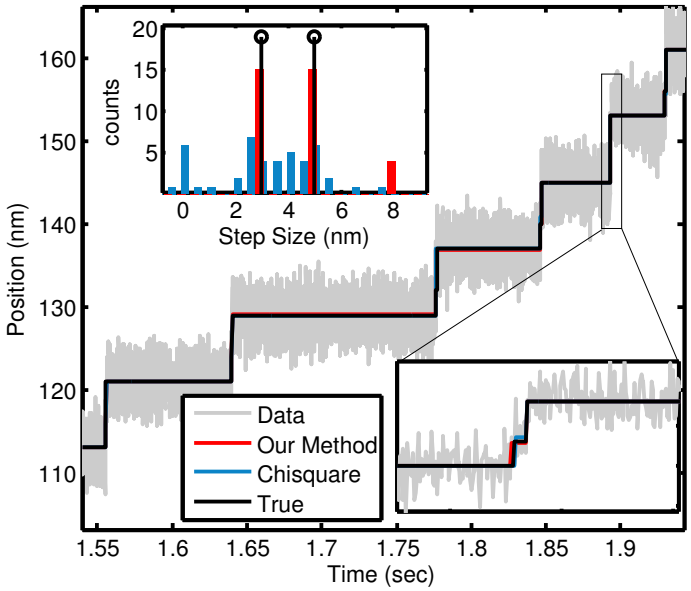


Figure 5.4: Combination of fast 3 nm and slow 5 nm steps is generated (black trace) with noise of SD 2 nm (gray trace). Dwell time of the 3 nm step is chosen to be 0.6 ms. Dwell time of the slow step is random. From visual inspection the data appears to be composed of 8 nm steps only. However, our method (red trace) instead is able to correctly detect the 3 nm and 5 nm step sizes (inset). χ^2 method (blue trace), when provided with the information on the total number of true steps yields the histogram shown in blue (inset) which shows that our method significantly outperforms χ^2 method. When the total number of steps in not provided, the χ^2 method predominantly detects the 8 nm steps.

Similar step sizes

A challenging problem in step detection is the resolution of steps of similar sizes. Since our algorithm is based on histogram iteration, similar sized steps may get diffused in the histogram smoothing process, leading to fusion of two or more step sizes into single averaged step size. To test the effect of smoothing, a simulation was performed with three step sizes, 3 nm, 4 nm and 5 nm, occurring independently in the stepping signal. The three step sizes were easily resolved when using a noise level of 2 nm and sampling rate of 10 kHz (Fig. 5.5). χ^2-method was not able to distinguish the three steps. At higher noise levels ($>$ 2 nm), the individual histogram peaks were either dislocated and/or additional spurious peaks appeared. In general, if the noise level is σ then our method can distinguish step sizes that differ by more than $\frac{\sigma}{2}$. However, this estimate also depends on sampling rate

and dwell times as discussed below in the section, 'Trade-off between SNR and dwell time'.

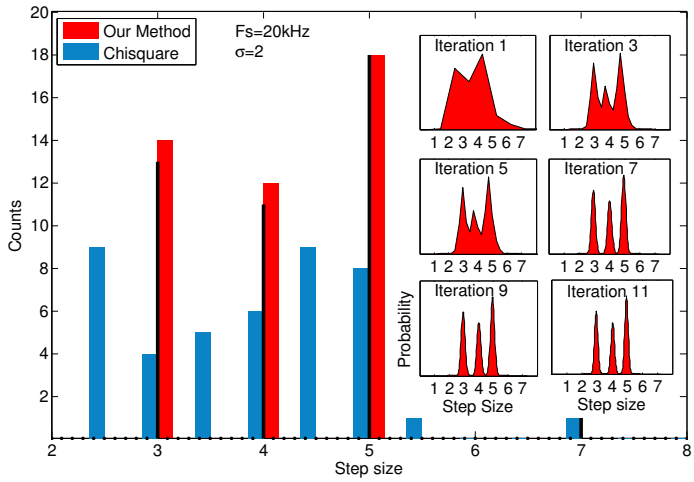


Figure 5.5: Stochastic stepping with mixed step sizes of 3,4 and 5 nm is analyzed from data that has noise with SD 2 nm. Our method (red) is able to resolve these step sizes distinctly as evident in the step-size histogram. Insets show the progress of the algorithm iterations as the stepping probability is updated. Initial histogram is broad and steps sizes are not well resolved. However, in just a few iterations, distinct peaks appear and towards the end of the iterations, separated peaks are obtained. For the same data, χ^2 method's estimates are spread out and incorrectly distributed .

Distributed step sizes

We next tested efficacy of our step detection methodology when there is no predominant step size that repeats or appears frequently in the data. In such a scenario, if the SNR is high, then most steps are identified and the distribution is reconstructed fairly well. If, however, SNR is low, then our algorithm will converge to multiple step sizes with gaps between them in the histogram (see Fig. 5.6a). Nonetheless, if the histograms are strongly smoothed before updating the penalty on steps, then the resulting histogram reflects a broader distribution(Fig. 5.6b). Therefore, our algorithm cannot predict whether underlying step-size distribution is continuous or discrete. However, if this information is available, then our algorithm can be suitably modified to improve its detection capability.

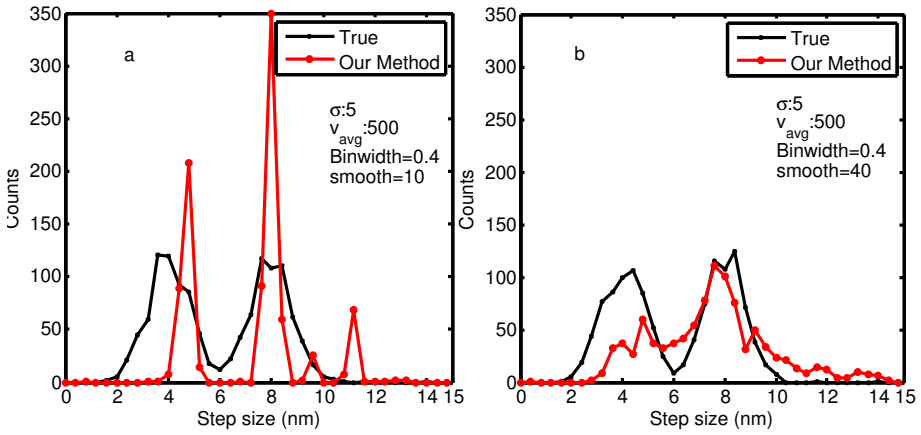


Figure 5.6: Histogram comparison for distributed step size test. Stochastic stepping with average velocity of 500 nm/sec and random step sizes was corrupted with noise of SD 5 nm and analyzed by our method. (a) Actual step size distribution (black trace) is broad with peaks at 4 and 8 nm. Our method (red trace) concentrates most of the steps at the 4 and 8 nm due to its preferential treatment to higher probability steps. (b) In this case, our method was modified to artificially smoothen the histogram for computation of step-size probability. The parameter 'smooth' in the figure refers the to the spread of the Gaussian filter applied to the histogram. The resulting step-size histogram reflects this as reproducing the distributed nature of the step sizes.

Dynamics compensation

The efficacy and limitations of our step detection methodology in addressing probe-dynamics was assessed for three cases. In the first case, a train of noisy 5 nm steps is filtered by the slow dynamics of the probe using cutoff frequency of 20 Hz. The steps are hard to discern (gray trace in Fig. 5.7a) by eye. In addition, χ^2 method clearly fails to identify correct step size and location even when the information on the true number of steps is made available. In contrast, our method uses knowledge about probe dynamics to compensate for the dynamics and reveal the stepping signal (Fig. 5.7a).

In a second test case, a square wave input is generated with decreasing dwell time. Noise is added to this signal and then passed through a second order filter, which has a clear resonance. Our method is able find the hidden steps with a good estimation of the step sizes that generated the observed data. The χ^2 method fails to do so (Fig. 5.7b).

In the third case, we evaluated the capability of detecting spikes or impulses, a combination of a rising step quickly followed by a falling step of the same size. Noise was added to such spikes and passed through a first order filter. The resulting signal looks highly distorted (gray trace in Fig. 5.7c). Our method handles such data without any modification if provided with the correct filter model. This kind of data is much more difficult to analyze and noise has severe adverse effects compared to other stepping signals. χ^2 method gives

no indication of spikes in the data.

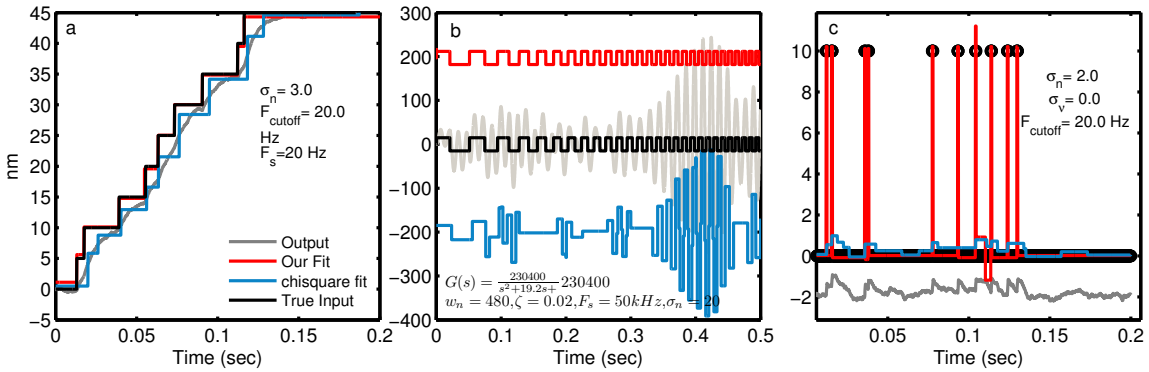


Figure 5.7: Dynamics compensation feature tested in simulations for different scenarios. (a) Stepping train of 5 nm is generated with stochastic dwell times. Noise of $\sigma = 3$ nm is added to the step signal and passed through a low pass filter with cutoff at 20Hz. The filtered noisy signal is analyzed by our method(red trace) and the χ^2 method (blue trace). Our method is able to underlying step signal with good accuracy and the histogram (inset) shows most steps were 5 nm except some steps (that had small dwell times) were identified as 4 nm instead. On the other hand, χ^2 method fits steps disregarding dynamics effects hence all the identified steps are spurious. (b) A second order dynamics effect is tested. Square signal is passed through a filter that amplifies certain frequencies and therefore we observe amplified oscillations and overshoots for square steps (it is not due to noise). The input signal has moderate amount of noise, SD=4. Under these conditions as well, our algorithm finds steps correctly with histogram (inset) of step sizes matching the true one. χ^2 method instead shows a variable step size as different stepping frequencies have different amplification dictated by the dynamical model. (c) Spikes in the data are also detected by our method under moderate noise assumptions. Impulses generated by a rising step rapidly followed by a falling step are filtered via first order dynamics. From the resulting trace (gray) , it is difficult to make out all the steps and their magnitudes. Our method does much better job of detecting step location and their magnitudes (see histogram in inset). χ^2 method instead tries to fit steps to the observed data and fails to identify the impulsive inputs.

