

University of Nevada, Reno

**Single Molecule Imaging Methods and Novel Computational
Motion Analysis used to Characterize the Transitions of
Single Molecules between Multiple States of Motion:
Determining the Biochemical Kinetics of Single Molecules in
In Vitro and In Vivo Systems**

A dissertation submitted in partial fulfillment of the
requirements for the degree of the Doctor of Philosophy in
Cellular Molecular Pharmacology and Physiology

by

Michael Sean Carter

Jonathan Baker / Dissertation Advisor

December, 2014

Copyright by Michael Sean Carter 2014

All Rights Reserved



THE GRADUATE SCHOOL

We recommend that the dissertation
prepared under our supervision by

MICHAEL SEAN CARTER

Entitled

**Single Molecule Imaging Methods And Novel Computational Motion Analysis Used
To Characterize The Transitions Of Single Molecules Between Multiple States Of
Motion: Determining The Biochemical Kinetics Of Single Molecules In In Vitro
And In Vivo Systems**

be accepted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

"Josh" Jonathan Baker, Advisor

James Kenyon, Committee Member

Cynthia Masktic, Committee Member

Normand LeBlanc, Committee Member

Christine Cremo, Graduate School Representative

David W. Zeh, Ph. D., Dean, Graduate School

December, 2014

Abstract

In recent years, the field of biophysics has placed an increasing emphasis on characterizing single molecule motion as a tool to understand the complex biochemistry of experimental systems such as molecular motors which convert chemical energy to mechanical force. The interpretation of these motions remains limited because single molecules rarely display a single type of motion and instead dynamically switch between many different states of motion as they change biochemical state and interact with their local environment. This dynamic switching complicates the analysis of single molecule motion because the classical motion equations assume homogenous behavior preventing their direct application to switching trajectories. In this dissertation we develop a novel non-averaging displacement analysis (NADA, Chapter 2) that graphically represents all of the motion within single molecule trajectories at once allowing the different states of motion to segregate into different populations where their mechanical properties and lifetimes can be measured. We then applied our NADA method to analyze the motion of individual regulated thin filaments to provide insights into the basic mechanisms of striated muscle regulation. (Chapter 3). In the final chapter we described the software developed to perform the analysis within the dissertation including extracting positional information from video images of fluorescently labeled molecules with intensity profile based positional refinement at high temporal resolution, curating the measured trajectories and performing the MSD and NADA methods on the measured trajectories. (Chapter 4).

Table of Contents

Chapter 1 – Introduction	1
General Background	1
Muscles	2
Structure of the Sarcomere	3
Thick Filaments and Myosin	6
Thin Filaments	10
Regulation of Contraction	12
Intracellular Transport.....	14
Single Molecule Experiments	17
<i>In Vitro</i> Motility	18
Single Particle Tracking/Single Molecule Motion Analysis	21
References	27
Chapter 2 - A Non-Averaged Displacement Analysis to Determine Rates of Transition Among Multiple States of Motion Applied to Two Systems.	32
Abstract.....	33
Introduction.....	35
Materials and Methods	38
Simulations.....	38
Transport of GLUT4-GFP Vesicles in 3T3-L1 Fibroblasts.....	40
Protein Purification.....	41
Buffers.....	41
<i>In Vitro</i> Single Molecule Binding Assay	42
Custom Centroid Tracking Algorithm	43
NADA Algorithm	44
Results	46
Kinetics of Single Molecule Switching between Multiple Dynamic Populations.....	46
Using NADA to Determine Kinetics of Switching Between Multiple Dynamic States in Simulated Trajectories.	48
Using NADA to Determine the Kinetics of Experimentally Measured Single Actin Fragments Binding to (k_{DT}) and Dissociating from (k_{TD}) skS1 on a Coverslip Surface.	51
Using NADA to Determine the Kinetics of Experimentally Measured Glut4-GFP Fusion Vesicle Transport within 3T3-L1 Cells.....	54
Conclusion.....	58