Non stepping data

An important aspect of step detection is to discern the difference between stepping and non-stepping data. When presented with a smoothly varying signal with added noise, our method tries to fit minimum number of small steps, such that the χ^2 error is nearly the same as the true variance. As discussed in a later section, beyond a limit, small step sizes and fast arrival rates are not handled well by our algorithm and it tends to fit steps that are not actually present as in Fig. 5.8a. Similarly, other algorithms also fail to accurately detect the steps. However, we propose a method to evaluate the quality of fit in terms of confidence that underlying data has steps.

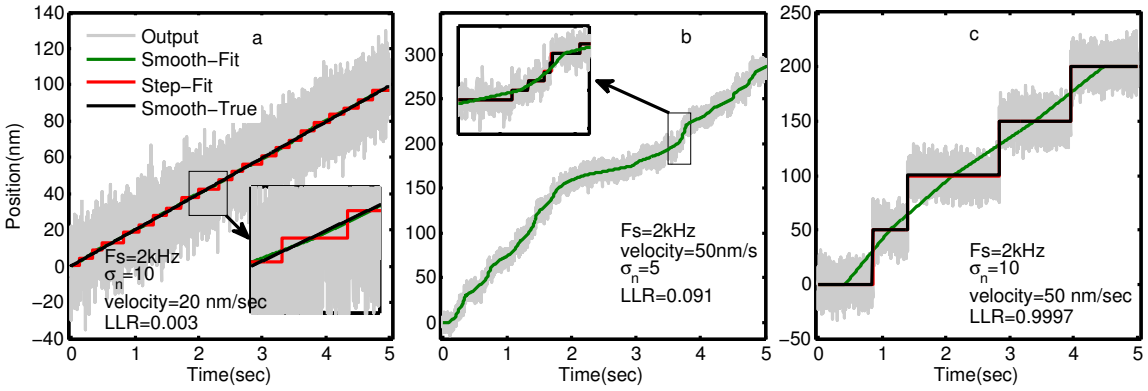


Figure 5.8: Plots here compare the average log likelihood ratio (LLR) for different types of data. LLR compares the likelihood of the observed data being originated from stepping action against a smoothly varying motion. (a) Data (gray) is originating from a smooth signal (black). Our algorithm fits a step signal to it with distinct steps. By connecting the plateaus of the steps, a smooth signal is generated (green). This smooth signal closely matches the true smooth signal (see inset). By comparing the χ^2 error for the smooth signal (green) against that of the step fit (red), LLR can be obtained. Smaller LLR indicates that a smooth signal may fit the data as well as a stepping signal. (b) Underlying the data is a stepping signal but not evident by looking at the data. An LLR of 0.09 indicates the underlying data is better explained by stepping signal rather than a smooth signal. (c) Steps are evident from the data itself. The corresponding LLR is also huge which is a confident measure of underlying signal being a stepping signal.

5.2.2 Evaluation with experimental data

Kinesin experiment

We used a bead-motility assay to collect experimental data on kinesin steps obtained under a constant load provided by optical tweezers. The sample preparation is similar to that described in [50] and is discussed briefly in the 'Methods' section. The experimental setup is similar to [9] and instrument calibration was done as described in [51]. There is drift in the data and extrinsic disturbances also introduce significant non-Gaussian noise. In this regard, the experimental data provide a good test of the robustness of our method under non-ideal conditions. We compared the fit of the steps to the experimental data using the χ^2 method, as well as our new method. It was determined that the cargo was carried by a single kinesin molecule and thus it is expected that the cargo steps by approximately 8 nm with deviations expected due to experimental uncertainties. The χ^2 method detects step sizes that are broadly and asymmetrically distributed about 7.5 nm spanning 6-10 nm range, whereas our method results in a predominant peak near 7.5 nm (see Fig. 5.9a). These results are consistent with 8 nm step-size assumption. Other peaks arise possibly due to drift and disturbance effects on the bead. Some negative steps near 7.5 nm were also observed and may be attributed to kinesin slippage under load [52].

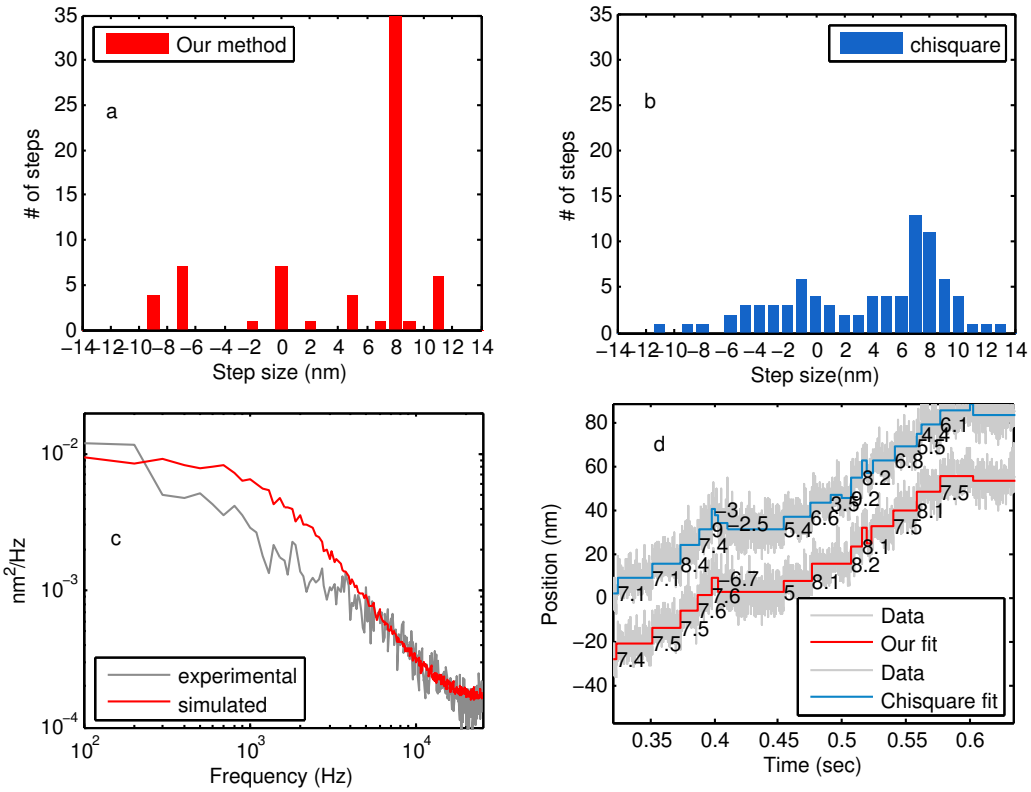


Figure 5.9: Fitting on experimental data obtained from kinesin-bead assay. (a) Histogram of the step sizes obtained by using our method. (b) Histogram of step sizes obtained by using chi-square method (number of steps was constrained to be equal to that obtained by our method. (c) Experimental power spectrum (gray) was obtained from a portion of data that did not contain any steps on visual inspection. This was fitted with simulated power spectrum of filtered noise (red) for a thermal noise level of 15 nm, measurement noise of 1.5 nm and cutoff frequency of 600 Hz. The dynamical model obtained from this fit was provided to our algorithm for fitting. The fit is not good representing deviations from assumed Gaussian noise statistics. Expected cut-off frequency for a stretched kinesin linkage is much higher but due to large extrinsic noise. Therefore, a conservative model (simulated power spectrum should be above experimental spectrum) is a better choice to avoid fitting spurious steps. (d) Step fits, using our method (red) and chi square method (blue) on experimental data (gray). Fits look similar but histograms differ considerably. Our method gives a strong peak around 7.5 nm in contrast to a broad distribution given by chi-square method. Deviations from expected 8 nm step size is attributed to experimental uncertainties, and external noise sources including drift and vibrations and electrical line noise that do not fit well to assumed Gaussian statistics.

In addition to optical tweezers data, we collected AFM experimental data on the unfolding behavior of the muscle protein, titin this data was particularly useful to validate the step detection method and its ability to compensate for probe-dynamics. The 27^{th} immunoglobulin-like module of cardiac titin has 8, nearly identical, domains that unfold when a force is applied [53]. In a typical AFM experiment, the molecule's ends are attached

between AFM cantilever tip and the substrate that sits on a piezo actuated positioning system. The separation between the substrate and the cantilever tip is controlled to maintain a constant tension in the molecule. When the protein unfolds, it relaxes quickly and the instrument responds by pulling the molecule to maintain a constant force. However, the response of the system to the unfolding event takes a certain time to settle and consequently the transition of the cantilever state that results from a protein conformation change is smooth, unlike the sharp unfolding event itself. If the unfolding events occur rapidly then, the protein configuration changes while the system is reaching its new equilibrium position, determined by the effects of the previous unfolding event of the protein. This is indeed the case with our experimental data shown in Fig. 5.9. We hypothesized a simple first order model for the system behavior in response to an unfolding event. The model was estimated offline using only one step response (see Methods) and was used to correct for the smoothing process introduced by the dynamics. An estimate of 24 nm for the unfolded protein module length was obtained as shown in the histogram of Fig. 5.9. The χ^2 method finds fewer 24 nm steps, corresponding to those steps that have large dwell times. Currently, researchers discard experimental data, that is distorted due to probe dynamics which current methods cannot incorporate. With our method, such data can now be retained and analyzed. Consequentially, the range of loads under which protein folding and unfolding is investigated can be increased. This is because under increased loads, domains unravel faster whereby the effect of an unfolding event can overlap with another unfolding event. With our step detection methodology, steps can be recovered from such data.

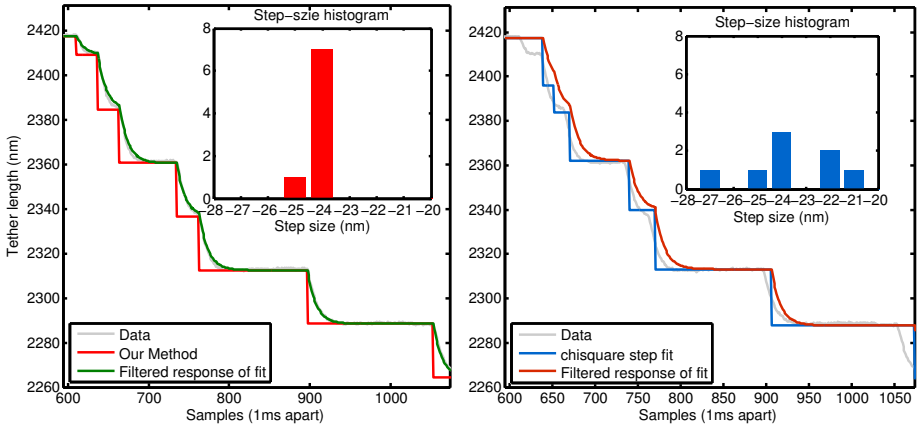


Figure 5.10: Dynamics compensation feature of our method for titin pulling experiment. The titin pulling experiment using AFM results in data that has steps but sharp transitions are smoothed due to limited response time of instrumentation and/or the sample itself. The response time was estimated from the data itself by inspecting one of the steps that has large dwell time. This response time was provided to the algorithm for fitting and noise was also estimated from the stationary portion of the data. The resulting fit is relatively accurate finds steps of 25-25 nm. A small step (in the initial portion of the data) is experimental artifact of AFM engaging with the sample. In contrast, χ^2 method only identifies steps that have large dwell time. Location of identified steps is clearly erroneous and fast steps are incorrectly estimated in its size as well. As a result, multiple step sizes are estimated instead of a uniform 24 nm steps.

AFM Experiment

A sample of solution of the engineered octamer of I27 (the 27th Immunoglobulin-like module of cardiac titin) was deposited for 20 minutes on the surface of a freshly exposed template stripped gold [54]. This is then placed inside a AFM liquid cell and the surface was rinsed with Phosphate Buffered Saline (the same buffer of the specimen). After optical realignment and thermal equilibration of the system, force-clamp AFM experiments were performed applying on the protein a pulling force of 110 pN. The details of the instrumentation are reported in [55]. The dynamics in the titin stretching experiment was assumed to be governed by the ordinary differential equation given by $\tau\dot{y} + y = x$, where x is a staircase function representing the length of the polymer in its current (partially) unfolded state at equilibrium, y is the measured extension of the piezoelectric actuator plus the cantilever deflection and τ is a time constant fitted to match the last step in the curve (that is usually an isolated unfolding event). τ was determined by giving a step input to the system and recording the time that it took for the system to reach 63% of the commanded step input.

5.3 Limitations

Distinguishing between stepping and non-stepping data

To determine whether observed data contains discrete steps or reflects a continuous trajectory is particularly challenging when steps occur at a fast pace and step-sizes are small with respect to the standard deviation of the noise. Under these condition, step-detection methods, including our own will show steps even when there might be no steps in the true signal.

Here we provide a measure to guard against the false interpretation of steps. We compare the fit by discrete steps(denoted by $\hat{x}$) to a fit by a smooth signal(denoted by $\tilde{x}$). The smooth signal, $\tilde{x}$, is generated from $\hat{x}$ by joining the mid-points of two subsequent step-plateaus thereby resulting in smoothing of discrete stepping signal. Simulations show that if the underlying signal is smooth, then the smooth fit to the data, determined from the estimate produced by the step-detection methodology will provide a good match to the true smooth signal (Fig. 5.8a).

To further determine the quality of a step fit with respect to a smooth fit, we compute the ratio of probability for a step fit $\hat{x}$ over that for a smooth fit $\tilde{x}$. The larger the logarithm of the ratio (called log-likelihood ratio (LLR)), the greater the confidence that the stepping is accurate reflection of true signal. Conversely, a negative or small LLR indicates that the data is equally likely to result from a smooth continuous trajectory. LLR is defined as follows

$$\frac{P(\hat{x})}{P(\tilde{x})} = \frac{\prod_{k=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_k - \hat{x}_k)^2}{2\sigma^2}}}{\prod_{k=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_k - \tilde{x}_k)^2}{2\sigma^2}}};$$

$$\log P(x) - \log P(\tilde{x}) = \frac{1}{2\sigma^2}\left[\sum_{k=1}^{N}(y_k - \tilde{x}_k)^2 - \sum_{k=1}^{N}(y_k - x_k)^2\right]$$

$$\text{LLR} := \frac{1}{N}\left[\log P(x) - \log P(\tilde{x})\right]$$

where $P(\hat{x})$ is the probability of step-fit, x. $P(\tilde{x})$ is the probability of the smooth fit, $\tilde{x}$. LLR greater than 0.08 should typically indicate a stepping behavior and LLR less than 0.05 is likely to mean the fit is not reliable and underlying fit could be smooth. Fig. 5.8 compares the LLR of a fit for a smooth signal vs. a stepping signal for different scenarios.

Trade-off between SNR and dwell time

There is general agreement that the ability to detect steps depends on the signal-to-noise ratio (SNR), that is the ratio of step-size to the standard deviation of the noise. When the SNR is small, steps are harder to detect with accuracy. Another, less referenced factor in resolving steps is how the dwell-times, or equivalently the number of samples obtained for a step, will affect the detection capability. Consider, for example, a stepping signal, x, where there is single step of size m in the middle of data that has N samples. Given that $y_k = x_k + n_k$ where noise n_k has zero mean with a standard deviation σ, then the SNR is $\frac{m}{\sigma}$. Also, as discussed earlier, the cost $J(x)$ of the true signal x is such that $J(x) \approx N\sigma^2 + W$, assuming a constant penalty for every step irrespective of step-size. Let us consider an estimate $\hat{x}$ that has no steps with a constant value equal to the mean of the data given by $\frac{m}{2}$. If we assume that $W = 9\sigma^2$, then we have $J(\hat{x}) = \sum_k (y_k - \hat{x}_k)^2 \approx N\sigma^2 + N\frac{m^2}{4}$. We would like the true signal to have the smallest cost and therefore we desire $J(\hat{x}) - J(x) = N\sigma^2 + N\frac{m^2}{4} - N\sigma^2 - W = N\frac{m^2}{4} - 9\sigma^2 > 0$. Thus we would like $N > 36(\frac{\sigma}{m})^2 = 36\frac{1}{SNR^2}$. This relationship, although illustrated on an instance of an estimate $\hat{x}$, holds qualitatively in general. Thus the number of samples required between steps, for good performance of our step-detection methodology has to be greater than $1/SNR^2$.

The quantitative relationship between the SNR, dwell-time and detection capability for step-detection methodologies is typically not assessed in the literature. Here, we provide such an analysis. We perform step detection using a square wave signal that has gradually decreasing dwell time (gray trace in Fig. 5.11a). It is similar to a sinusoidal frequency sweep, except the sine wave is replaced by a square wave. Several such simulations are performed for a given noise level and the resulting step fits are averaged (red trace in Fig. 5.11a). We observe that for longer dwell times (low frequency data) the averaged step-fits look like the original square waves as expected However, for shorter dwell times, the square wave amplitude of the averaged fit is reduced. This happens because, on an average, steps with shorter dwell times were missed. The dwell time beyond which the local amplitude curve drops (cutoff frequency, Fig. 5.11a) below a chosen threshold (85% of amplitude), the minimum dwell time required to detect a step can be determined. The corresponding SNR is the peak-to-peak amplitude of the original square wave divided by the known noise standard deviation. Hence the dwell-time and SNR pair is obtained for different noise levels and a graph is plotted (Fig. 5.11b). Given one of the quantities, the limit on the other can be predicted from the graph.

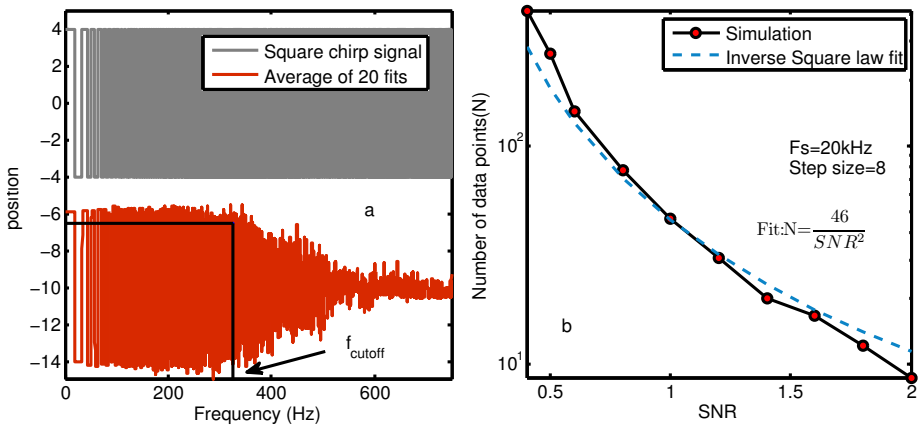


Figure 5.11: (a)Top trace (black) is a square chirp, a square wave with linearly decreasing dwell time. Bottom trace (red) is an average of 20 fits, obtained by running our algorithm on square chirp signal with noise added. The algorithm is able to detect steps with larger dwell time (low frequency square wave) in every simulation thus average fit has full amplitude, but steps with smaller dwell time (high frequency steps) are often missed out and therefore the average of the fits has diminished amplitude. By observing frequency after which amplitude drops below a threshold (here the original amplitude is 8nm and cutoff threshold is chosen to be 7nm), we can estimate the stepping frequency beyond which a fit is unreliable. The abscissa lists the frequency of the square wave. One square wave has two steps therefore the stepping frequency is twice the number obtained from this graph. By normalizing the sampling frequency (samples/s) with stepping frequency (steps/sec) we can obtain the required number of samples per step. This number is plotted against SNR in (b). (b) Black solid line is the number of samples points required to detect steps in reliable manner for a given signal to noise ratio (SNR). An inverse square law relationship is observed, evident from the fit (blue dashed line). The constant of proportionality (46) reflects the penalty on the steps. Bigger number would mean larger penalty. This graph can be used to predict whether a step of a given size and dwell time will be detectable under a given SNR in a reliable manner. For example, for a sampling rate of 10kHz, we wish to know the minimum dwell time of a 2nm step will be detectable when the noise is also 2nm. This corresponds to SNR=1. From the graph, approximately 45 samples per step will be required. This corresponds to a dwell time of $\frac{45}{10^4 \text{Hz}} = 4.5$ ms.

Computational costs

The straightforward implementation of our method can be computationally expensive. This reflects the large search space required for the many possible step functions and the need to compare their respective costs ($J(\hat{x})$). In the dynamic programming approach, candidate step functions are constrained to take values along closely spaced grid lines. Candidate step functions are also allowed to take steps at any sample point. The computational complexity has exponential dependence on the number of grid levels, M and number of sample points, N. Then the total number of computations can be estimated to be M^N. For example, if 1 sec of data ranges from 0-1000 nm and is sampled at 10 kHz, then a

grid resolution of 1 nm corresponds to $N = 10^4$, $M = 10^3$. Therefore, the number of computations required is 10^{30000}. This is a huge number. Dynamic programming technique can reduce the computational complexity to NM^2. For the example above, this computes to 10^{10} which is tractable. However, many of the computations are unnecessary because true signal is expected to interpolate the noisy data which span a small number of grid points. By eliminated computations for grid points that are unlikely to be a part of the step-fit, we can significantly reduce the computational complexity further. For example, if the noise level (σ) in the data is 5 nm, then one can expect that grid points lying beyond 3σ of the local mean of the data will not be a part of the step-fit. Therefore, we can bound the data in an envelope of width approximately $6\sigma = 30$ nm, within which we search for an optimal fit. In this case, $M = 30$ and the total number of computations is of the order of 10^7. However, during our iterations, we start with crude fineness to the data space such that $M = 50$. In subsequent iterations, the fineness is increased by reducing the envelope width but keeping the same M. This allows for better accuracy without taxing computations. Another level of efficiency is achieved because the computations can be parallelized without a loss in performance. Therefore, the methodology can take advantage of multi-core processors and the parallel computation features of MATLAB. Our implementation is on a quad-core computer (2.5 GHz),this brings additional speed acceleration by 4 times. Table 5.1 shows computation time required for different sample lengths. Given our code code optimization, the implementation of our method runs fast, in fact much faster than the simple implementation of χ^2-method (for larger data sets, $N > 10^4$).

Sample Length	10^3	5×10^3	10^4	5×10^4	10^5	5×10^5
Our method	2	9	17	32	54	242
χ^2 -method	0.6	6	7	30	60	305

Table 5.1: Mean computation time (in seconds) variation with data length. Sampling rate=10kHz. Noise σ=5nm. V_{avg} =500nm/s. Number of iterations=8. Final grid resolution less than 0.2nm. The nonlinear relation between sample length and computation time is likely due to the parallelization of the code into 4 cores. Our implementation of χ^2-method is much faster than provided by the original authors of the method, and also parallelized. The reported times are for our implementation of optimized χ^2 method. We observe that scaling of computational complexity of our method with number of samples is slower than that of χ^2-method. Times scales being comparable for the two methods indicates our method can be utilized for practical datasets.