Figures	59
References	64
Chapter 3: Non-Averaging Displacement Analysis of Dynamics of Regulated Thin Filament Motility	67
Abstract.....	67
Introduction.....	69
Methods	72
Proteins.....	72
Speckled Actin	72
Buffers.....	73
<i>In Vitro</i> Motility Assay	74
Analysis of <i>In Vitro</i> Motility Movies to Obtain Trajectories:.....	74
Low speed motility analysis	74
High Speed motility analysis.....	75
Lifetime Analysis	76
Analysis of NADA Histograms to Obtain V_{max}	77
Generation of Simulated Trajectories	77
Results	77
Average Velocities of Partially-Activated RTF Motility.....	77
Non-Averaging Displacement Analysis	78
Myosin-induced Activation of RTF Motility in the Absence of Calcium.....	78
Lifetime Analysis of Myosin-induced Activation of RTF Motility in the Absence of Calcium	80
Calcium-Activation of Regulated Thin Filament Motility	81
Lifetime Analysis of Calcium-Activation of Regulated Thin Filament Motility	82
Calcium Activation of Regulated Thin Filament Motility at 400 Frames per Second	83
Lifetime Analysis of Calcium Activation of Regulated Thin Filament Motility from High Speed Camera Trajectories	84
Discussion	85
Figures	91
References	99
Chapter 4: Single Particle Tracking Software	101
Abstract.....	101
Introduction.....	102
Collection of High Quality Images for Single Particle Imaging.....	102

Computation of Images to Obtain SPT Trajectories	104
Interpretation of Trajectories	106
Analysis Software	107
Particle Tracker	108
Loading Images	110
Loading and Saving Program Data	111
Centroid Based Single Particle Tracking Implementation.	113
Performing Centroid-Based Particle Tracking Analysis for Use in Trajectory Viewer	117
Performing Super Resolution Measurements using Intensity Profile Refinement	123
Trajectory Viewer	129
Opening Trajectory Viewer and Setting Calibration.....	130
Loading Trajectory Files.....	131
Interacting with Data Plots	132
Using the “Move” Tool	133
Using the “Zoom” Tool.....	133
Selecting and Copying Trajectory Data from the Plot Controls.....	134
Delete Selected Data from the Plot Controls	135
Qualifying Trajectories.....	136
Performing Advance Calculations on Trajectories	137
Calculating Mean Squared Displacement	137
Generating NADA Displacement Files for use in NADA Histogram Fitter.....	139
NADA Histogram Fitter.....	142
Opening NADA Histogram Fitter.....	144
Loading NADA Displacement Files	144
Adjusting the Binning Settings of the NADA Histogram Image.....	145
Filtering the NADA Histogram Image to Increase the Contrast of the Populations within the NADA Histogram	147
Obtaining Lifetimes from the NADA Histogram	149
Measuring the Lifetime of the Directed State of Motion	149
Measuring the Lifetime of the Tethered State of Motion	151
Measuring the Lifetime of the Diffusing State of Motion.....	153
Visualizing the Distribution of Displacements at Each Window Width (Vertical Read Tab).....	156

Exporting Calibrated NADA Histogram Images for the Generation of Figures....	157
Closing the NADA Histogram Fitter and Saving Current Settings.....	158
Summary	158
Figures	160
References.....	185

List of Tables

CHAPTER 4

TABLE 1 – DESCRIPTION OF PARTICLE TRACKER'S CONTROLS	158
TABLE 2 – DESCRIPTION OF PARTICLE TRACKER'S MAIN MENU ITEMS	159
TABLE 3 – DESCRIPTION OF PARTICLE TRACKER'S IMAGE INTERACTION TOOL BAR.....	160
TABLE 4 – DESCRIPTION OF PARTICLE TRACKER DRAWING LAYER TOOL BAR.....	161
TABLE 5 – DESCRIPTION OF PARTICLE TRACKER'S SCREEN AND PLAYBACK CONTROL'S CONTROLS	162
TABLE 6 – PARTICLE TRACKER'S MAIN TASK TABS	168
TABLE 7 – DESCRIPTION OF PARTICLE TRACKER'S ENLARGED VIEW WINDOW'S CONTROLS	169
TABLE 8 – DESCRIPTION OF TRAJECTORY VIEWER'S MAIN WINDOW'S CONTROLS ..	170
TABLE 9 – SUMMARY OF MENU FUNCTIONS IN TRAJECTORY VIEWER.....	171
TABLE 10 – DESCRIPTION OF THE CONTROL CLUSTERS CONTROLS	173
TABLE 11 – DESCRIPTION OF NADA HISTOGRAM FITTERS CONTROLS	174
TABLE 12 – SUMMARY OF MENU FUNCTION IN NADA HISTOGRAM FITTER.....	175
TABLE 13 – SUMMARY OF THE NADA HISTOGRAMS CONTROL'S COMPONENTS	177
TABLE 14 – DESCRIPTION OF NADA HISTOGRAM CONTROL'S RIGHT CLICK MENU OPTIONS	177
TABLE 15 – SUMMARY OF THE TASK TABS IN NADA HISTOGRAM FITTER	180
TABLE 16 – DESCRIPTION OF NADA HISTOGRAM FITTER'S PLOTS CONTROL'S RIGHT CLICK MENU OPTIONS.....	181