A metric on performance

Quantities that are commonly employed to evaluate detection methods are TPR (true positive rate) and FPR (false positive rate). TPR is the percentage of true steps that are correctly identified by the algorithm. FPR is the percentage of false steps that are identified by the algorithm and are not actually present in the data. The computation of these metrics is discussed in the 'Methods' section. None of these quantifiers, by themselves, is a good measure of the performance of the algorithm. Thus, a more comprehensive measure is needed. The Receiver Operator Characteristic (ROC) [56] provides a reasonable means to compare different methods. For a given algorithm, the FPR and TPR values can be determined and the coordinates (FPR,TPR) placed on a graph of TPR vs. FPR with the axis limits for both FPR and TP between 0 and 100. The directed distance of the (FPR,TPR) point from the diagonal line joining the point (0,0) to the point (100,100) provides a good measure of the method's performance. The measure (the ROC measure) is positive if it lies above the diagonal line. The largest positive distance from the line is the point (0,100) which corresponds to FPR=0 and TPR=100. Also for a fixed FPR value, the higher the TPR value the better the measure will be. Similarly, for a fixed TPR value, the lower the FPR value, the higher is the value of the measure (the distance from the diagonal line will be higher). Thus this measure has most of the characteristics desired for a quantifier. A single performance measure, a metric defined using Receiver Operator Characteristics (ROC) was employed for this purpose. Each data point was generated by simulating approximately 100, 8 nm steps with an average velocity of 500 nm/s (for comparison purpose, no filters are applied to simulate probe dynamics). Various noise levels were added to the input ranging from a standard deviation of 1 nm to 8 nm with a sampling rate of 2 kHz. The performance measure was compared for different step detection methods for different output noise levels as shown in Fig. 5.12. The ROC analysis reveals several interesting properties. (1) Performance for all methods is reduced with increasing noise, however our method is least affected. (2) The ROC plot has low variance which means it provides consistent results over various simulations, providing better reliability. (3) This metric effectively compares and quantifies performance of other methods as well. We find that the χ^2-method is the next best after our method, followed by the t-test and dGWT method. The VT method offered the lowest performance in our comparisons.

Methods were compared using the optimal settings for the respective parameters. The settings were computed based on the information about the actual number of steps in the simulations. Therefore the performance graph of other methods in Fig. 5.12 represents an upper bound on their actual performance. In contrast, our method operates in a parameter free manner, incorporating noise statistics, that which was estimated automatically).

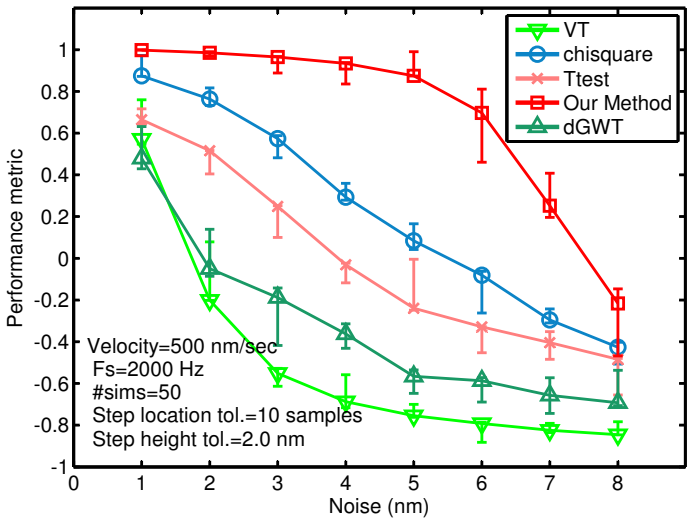


Figure 5.12: Comparison of ROC performance metric for different methods under various noise levels. Performance was computed by evaluating the average TPR (percentage of correctly found steps) and FPR (percentage of spurious steps) for 50 simulations of stochastic stepping consisting approximately 100 steps with white noise added to the stepping signal. The TPR and FPR was then fused into a single ROC performance number. Under this performance metric, our method is better than existing methods for all tested noise levels and the drop in performance with increasing noise is least for our method. Other parameters for the simulations are included in the plot. The error bars mark the maximum and minimum values of the performance number obtained over the 50 simulations. There is a sudden drop in performance of our method at around 6 nm noise variance for our method which is consistent with our analysis on limits of detection that predicts sudden drop in performance for noise SD of 6.5. However, it still performs better that other methods. The performance number of other methods plotted here is for optimized parameters using the knowledge of true signal, therefore represents an upper bound on their performance.

Performance measure computation

A step in the estimated signal is declared TP (true positive) if it is in the neighborhood of an actual step (defined as a window size of 10 samples on either side of the actual step) and the step height is within a tolerance (± 2 nm) of actual step height. If there are multiple such steps, the one closest to the actual step is selected to represent a true step and others are discarded. Also, the declared true step is not allowed to represent any other step in the actual signal. This way there is a unique step in the actual signal for each TP step. All the identified steps that are not true positives are declared false positives, FP. Likewise, actual steps that are not represented by any true positive step become unidentified steps or false negatives (FNs). These numbers can be expressed as a percentage of the total number of actual steps (N_0). Therefore we get the true positive rate, TPR=$100\frac{TP}{N_0}$, and the false positive rate, FPR=$100\frac{FP}{N_0}$. For a given method, these numbers vary with the amount of noise in the system, frequency of arrival of steps. Step

functions with staircase profile were generated with a Poisson distributed dwell times with a specified mean (500 nm/sec with sampling rate of 2kHz). Step size was also fixed (8 nm) but not known to any of the detection methods. For a given noise level, 50 simulations were performed and analyzed by various detection methods. The TPR and FPR was computed for each simulation. TPR and FPR is used to compute performance measure given by $\rho = \frac{1}{70.7}\sin\left(\tan^{-1}\left(\frac{\mathrm{TPR}}{\mathrm{FPR}}\right) - \frac{\pi}{4}\right)\sqrt{\mathrm{TPR}^2 + \mathrm{FPR}^2}$. It is essentially the distance of the point (FPR,TPR) from the line that is at 45° to the horizontal axis of TPR vs. FPR plot normalized by the maximum distance of 70.7. This is the distance of the point (0,100) from the 45° line. This line is also called line of no distinction because any point on this line represents equal probability of an identified step being true or spurious. Mean ρ is plotted in Fig. 5.12 with error bars indicating the sample standard deviation of 20 simulations.

5.4 Discussion

The advent of new instrumentation is driving a rapid expansion of single molecule studies. There is an urgent need for methods to better decipher single molecule data and resolve molecular events and behavior. This single molecule data frequently contains stepping characteristics, but noise and filter dynamics can obscure the steps. An automated method not only relieves the burden of manually picking out steps but also eliminates the bias introduced by manual detection. Advances in signal processing techniques have led to improved analytical capabilities that can be leveraged for a better treatment of experimental data from biological studies.

We have developed a novel step detection methodology that fits data while penalizing steps in an optimization framework with an iterative strategy for adjusting the penalty based on the histogram of step sizes. In the proposed method, a step fit is performed by optimizing a cost function composed of the χ^2 error and a penalty on total number of steps in the fit. The histogram of step sizes for the fit is used to reweight the penalty on the number of steps and the process is repeated in an iterative manner until step-size histograms do not change over iterations. In essence, it searches entire data set for a pattern of frequently occurring steps and then optimally place such steps in the appropriate locations. We have demonstrated that our methodology has several advantages over existing step detection methods: (1) It produces sharp step-size histograms when the underlying data has a unimodal step size distribution. This helps in quantifying the number of steps of a particular size. (2) It has the ability to compensate for probe dynamics if a suitable model of the dynamics is available. Probe dynamics distort step signals into smoothly varying signals and obscure the steps. Existing step detection methods fail to identify

the underlying steps, but our method corrects for the smoothing that results from probe dynamics and reveals the hidden step signal. (3) Unlike other methods, our algorithm also functions without specifying any parameters. Noise statistics is a required input, but can be automatically estimated from the experimental data. However, a model for sensor dynamics is required if compensation is desired. (4) In comparison with other methods, it has the highest accuracy in terms of detection of true steps and not producing false steps. (5) It can also detect events characterized by impulsive/spiking behavior. (6) The flexibility offered by having an optimization framework and an intuitive cost function can be utilized in a variety of ways. For example, estimation of parameters, compensation of nonlinearities and integrating existing knowledge about the system into the methodology.

We also developed a strategy to quantify the limitations of our step-detection method, as well as to compare the limitations of other step detection methods. An undesirable artifact of our method is that it will fit steps regardless of whether the underlying signal has steps or varies smoothly; most step detection methods suffer from this drawback. Nonetheless,we provide a quantitative means to judge the quality of the fits by computing the likelihood of the fit being produced from stepping data versus smoothly varying data. Another apparent limitation of the method is its computational complexity. The core of the algorithm can be implemented easily. We have implemented complexity reduction techniques that render fast execution of the algorithm. The code for the algorithm implemented on MATLAB can be downloaded from the website, http://nanodynamics.ece.umn.edu/. The code can further utilize the parallel processing capability of MATLAB with multiple-core computers if available.

We provide in-depth analysis of the effect of various properties of a stepping data, like dwell-times and noise levels, that affect the performance of step detection methods. The SNR and dwell-time trade-off analysis shows that higher dwell-times are required to detect steps when SNR reduces. An inverse square-law relation between dwell-time and SNR was observed. We further provide quantitative means to characterize these effects for validation and comparison. A new measure for comparison (ROC) was introduced that effectively quantifies the performance of a step detection algorithm based on its detection accuracy and ability to discard spurious steps. We analyzed the relation between SNR and stepping rate that needs to be satisfied to ensure that our step-fitting method can be relied upon. We have tested the algorithm for various scenarios of stepping data and demonstrated its effectiveness over other methods in a comprehensive manner. The method was also tested on experimental data obtained from optical tweezers and AFM studies.

With this step-detection method we have introduced a new means of determining steps in single molecule data, that outperforms existing methods. Our method yields higher true

positives, lower false positives, for shorter dwell-times and lower SNR. Our methodology also uniquely address the smoothing effect of probe-dynamics to increase step-detection performance and applicability. These capabilities should enable the exploration of single molecule behavior at higher temporal and spatial resolutions.