List of Figures

CHAPTER 1

FIGURE 1 – STRUCTURE OF A MUSCLE FIBER.....	2
FIGURE 2 – SARCOMERE STRUCTURE.....	3
FIGURE 3 – CARTOON DESCRIBING THE SLIDING FILAMENT THEORY OF MUSCLE.....	4
FIGURE 4 – DIAGRAM OF A THICK FILAMENT FROM SKELETAL MUSCLE AND A SINGLE MYOSIN MOLECULE.....	5
FIGURE 5 – THE MECHANOCHEMICAL CROSS-BRIDGE CYCLE FOR MYOSIN	8
FIGURE 6 – CARTOON OF THIN FILAMENT	9
FIGURE 7 – AN ILLUSTRATION OF THE GEOMETRY AND COMPONENTS OF A TYPICAL IN VITRO MOTILITY ASSAY.....	17
FIGURE 8 – IMAGES DEMONSTRATING THE APPLICATION OF SINGLE PARTICLE TRACKING ON IN VITRO MOTILITY.....	21

CHAPTER 2

FIGURE 1 – KINETICS OF SINGLE MOLECULE SWITCHING BETWEEN MULTIPLE DYNAMIC POPULATIONS	56
FIGURE 2 – NADA APPLIED TO SIMULATED DIRECTED-TETHERED TRAJECTORIES	57
FIGURE 3 – DETERMINING THE LIFETIME OF THE DYNAMIC POPULATIONS SIMULATED IN FIGURE 2.	58
FIGURE 4 – USING NADA TO DETERMINE THE KINETICS OF EXPERIMENTALLY MEASURED SINGLE ACTIN FRAGMENTS BINDING TO (KDT) AND DISSOCIATING FROM (KTD) SKS1 ON A COVERSIP SURFACE.....	59
FIGURE 5 – USING NADA TO DETERMINE THE KINETICS OF EXPERIMENTALLY MEASURED GLUT4-GFP FUSION VESICLE TRANSPORT WITHIN 3T3-L1 CELLS.....	60

CHAPTER 3

FIGURE 1 – AVERAGE VELOCITIES OF PARTIALLY-ACTIVATED RTF MOTILITY.	88
FIGURE 2 – COMPARISON OF THE VARIABLE VELOCITY AND DYNAMIC SWITCHING MODELS FOR RTF VELOCITIES AT PARTIAL ACTIVATION ANALYZED BY OUR NON-AVERAGING DISPLACEMENT ANALYSIS.....	89
FIGURE 3 – MYOSIN-INDUCED ACTIVATION OF RTF MOTILITY IN THE ABSENCE OF CALCIUM	90
FIGURE 4 – LIFETIME ANALYSIS OF MYOSIN-INDUCED ACTIVATION OF RTF MOTILITY IN THE ABSENCE OF CALCIUM	91
FIGURE 5 – CALCIUM ACTIVATION OF REGULATED THIN FILAMENT MOTILITY	92
FIGURE 6 – LIFETIME ANALYSIS OF CALCIUM ACTIVATION OF REGULATED THIN FILAMENT MOTILITY	93
FIGURE 7 – CALCIUM ACTIVATION OF REGULATED THIN FILAMENT MOTILITY AT 400 FRAMES PER SECOND.....	94
FIGURE 8 – LIFETIME ANALYSIS OF CALCIUM ACTIVATION OF REGULATED THIN FILAMENT MOTILITY FROM HIGH SPEED CAMERA TRAJECTORIES.....	95

CHAPTER 4

FIGURE 1 – PARTICLE TRACKER'S MAIN WINDOW	157
FIGURE 2 – SCREEN AND PLAYBACK CONTROLS	162
FIGURE 3 – PARTICLE TRACKER'S ENLARGED VIEW WINDOW	168
FIGURE 4 – TRAJECTORY VIEWER'S MAIN WINDOW	169
FIGURE 5 – CONTROL CLUSTER FROM TRAJECTORY VIEWER	172
FIGURE 6 – NADA HISTOGRAM FITTER'S MAIN WINDOW	173
FIGURE 7 – THE NADA HISTOGRAM CONTROL	176

Chapter 1 – Introduction

General Background

Molecular motors are biological machines composed of proteins that utilize chemical energy, most often in the form of ATP, to produce force and motion within cells

¹. Many different types of molecular motors have been characterized such as polymerization-based motors, nucleic acid motors, and cytoskeletal motors.

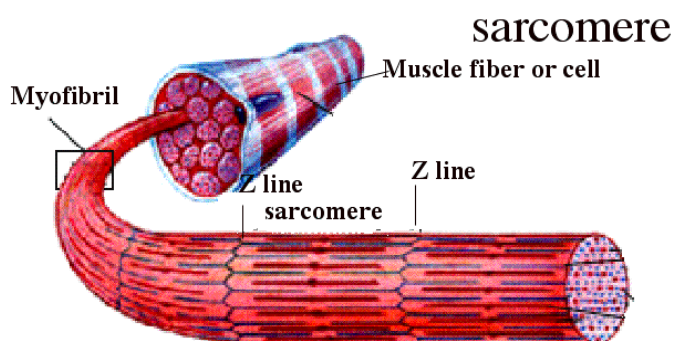
Polymerization-based motors such as actin and microtubules^{2,3} use the incremental assembly and disassembly of protein polymers to generate force. For example the polymerization and stabilization of actin is used by cells to create membrane projections such as filopodia ⁴. Nucleic acid motors, such as RNA and DNA polymerases, use the synthesis of DNA and RNA molecules to move large complexes of proteins along the template strand ⁵. Cytoskeletal motors such as myosin and kinesin utilize the binding of ATP to control their strong interactions with the cell's cytoskeletal elements to produce the large forces required to move large cargo such as vesicles and reshape the cell itself ¹.

Since the development of single molecule methods, the field of biophysics has placed an increasing emphasis on characterizing single molecular movements as a tool to understand the function of molecular motors. However, observing the motion of single molecules and interpreting the movement as the result of unique mechanical and biochemical states poses significant challenges. In this dissertation we develop novel analysis of this positional information from single molecular trajectories to extract the hidden biochemical states of the molecule using our non-averaging displacement

analysis (NADA, Chapter 2) and we will use our approach to analyze the motion of individual regulated thin filaments to provide insights into basic mechanisms of striated muscle regulation. (Chapter 3). In the final chapter we will describe our approach to extracting positional information of fluorescently labeled molecules with high spatial and temporal resolution (Chapter 4).

Muscles

The most studied molecular motor system is striated muscle. Muscles are highly structured excitable tissues that contract when stimulated. These tissues are used by animals to facilitate motions ranging from physical locomotion to the processes required to maintain homeostasis such as circulation, digestion and respiration ⁶.



7

Figure 1 – Structure of a Muscle Fiber.

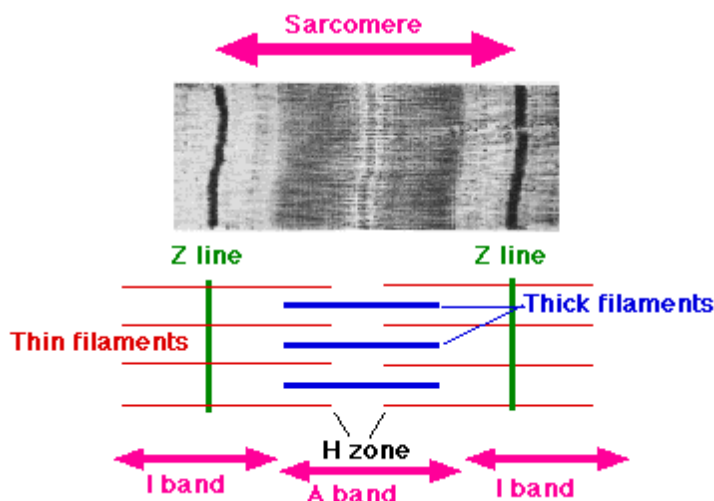
Diagram of the components of a muscle fiber from mammalian skeletal muscle.