5.5 Further applications

5.5.1 Detection of protein unfolding events and parameter fitting

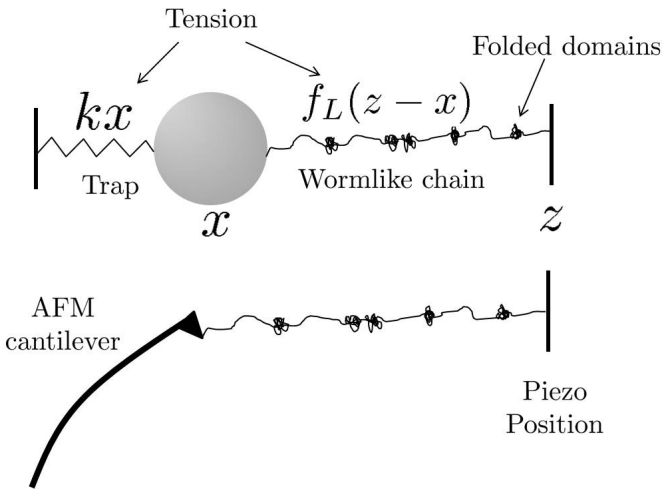


Figure 5.13: Physical model of experimental setup for protein unfolding using optical trap and AFM

As shown in Figure 5.13, the physical model for protein pulling/unfolding experiment is setup such that one end of the protein is fixed to a movable platform with its measured position z. The other end of the protein is connected to the optically trapped bead. The bead position is also measured and denoted by x . The optical trap is modeled as a linear spring of stiffness k. The entire system is in a viscous thermal bath which results in random forces on the bead and entropic elasticity of the protein. The entropic elasticity is well modeled as a worm-like chain given by

$$f_L(z-x) = \frac{K_b T}{L_p}\left[\frac{1}{4}\left(1-\frac{z-x}{L}\right)^{-2} + \frac{z-x}{L} + \frac{1}{4}\right]$$

where $f_L(z-x)$ is the tension in the molecule when it is extended by an amount of $z-x$. L is the contour length of the protein molecue which is not known apriori and is the quantity of interest here. L_p is the persistence length of of the molecule and is assumed to be known in this text. K_b is the Boltzmann constant and T is the absolute temperature of the bath.

In continuous time, the dynamics for the bead position is given by the following differential equation.

$$\gamma\dot{x} + kx = f_L(z - x) + \eta$$

where η is the Langevin forcing representing the effect of thermal bath on the bead position such that its power spectral density is given by $S^2(f) = 4K_bT\gamma$. In block diagram framework, this system appears as a feedback loop as shown in Figure 5.14. In this closed loop system, z is the external input and x is the measured output (assuming no measurement noise) and $G(s) = \dfrac{1}{\gamma s + k}$.

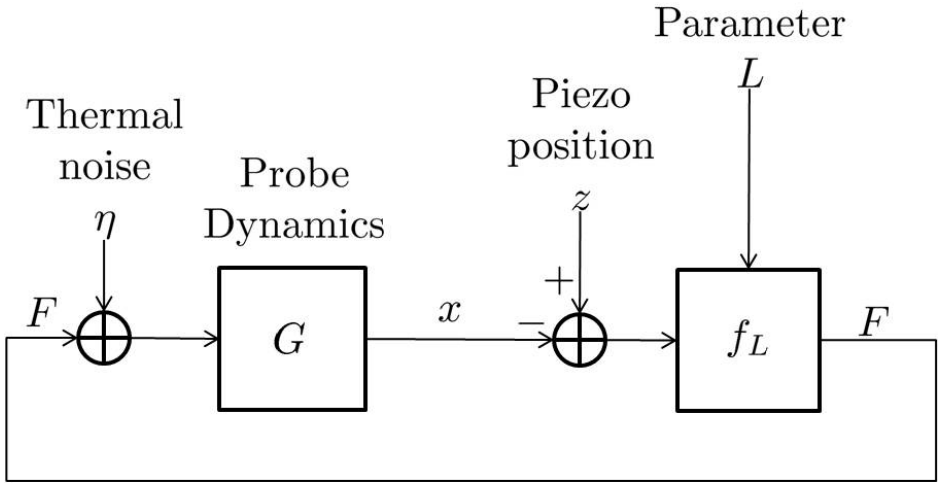


Figure 5.14: Continuous-time block diagram

To find the equilibrium or mean steady-state solution to the above equation, put $\dot{x} = 0$, $\eta = 0$. Therefore we have

$$kx = \frac{K_bT}{L_p}\left[\frac{1}{4}\left(1 - \frac{z-x}{L}\right)^{-2} + \frac{z-x}{L} + \frac{1}{4}\right]$$

$$\underbrace{\frac{4}{L^2}\left(\frac{L_pk}{K_bT} + \frac{1}{L}\right)}_{\alpha}x + \underbrace{\frac{4}{L^2}\left(\frac{1}{4} - \frac{z}{L}\right)}_{\beta} = \left(x + \underbrace{L - z}_{\theta}\right)^{-2}$$

$$(x+\theta)^2(\alpha x + \beta) - 1 = 0$$

$$\alpha x^3 + (\beta + 2\alpha\theta)x^2 + \left(\alpha\theta^2 + 2\theta\beta\right)x + \theta^2\beta - 1 = 0 \tag{5.16}$$

We see from the above equation that bead position satisfies cubic equation which has at least one real root which can be obtained numerically. The dynamics are nonlinear and

not easy to solve. However much insight can be obtained by doing a localized linearized analysis which is done in the next section.

Closed loop transfer functions In this section, closed loop transfer functions from external signals to internal signals will be obtained assuming linearized representation for f_L. Linearization of f_L is obtained around an equilibrium set of values that parametrize f_L which are z, x, and L. Let their equilibrium values be represented by $\bar{z}$, $\bar{x}$ and $\bar{L}$ respectively. These values will satisfy Eq. 5.16. Let $\bar{F}$ represent equilibrium tension in the molecule in this state. Let small deviations of the parameters be notated by $\tilde{z}$, $\tilde{x}$ and $\tilde{F}$. $\bar{L}$ is not assumed to change for this analysis. From linearization theory, we get following relations

$$\begin{aligned}
\tilde{F} &= \frac{\partial F}{\partial z}dz + \frac{\partial F}{\partial x}dx \\
&= \frac{\partial F}{\partial t}\frac{\partial t}{\partial z}dz + \frac{\partial F}{\partial t}\frac{\partial t}{\partial x}dx \;\; ; \;\; t = z - x \\
&= \left.\frac{\partial F}{\partial t}\right|_{(\bar{z},\bar{x},\bar{L})} (dz - dx) \\
\left.\frac{\partial F}{\partial t}\right|_{(\bar{z},\bar{x},\bar{L})} &= \frac{K_b T}{L_p \bar{L}}\left[\frac{1}{2}\left(1 - \frac{\bar{z}-\bar{x}}{\bar{L}}\right)^{-3} + 1\right] \\
&:= k_{wlc} \\
\tilde{F} &= k_{wlc}(\tilde{z} - \tilde{x})
\end{aligned} \tag{5.17}$$

Therefore we see that locally, worm-like chain model acts as a spring (positive for $\tilde{z}$ and negative for $\tilde{x}$). The stiffness of this spring increases cubically as $\bar{z} - \bar{x}$ approach $\bar{L}$. It is now easy to obtain the closed loop transfer functions as follows

$$
\begin{aligned}
TF_{\tilde{z}\to\tilde{x}} &= \frac{k_{wlc}G}{1+k_{wlc}G} \\
&= \frac{k_{wlc}}{\gamma s + k + k_{wlc}} \\
TF_{\eta\to\tilde{x}} &= \frac{G}{1+k_{wlc}G} \\
&= \frac{1}{\gamma s + k + k_{wlc}} \\
TF_{\tilde{z}\to\tilde{F}} &= \frac{k_{wlc}}{1+k_{wlc}G} \\
&= \frac{k_{wlc}\left(\gamma s + k\right)}{\gamma s + k + k_{wlc}} \\
TF_{\eta\to\tilde{F}} &= \frac{-k_{wlc}G}{1+k_{wlc}G} \\
&= \frac{-k_{wlc}}{\gamma s + k + k_{wlc}}
\end{aligned}
$$

Following useful observations can be made from the obtained transfer functions

- $TF_{\eta\to\tilde{F}} = k_{wlc}TF_{\eta\to\tilde{x}}$ or $TF_{\eta\to\tilde{F}} = \dfrac{k_{wlc}}{k}TF_{\eta\to k\tilde{x}}$. Therefore, fluctuations in force within the molecule increase and exceed fluctuations in the trapping force as k_{wlc} increase. The opposite is true for k. As expected, the stiffer spring shares the larger load of fluctuations.
- $\left\langle \tilde{F}^2 \right\rangle = \int_0^\infty \dfrac{k_{wlc}^2 \left\langle \eta^2 \right\rangle}{\gamma^2\omega^2 + (k+k_{wlc})^2} d\omega = \dfrac{4\pi K_b T k_{wlc}^2}{2\left(k+k_{wlc}\right)}$ which implies that there will be larger fluctuations in the molecular tension as it is stretched (larger k_{wlc}) . This is not true physically as the worm- like chain model does not work well for large force regimes.
- $k^2\left\langle \tilde{x}^2 \right\rangle = \int_0^\infty \dfrac{k^2 \left\langle \eta^2 \right\rangle}{\gamma^2\omega^2 + (k+k_{wlc})^2} d\omega = \dfrac{4\pi K_b T k^2}{2\left(k+k_{wlc}\right)}$. The opposite happens for force fluctuations in the trapping force or fluctuations in the bead position which decreases as the molecule is stretched and k_{wlc} is increased. This expression can be used to compute localized standard deviation of bead position fluctuations. Fluctuations decrease with increasing trapping force as well.

These observations are verified in a simulation result presented in Figure 5.15. Sudden jumps in the bead position marks the protein unfolding event characterized by sudden change in its contour length.

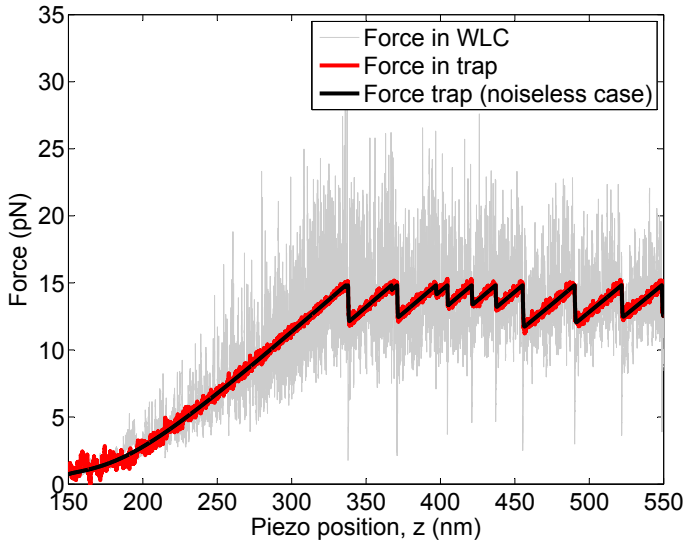


Figure 5.15: Continuous-time block diagram

Discrete time model Discrete time model of the entire system will be used in further text so that detection algorithms can conveniently be applied. In discrete time model, the probe dynamics, G is sampled into a discrete filter, H. All internal and external signals are sampled and indexed by sample number j. f_L is a static model therefore discretization has no effect on its behavior. However, the closed loop stability is not guaranteed in such a model. Since we know that the physical system is stable, to ensure stability in discrete time model, the sampling time has to be sufficiently small to capture all dynamics and allow the equations to reach a steady state solution. In particular noise has to be simulated as a band-limited white noise with cutoff much smaller than the Nyquist frequency. The discretization method used may not be an arbitrary choice to avoid solving cubic equation. zero-order-hold ensures that delayed values of the input signal to H are utilized during computation of the output. Using this method, the structure of H will be as in Eq. 5.18 . Figure 5.16 shows the block diagram for discrete case.