The inherent structure of striated muscle is complex (Figure 1); however from a mechanical point of view, muscle tissue can be described as a collection of individual elongated cylindrical cells. These cells contain large numbers of repeating contractile units, referred to as sarcomeres, which are arranged in both series, to increase the speed of the muscles contraction, and parallel to increase the amount of force ⁸. While these contractile units vary in composition between muscle types, they share the same

basic elements and serve as the fundamental unit of contraction in striated muscle cells

6

Structure of the Sarcomere



9

Figure 2 – Sarcomere Structure

(Top) a transmission electron micrograph of sarcomere from an isolated human cardiac myocyte. Bottom a cartoon diagram outlining the basic structure found in the sarcomere including Z lines (green line), thin filaments (red lines), and the thick filaments (blue lines).

Each sarcomere is composed of overlapping filamentous proteins referred to as thick and thin filaments. The thick filaments are composed primarily of myosin and the thin filament is composed primarily of actin. The arrangement of these protein polymers, the interaction of the thick and thin filaments, and the presence of additional scaffolding creates a series of bands visible through methods such as interference contrast microscopy¹⁰. These bands have been used for more than fifty years to describe the structure of the sarcomere.

The edges of each sarcomere are defined by an optically dense region referred to as the Z-line. The Z-line connects many parallel sarcomeres together allowing for mechanical transduction of force generated by the sarcomere¹¹. Adjacent to the Z line

is the isotropic band (I-band) which is a band of low optical density composed primarily of thin filaments. Next to the I-band is the anisotropic band (A-band). The A-band is composed of three different regions. The first region, referred to as the C-zone, contains overlapping thick and thin filaments resulting in a dark appearance. The second region is a narrow light region referred to as the H zone. The H zone is the region of the thick filament which contains myosin heads but does not overlap with the thin filament. The third region of the A-band is the M-line. The M-line is the center of the bipolar thick filament and does not contain any myosin heads. This region in the center of the sarcomere may function in establishing symmetry ¹².

Sliding Filament Theory of Muscle Contraction

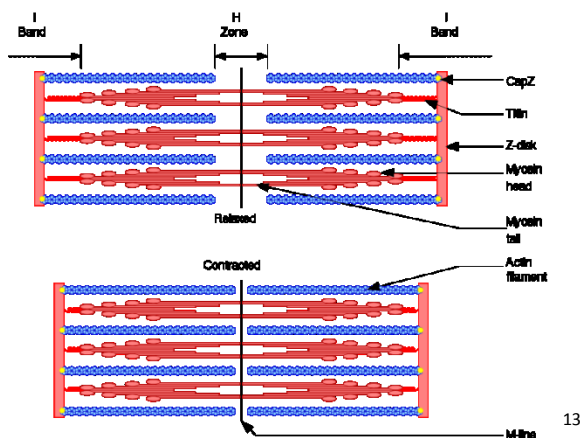


Figure 3 - Cartoon Describing the Sliding Filament Theory of Muscle

The top image contains a relaxed sarcomere before contraction and the bottom contains the contracted form of the same sarcomere. This figure demonstrates how sarcomeres can contract by increasing the amount of overlap between the thick and thin filaments, which decreases the length of the I-bands without changing the length of the filaments.

The currently accepted mechanism of sarcomere contraction was described independently by two different groups ^{10,14}, Andrew Huxley and Rolf Niedergerke and Hugh Huxley and Jean Hanson. Both studies were published in the same issue of Nature in May of 1954. These studies demonstrated that the A-band of the sarcomere

maintained a constant width during contraction. Both groups suggested a novel sliding mechanism to describe the observed shortening of sarcomeres during muscle contraction. This proposed mechanism, referred to as the “Sliding Filament Theory of Muscle Contraction”, holds that the thick and thin filaments which compose the sarcomere are of constant length and that during contraction these filaments slide past each other¹⁵. This sliding effectively reduces the size of the I-band, the region of the sarcomere without thick filaments, by increasing the overlap between the thick and thin filaments resulting in a net reduction of the total length of the sarcomere without shortening either the thick or thin filaments. This decrease in sarcomere length is then transmitted to the rest of the muscle. Since 1954 the molecular identity and function of the proteins within the thick and thin filaments have been resolved and their individual functions have been used to further develop the sliding filament model.

Thick Filaments and Myosin

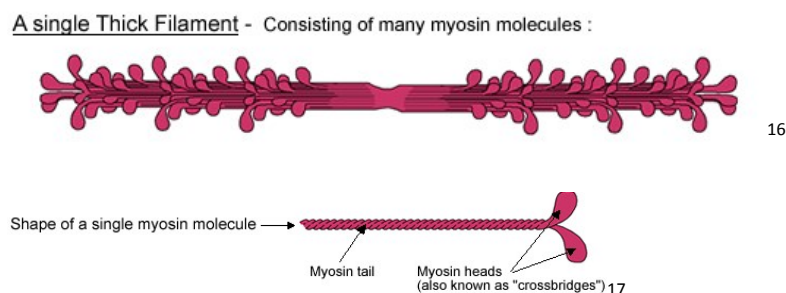


Figure 4 - Diagram of a Thick Filament from Skeletal Muscle and a Single Myosin Molecule

(Top) The body of the thick filament is composed a bipolar polymer of myosin II molecules that are bound together using the tail region of individual myosin molecules resulting in a thick filament with heads pointed in opposing directions joined by a bare region that contains no myosin heads. (Bottom) An individual myosin showing the myosin heads and tail region.

The skeletal muscle thick filament is a bipolar polymer of myosin II molecules. The body of the thick filament is composed of myosin tail domains, which interact in both an

anti-parallel manner, constituting the M-line, and parallel manner along the length of the myosin tail stabilizing the thick filament ^{18, 19}.

Each myosin molecule (Figure 4, bottom) in the thick filament is a protein hexamer composed of two identical heavy chains and two pairs of light chains ²⁰. Every heavy chain is composed of three functional domains: The globular head (N-terminal ~ 770 AA ²¹) containing the actin and ATP binding sites; the neck region that binds the regulatory and essential light chains and functions as the lever arm during force production ¹⁸; and the tail region (C-terminal) that forms an α -helical coil-coil interaction with another myosin heavy chain forming the myosin heavy chain dimer.

Every myosin heavy chain binds an essential (ELC) and a regulatory light chain (RLC). These light chains bind to the myosin heavy chain in the head-neck region. The specific function of the light chains is an ongoing topic of research ²² however the essential light chain may function to stabilize the myosin head domain and the phosphorylation state of the regulatory light chain (RLC) modifies the activity of the myosin molecule. The role of RLC phosphorylation is dependent on the specific myosin type. For example smooth muscle myosin is inactive until the RLC is phosphorylated yet the phosphorylation state of the RLC functions to tune the kinetics of thick filaments in cardiac cells leading to a cardiotoxic effect ²³.

Myosin like all molecular motors exhibits strong coupling between the chemical state of the motor as an enzyme and the motors mechanical state. In the case of myosin, the status of the ATPase reaction is reported as the presence or absence of bound reagents (ATP) or enzymatic products (ADP and P_i) within the nucleotide binding pocket while the mechanics can be considered as either an actin bound or unbound state. The

chemical state of the myosin head regulates myosin's affinity for actin thus controlling the binding and release of actin. Similarly myosin's ATPase reaction is catalyzed through the accelerated release of phosphate upon actin binding. Further, the binding of actin and release of phosphate is also associated with a mechanical rearrangement within myosin and movement of the lever arm referred to as a power stroke. This power stroke results in the production of force and a mechanical displacement of actin approximately 10 nm²⁴. Another smaller mechanical rearrangement (approximately 2 nm) exists following the release of ADP²⁵.

Because of the coupling of mechanical and chemical states, any kinetic scheme involving myosin must account for four distinct nucleotide states (Rigor (no nucleotide), ATP, ADP-P_i, and ADP) and 2 actin binding states (AM and M) per head, resulting in eight possible states per head or 64 possible states to account for the two myosin heads present in the myosin molecule¹. However it has been shown that in myosin II each head operate independently²⁶ allowing for the myosin kinetic scheme to be simplified to just eight states. Because of myosin's mechanochemical relationships many of the eight states are rarely populated allowing a further simplification of myosin's kinetics to a 4 state cycle (Figure 5) described by Lymn and Taylor in 1971²⁷.