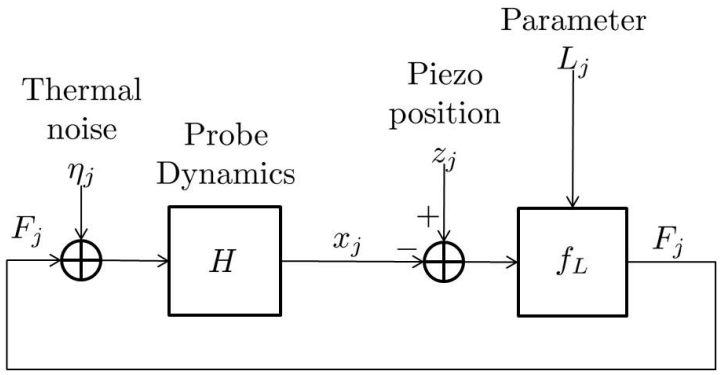


Figure 5.16: Discrete-time block diagram

The dynamical equations are described by the following equations

$$\begin{aligned} H &= \frac{b_1 D}{1 + a_1 D} \\ x_j &= H \times (F_j + \eta_j) \\ &= b_1 F_{j-1} + b_1 \eta_{j-1} - a_1 x_{j-1} \\ F_j &= f_{L_j}(z_j - x_j) \end{aligned} \tag{5.18}$$

where D denotes the delay operator. It is seen that simulating the discrete system is much simpler as there is no need to solve a cubic equation.

Detection of unfolding events Unfolding events are characterized by a change in the parameter L_j such that $L_{j+1} = L_j + u_j$. This change will alter the nonlinear map f_L. To estimate L_j from observations x_j that has been corrupted by noise, dynamic programming approach is employed where a sequence of possible candidates are chosen that minimizes a cost function. This is expressed in following equations

$$\hat{L} = \arg \min_{[u]_0^{N-1}} \sum_{j=0}^{N-1} (x_j - \hat{x}_j)^2 + W(u_j)$$

subject to

$$\begin{aligned} \hat{L}_j &= \hat{L}_{j-1} + u_{j-1} \\ \hat{x}_j &= H\hat{F}_j \\ \hat{F}_j &= f_{\hat{L}_j}(z_j - \hat{x}_j) \end{aligned}$$

A simulation result is shown in Figure 5.17.

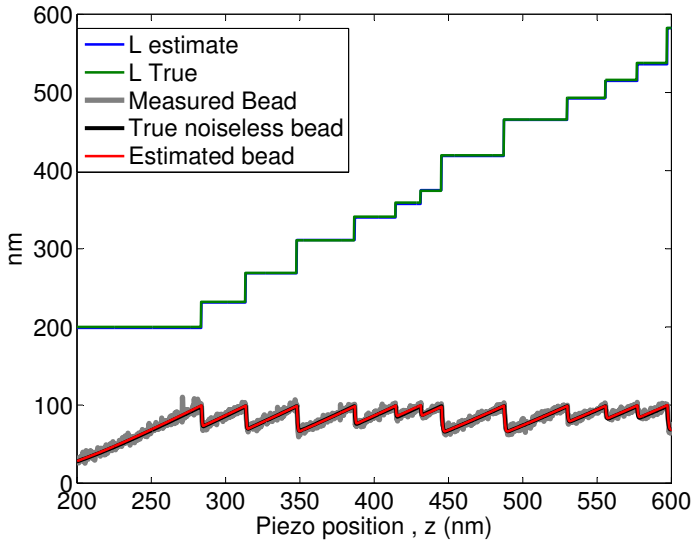


Figure 5.17: Fitting output of protein unfolding data to find change in the contour length of the molecule.

The above approach is computationally intensive especially when bounds on the possible L values is not known. The bounds on L can be estimated by inverting the WLC model for given force values and persistence length. Due to noise in the system, the obtained L values will look like a noisy step signal. The noise will be not be stationary; its variance will depend on the parameters and also the noise distribution will be very different from Gaussian distribution. Figure shows a result of inversion. Algorithm for inversion is summarized in Table 5.2.

Given z, F, P
Initialize ϵ (e.g. $\epsilon = 0.01$)
Initialize L (e.g. $L = z$)
Choose α (e.g. $\alpha = 0.1$)
WHILE $\lvert e \rvert > \epsilon$
$\quad e = F - WLC(z, P, L)$
$\quad L = L - \alpha e$
END

Table 5.2: Definitions: z is the measured extension of molecule (piezo position minus bead position). F is the measured force. P is the persistence length of the molecule. L is the contour length of the molecule and the output of this algorithm. ϵ is the desired error tolerance in estimated force for estimated L. WLC represents the model of worm-like chain. α is the rate of convergence. Very high α may make the convergence unstable and oscillatory. Very small α will slow down convergence.

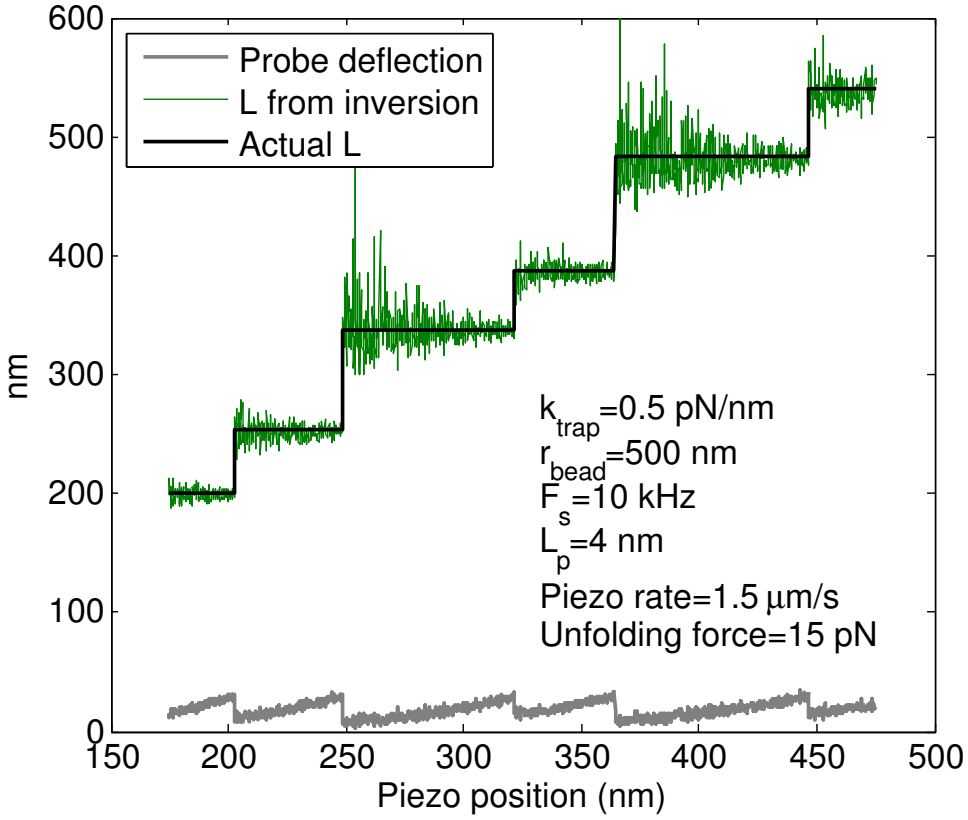


Figure 5.18

5.5.2 Simultaneous estimation of persistence length and contour length

In the previous section it was assumed that persistence length of the molecule is known. When it is not known then an approach, summarized in Table 5.3, can be used to estimate both the parameters. Figure shows a result of such an approach.

Given z, F
FOR $P \in P_0, ..., P_n$
  Estimate $L|_P$
  $e = \{F - WLC(z, P, L|_P)\}^2 + WN_{steps}$
END
$\hat{P} = \arg\min_P e$
$\hat{L} = L|_{\hat{P}}$
END

Table 5.3: Algorithm for estimating persistence length as well as contour length. For a range of P values, estimate L. Compute the force values using WLC model and compare this with the force data. Define an error, e as the χ^2 of data minus the fit and a penalty (W) on the total number of steps N_{steps} in the estimate, L. W is chosen as $9\sigma^2$ where σ is the average noise deviation in the data.

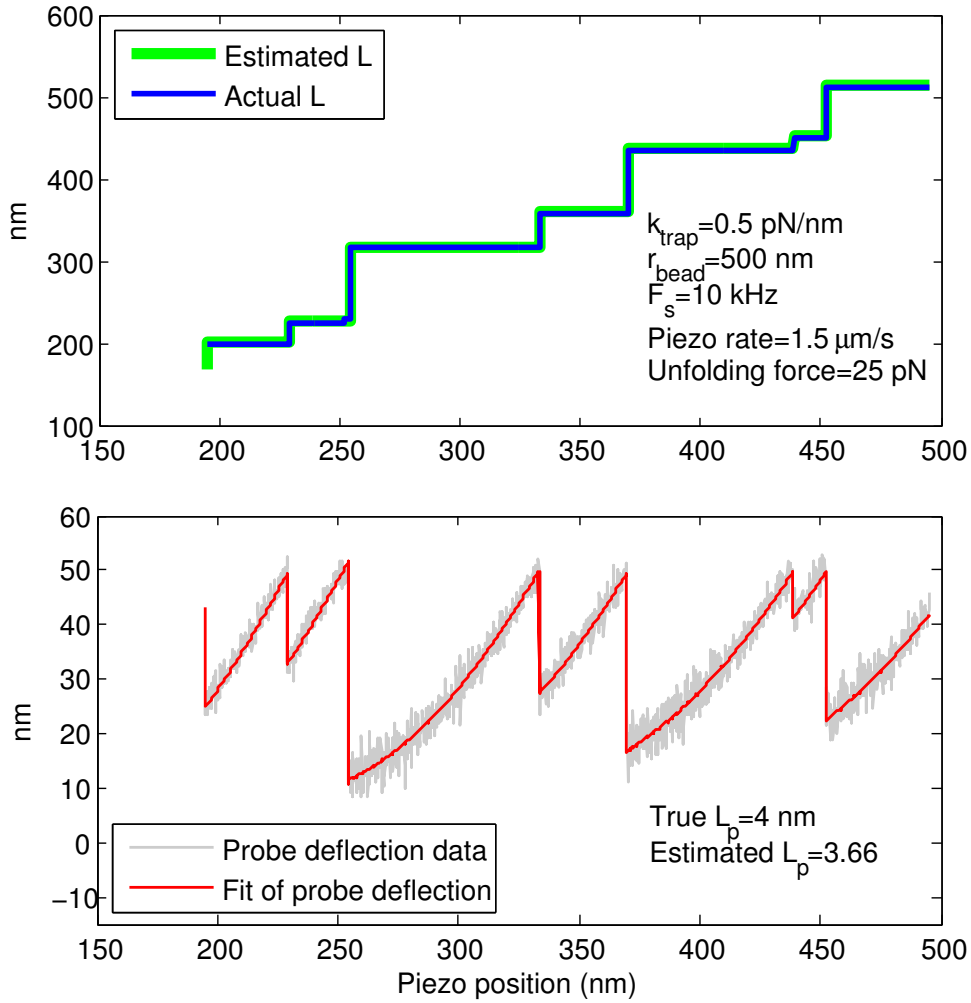


Figure 5.19: Estimation of persistence length, L_p and contour length, L changes. L_p is assumed to be constant but unknown. The simulation here is for $L_p = 4$ nm and the estimated values comes fairly close to $\hat{L}_p = 3.66$ nm. Estimation of L is also good.

5.5.3 $\mathbb{L}_0$ Optimization

Standard problem where steps are given a fixed penalty is described below. Depending on the penalty, the number of steps in the final estimate will be small or large.

$$x = \arg \min_{x_{k+1}=x_k+u_k} \sum_{k=1}^{N} (y_k - x_k)^2 + W(u_k)$$

In $\mathbb{L}_0$ framework, we would like to have at most m number of steps. This can be ensured by modifying the above cost function to the following.

$$
\begin{aligned}
x = \arg \quad & \min_{\substack{x_{k+1} = x_k + u_k \\ z_{k+1} = z_k + \delta(u_k)}} \quad \sum_{k=1}^{N} (y_k - x_k)^2 + W(z_k) \\
= \arg & \min_{\boldsymbol{X}_{k+1} = f(\boldsymbol{X}_k, u_k)} \sum_{k=1}^{N} g_k(\boldsymbol{X}_k, u_k)
\end{aligned}
$$

where,

$$
\boldsymbol{X}_k = \begin{bmatrix} x_k \\ z_k \end{bmatrix}
$$

$$
f = \begin{bmatrix} x_k + u_k \\ z_k + \delta(u_k) \end{bmatrix}
$$

$$
W(z_k) = \begin{cases} 0 & z_k \leq m \\ \infty & z_k > m \end{cases}
$$

z_k represents the number of steps in the estimate up to time k. To solve the above problem using dynamic programming, states need to be defined. States in this problem is obtained by stacking all possible combinations of x_k and z_k and therefore the number of possible states at each time is product of number of possible values of x_k and z_k. In order to apply dynamic programming to solve the above problem, these many states will be required. Note that this is independent of the functional form of W. Also note that many state transitions will have infinite cost to account for impossible state transitions. A Simulation result is shown in Fig. 5.20. It is seen that, as expected, the optimal solution contains m steps (maximum allowable) to minimize the variance between output and the fit.

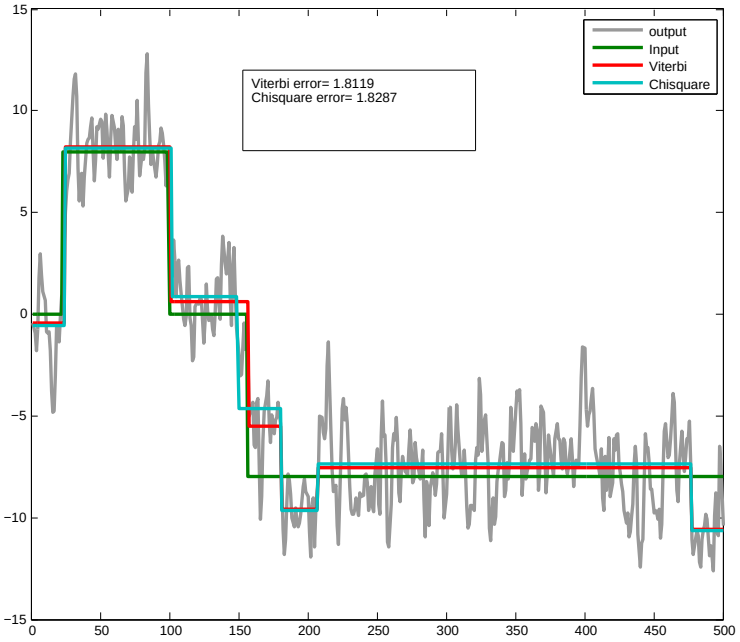


Figure 5.20: Random steps simulated in green. Noise is added to get output in gray. Three are 3 steps in the original signal. Constrained Viterbi estimation is performed with a maximum of 6 allowable steps. The optimal solution (in red) contains 6 steps. Chisquare estimation is also performed with 6 steps as the stopping criterion .

Chapter 6

Miscellaneous

6.0.4 Microfluidics

It is observed that trapped beads when actuated within fluid medium generate microflows due to the drag force. This ability can be utilized in generating microfluidic pumps that is of interest to lab-on-chip applications. Flow is generated by a set of 4 trapped beads located on a virtual Circe and equidistant apart on that circle, when rotated create a vortex like flow. The beads are trapped using single beam and multiplexed among four positions. These positions are changed dynamically to create a synchronized movement of the beads in circular fashion. Beads that are freely diffusing move along the flow and their fluorescence emission is recorded on a camera. Figure 6.1 shows an integrated image obtained by summing the values of video frames for several seconds. The effect of flow can be clearly seen. Interestingly, from the image it appears that the flow not only has tangential component but a radial component towards the flow.

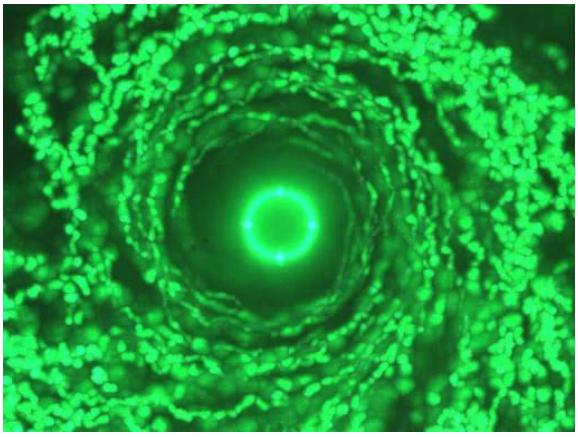


Figure 6.1: Flow generated by revolving microspheres

A preliminary quantitative study was done to measure the strength of the flow created

by the combination of such 4 revolving beads. Another independent trap measurable by photodiode was used in the flow and deflections from the nominal position indicated the strength of the flow. Figure 6.2 shows and exponential increment in the tangential flow strength (y-axis) as the separation between the sensing bead and the circle, along with the beads were moving, was reduced. The radial component (x-axis) is also registered. However, it seems it is an artifact rather than actual flow because the measured radial forces and profile do not change with angular velocity. It appears to be an artifact of the traps interfering with each other when brought closed together.

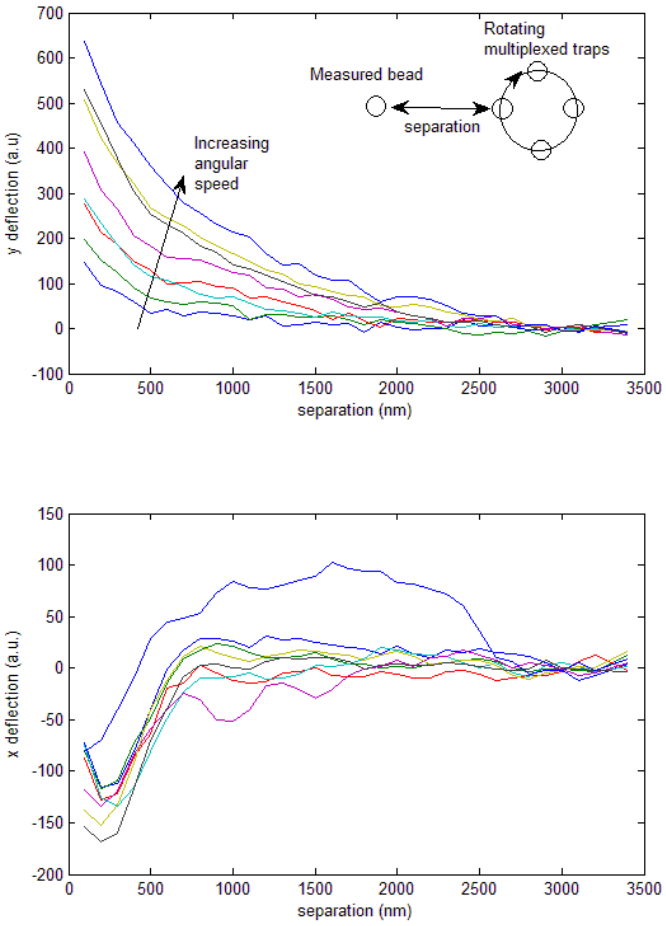


Figure 6.2: Flow velocity measurement

Chapter 7

Conclusion

Through this dissertation, I have demonstrated that it biophysics research can benefit from the tools of different disciplines like systems theory, communication and artificial intelligence. This work is a comprehensive coverage of research methodology starting with construction of optical tweezers construction followed by identifying and elimination of noise sources that gives the setup high resolution sensing capability. A flexible programming environment, LabVIEW is used in conjunction with FPGA hardware to get high bandwidth measurements. The instrumentation platform has been automated for most calibration tasks and experiments like DNA stretching and kinesin tracking with relative ease. The programming not only logs data but also provides realtime diagnostic display and ability to control experimental parameters. Benefits of such flexibility cannot be over-emphasized. During experiments, a need for larger range and elimination of nonlinearity in photodiode sensing led me to develop a neural network based sensing mechanism that increased the linear detection range by 100%. This enabled me to probe molecules like kinesin for a greater force regime without using potentially damaging high intensity laser. The ultimate goal to develop the optical tweezers is to understand biological systems up close. In this direction two popular model systems were chosen, both essential to life processes playing multiple significant roles within a cell. One is nucleic acid chain, DNA and the other is a motor protein, kinesin.

DNA is studied by optical tweezers by large number of researchers, involving stretching and breaking of bonds in DNA. One such experiment is characterizing the stiffness of DNA by its persistence length. While most analysis techniques are done offline by fitting models to the data after the experiment, I applied a recursive least squares based method that does the same fitting but capable of producing results in real-time. The main benefit in getting fast estimates is that changing parameters can be tracked and if any control action needs to be taken based on this parameter, real-time feedback become necessary. Looking at the

experimental setup for DNA stretching, that has two beads coupled to a single DNA, it occurred to some researchers that by measuring the position of both the beads, better noise cancellation is possible. It was demonstrated in [57] that this is indeed the case if one is looking at the DNA tether length. We analyzed this system from systems perspective and questioned if a general framework of network based sensing could be developed that gives higher signal to noise ratio. In the process, it was proven that the effect of thermal bath on a coupled beads system results in force fluctuations on the measured coordinates in the same way as a single bead system. Moreover, with respect to force as an estimated signal, signal to noise ratio always worsens due to the addition of thermal noise components from both beads. Furthermore, an optimal Wiener filter was constructed to see if noise reduction is possible. Due to the complexity of expressions, even for two bead system , the results were simulated rather than analytically produced. The results showed that in most cases SNR got worsened except in some cases where measurement noise was excessive compared to thermal noise.

Kinesin is an interesting model system for optical tweezers users. Optical tweezers provides a unique platform to track kinesin movement over microtubules with resolution and bandwidth unmatched by any other instrument yet. Still, the measured the data is noisy due to the thermal fluctuations of the bead. Typically, researchers compromise bandwidth to filter out noise and then fit staircase function to get an estimate of the step sizes within the data. Various heuristic approaches were used by the biophysics community to address this problem which is a niche area of the signal processing field. Therefore, we borrowed ideas from signal processing techniques to develop a step fitting algorithm, a modified Viterbi algorithm with added flexibility and generality. The result was a better fitting staircase signals with step size histograms giving more conclusive results in terms of well defined peaks. The algorithm has strong mathematical support along with heuristic intuition that goes with it. Experiments with kinesin protein was performed to validate the algorithm.

Bibliography

[1] A. Ashkin and JM Dziedzic. Optical levitation by radiation pressure. *Applied Physics Letters*, 19:283, 1971.

[2] A. Ashkin, JM Dziedzic, JE Bjorkholm, and S. Chu. Observation of a single-beam gradient force optical trap for dielectric particles. *Optics Letters*, 11(5):288–290, 1986.

[3] H. Sehgal, T. Aggarwal, and M. Salapaka. Characterization of dual beam optical tweezers system using a novel detection approach. In *American Control Conference, 2007. ACC'07*, pages 4234–4239. IEEE, 2007.

[4] A. Ranaweera and B. Bamieh. Calibration of the characteristic frequency of an optical tweezer using a recursive least-squares approach. In *American Control Conference, 2004. Proceedings of the 2004*, volume 2, pages 1836–1841. IEEE, 2004.

[5] K. Berg-Sørensen and H. Flyvbjerg. Power spectrum analysis for optical tweezers. *Review of Scientific Instruments*, 75(3):594–612, 2004.

[6] H. Sehgal, T. Aggarwal, and M.V. Salapaka. Systems approach to identification of feedback enhanced optical tweezers. In *Proceedings of SPIE*, volume 7038, page 703821, 2008.

[7] A. Pralle, M. Prummer, E.L. Florin, EHK Stelzer, and JKH Hörber. Three-dimensional high-resolution particle tracking for optical tweezers by forward scattered light. *Microscopy research and technique*, 44(5):378–386, 1999.

[8] K. Visscher, S.P. Gross, and S.M. Block. Construction of multiple-beam optical traps with nanometer-resolution position sensing. *IEEE journal of selected topics in quantum electronics*, 2(4):1066–1076, 1996.

[9] M.J. Lang, C.L. Asbury, J.W. Shaevitz, and S.M. Block. An automated two-dimensional optical force clamp for single molecule studies. *Biophysical journal*, 83(1):491–501, 2002.

[10] F. Gittes and C.F. Schmidt. Interference model for back-focal-plane displacement detection in optical tweezers. *Optics letters*, 23(1):7–9, 1998.

[11] W. Zi-qiang, LI Yin-mei, LOU Li-ren, WEI Heng-hua, and W. Zhong. Application of BP neural network to nonlinearity correction of optical tweezer force [J]. *Optics and Precision Engineering*, 1, 2008.

[12] S. Perrone, G. Volpe, and D. Petrov. 10-fold detection range increase in quadrant-photodiode position sensing for photonic force microscope. *Review of Scientific Instruments*, 79:106101, 2008.

[13] TT Perkins, S.R. Quake, DE Smith, and S. Chu. Relaxation of a single DNA molecule observed by optical microscopy. *Science*, 264(5160):822, 1994.

[14] TT Perkins, DE Smith, and S. Chu. Direct observation of tube-like motion of a single polymer chain. *Science*, 264(5160):819, 1994.

[15] TT Perkins, DE Smith, RG Larson, and S. Chu. Stretching of a single tethered polymer in a uniform flow. *Science*, 268(5207):83, 1995.

[16] S.B. Smith, Y. Cui, and C. Bustamante. Overstretching B-DNA: the elastic response of individual double-stranded and single-stranded DNA molecules. *Science*, 271(5250):795–799, 1996.

[17] M.C. Williams, J.R. Wenner, I. Rouzina, and V.A. Bloomfield. Entropy and Heat Capacity of DNA Melting from Temperature Dependence of Single Molecule Stretching. *Biophysical Journal*, 80(4):1932–1939, 2001.

[18] MD Wang, H. Yin, R. Landick, J. Gelles, and SM Block. Stretching DNA with optical tweezers. *Biophysical Journal*, 72(3):1335–1346, 1997.

[19] C.G. Baumann, S.B. Smith, V.A. Bloomfield, and C. Bustamante. Ionic effects on the elasticity of single DNA molecules. In *Proc National Acad Sciences*, volume 94, pages 6185–6190, 1997.

[20] C.G. Baumann, V.A. Bloomfield, S.B. Smith, C. Bustamante, M.D. Wang, and S.M. Block. Stretching of Single Collapsed DNA Molecules. *Biophysical Journal*, 78(4):1965–1978, 2000.

[21] U. Bockelmann, P. Thomen, B. Essevaz-Roulet, V. Viasnoff, and F. Heslot. Unzipping DNA with optical tweezers: high sequence sensitivity and force flips. *Biophysical journal*, 82(3):1537–1553, 2002.

[22] RM Zimmermann and EC Cox. DNA stretching on functionalized gold surfaces. *Nucleic Acids Res*, 22(3):492–497, 1994.

[23] D.N. Fuller, G.J. Gemmen, J.P. Rickgauer, A. Dupont, R. Millin, P. Recouvreux, and D.E. Smith. A general method for manipulating DNA sequences from any organism with optical tweezers. *Nucleic Acids Research*, 34(2):e15, 2006.

[24] S.M. Block, L.S.B. Goldstein, and B.J. Schnapp. Bead movement by single kinesin molecules studied with optical tweezers. *Nature*, 348(6299):348–352, 1990.

[25] K. Svoboda, C.F. Schmidt, B.J. Schnapp, and S.M. Block. Direct observation of kinesin stepping by optical trapping interferometry. *Nature*, 365(6448):721–727, 1993.

[26] C. Kural, H. Kim, S. Syed, G. Goshima, V.I. Gelfand, and P.R. Selvin. Kinesin and dynein move a peroxisome in vivo: a tug-of-war or coordinated movement? *Science*, 308(5727):1469, 2005.

[27] A. Kunwar, M. Vershinin, J. Xu, and S.P. Gross. Stepping, strain gating, and an unexpected force-velocity curve for multiple-motor-based transport. *Current Biology*, 18(16):1173–1183, 2008.

[28] C.W. Wolgemuth and S.X. Sun. Elasticity of α-helical coiled coils. *Physical review letters*, 97(24):248101, 2006.

[29] K. Svoboda and S.M. Block. Force and velocity measured for single kinesin molecules. *Cell*, 77(5):773–784, 1994.

[30] P.J. Atzberger and C.S. Peskin. A Brownian dynamics model of kinesin in three dimensions incorporating the force-extension profile of the coiled-coil cargo tether. *Bulletin of mathematical biology*, 68(1):131–160, 2006.

[31] D.S. Friedman and R.D. Vale. Single-molecule analysis of kinesin motility reveals regulation by the cargo-binding tail domain. *Nature Cell Biology*, 1:293–297, 1999.

[32] T.C. Elston and C.S. Peskin. The role of protein flexibility in molecular motor function: coupled diffusion in a tilted periodic potential. *SIAM Journal on Applied Mathematics*, 60(3):842–867, 2000.

[33] J.F. Marko and E.D. Siggia. Stretching dna. *Macromolecules*, 28(26):8759–8770, 1995.

[34] T. Sakamoto, I. Amitani, E. Yokota, and T. Ando. Direct observation of processive movement by individual myosin V molecules. *Biochemical and Biophysical Research Communications*, 272(2):586–590, 2000.

[35] N. Hirokawa, Y. Noda, Y. Tanaka, and S. Niwa. Kinesin superfamily motor proteins and intracellular transport. *Nature Reviews Molecular Cell Biology*, 10(10):682–696, 2009.

[36] J. Kubelka, J. Hofrichter, and W.A. Eaton. The protein folding 'speed limit'. *Current opinion in structural biology*, 14(1):76–88, 2004.

[37] E. Schneidman, B. Freedman, and I. Segev. Ion channel stochasticity may be critical in determining the reliability and precision of spike timing. *Neural Computation*, 10(7):1679–1703, 1998.

[38] B. Sakmann. Elementary steps in synaptic transmission revealed by currents through single ion channels. *Bioscience reports*, 12(4):237–262, 1992.

[39] C.M. Coppin, J.T. Finer, J.A. Spudich, and R.D. Vale. Detection of sub-8-nm movements of kinesin by high-resolution optical-trap microscopy. *Proceedings of the National Academy of Sciences of the United States of America*, 93(5):1913, 1996.

[40] M. Rief, R.S. Rock, A.D. Mehta, M.S. Mooseker, R.E. Cheney, and J.A. Spudich. Myosin-V stepping kinetics: a molecular model for processivity. *Proceedings of the National Academy of Sciences of the United States of America*, 97(17):9482, 2000.

[41] M. Vershinin, B.C. Carter, D.S. Razafsky, S.J. King, and S.P. Gross. Multiple-motor based transport and its regulation by Tau. *Proceedings of the National Academy of Sciences*, 104(1):87, 2007.

[42] A. Engel and D.J. Müller. Observing single biomolecules at work with the atomic force microscope. *Nature Structural & Molecular Biology*, 7(9):715–718, 2000.

[43] J.R. Moffitt, Y.R. Chemla, S.B. Smith, and C. Bustamante. Recent advances in optical tweezers. *Biochemistry*, 77(1):205, 2008.

[44] P.V. Cornish and T. Ha. A survey of single-molecule techniques in chemical biology. *ACS chemical biology*, 2(1):53–61, 2007.

[45] A.F. Oberhauser, P.K. Hansma, M. Carrion-Vazquez, and J.M. Fernandez. Stepwise unfolding of titin under force-clamp atomic force microscopy. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2):468, 2001.

[46] D.B. Rowe. *Multivariate Bayesian Statistics*. Chapman & Hall/CRC, 2002.

[47] H. Sehgal, T. Aggarwal, and M.V. Salapaka. High bandwidth force estimation for optical tweezers. *Applied Physics Letters*, 94(15):153114, 2009.

[48] S.M. Block. Kinesin motor mechanics: Binding, stepping, tracking, gating, and limping. *Biophysical journal*, 92(9):2986, 2007.

[49] M. Nishiyama, E. Muto, Y. Inoue, T. Yanagida, and H. Higuchi. Substeps within the 8-nm step of the ATPase cycle of single kinesin molecules. *Nature cell biology*, 3(4):425–428, 2001.

[50] W.O. Hancock and J. Howard. Processivity of the motor protein kinesin requires two heads. *The Journal of cell biology*, 140(6):1395, 1998.

[51] T Aggarwal and M Salapaka. Real-time Nonlinear Correction of Back-Focal-Plane Detection in Optical Tweezers. *Submitted for publication*, 2010.

[52] C. Hyeon, S. Klumpp, and J.N. Onuchic. Kinesin's backsteps under mechanical load. *Physical Chemistry Chemical Physics*, 11(24):4899–4910, 2009.

[53] J.M. Fernandez and H. Li. Force-clamp spectroscopy monitors the folding trajectory of a single protein. *Science*, 303(5664):1674, 2004.

[54] M. Hegner, P. Wagner, and G. Semenza. Ultralarge atomically flat template-stripped Au surfaces for scanning probe microscopy. *Surface Science*, 291(1-2):39–46, 1993.

[55] D. Materassi, P. Baschieri, B. Tiribilli, G. Zuccheri, and B. Samorì. An open source/real-time atomic force microscope architecture to perform customizable force spectroscopy experiments. *Review of Scientific Instruments*, 80:084301, 2009.

[56] J.P. Egan. *Signal detection theory and ROC-analysis*. Academic Pr, 1975.

[57] J.R. Moffitt, Y.R. Chemla, D. Izhaky, and C. Bustamante. Differential detection of dual traps improves the spatial resolution of optical tweezers. *Proceedings of the National Academy of Sciences*, 103(24):9006, 2006.