Strongly Bound States

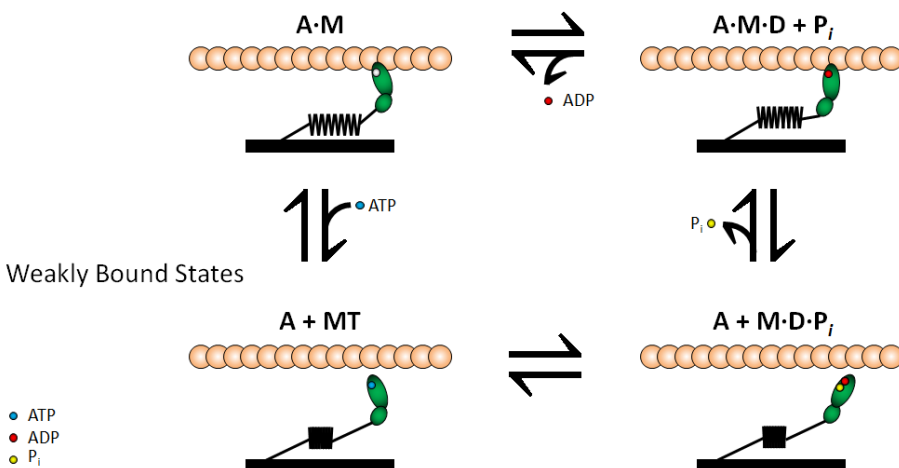


Figure 5 – The Mechanochemical Cross-bridge Cycle for Myosin.

The myosin cross-bridge cycle is shown essentially as described by Lymn & Taylor (1971)²⁷. The orange circles represent individual actin monomers, the myosin motor domain is in green, ATP (blue), ADP (red), and P_i (yellow). The diagram starts in the A·M state where myosin is strongly bound to actin with no nucleotide bound. In the presence of free ATP, myosin binds ATP releasing actin and transitioning to the MT state. ATP is hydrolyzed by myosin creating the MDP_i state. Actin binding accelerates phosphate release and ADP retained by myosin. Actin binding is associated with force production resulting in an 8 nm displacement. The final transition in the mechanochemical cycle is ADP release, which transitions myosin from the AMD state to the AM state. This transition is associated with additional force production and a 2 nm step²⁵.

The cycle starts with myosin strongly bound to actin in the absence of bound nucleotide. (AM state) Myosin binds ATP thus changing its affinity for actin, which results in myosin dissociating from the actin filament (MT State). While detached from actin, myosin hydrolyzes ATP into ADP-P_i and byproducts are retained within the myosin head. (MD-P_i State) The precise order of the next step remains unclear however it is known that myosin binds to actin with ADP still bound, phosphate is released and myosin's lever arm moves producing a power stroke (~8 – 10 nm; AMD state). Following the release of phosphate, ADP is released from myosin. This release is also thought to be associated with a further movement of the lever arm (AM State). Once

ADP is released, myosin is in the AM state and the cycle can be repeated allowing myosin to produce force and displace the thin filament ¹.

In addition to myosin molecules, thick filaments are also composed of many different accessory proteins ¹⁸. Titin extends from the center of the thick filament through the I-band to the Z-line and plays an important role as a molecular spring and as a mechanism for organizing the sarcomere. The thin filaments also include other important myosin binding proteins such as myosin-binding protein C. Mutations of this protein have been implicated in a number of familial hypertrophic cardio- myopathies. While myosin-binding protein C most likely contributes to the organization of myosin molecules in the thin filament the exact role it plays remains a topic of ongoing research

28 .

Thin Filaments

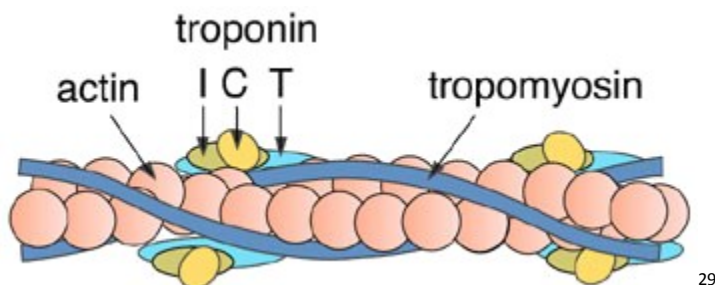


Figure 6 - Cartoon of Thin Filament

The thin filament is composed of an actin core (magenta) associated with the actin-binding regulatory complex TmTn. TmTn is composed of tropomyosin (blue) and troponin (TnI dark yellow, TnC light Yellow, and TnT cyan), which inhibits myosin strong binding in the absence of Ca^{2+} through a steric inhibition mechanism.

Thin filaments (Figure 6) are polymers composed of a core of filamentous actin (F-actin) and tropomyosin and troponin complexes that act to regulate the interaction of the thin filament with the thick filament. Actin, like most cytoskeletal elements, exists in two distinct forms: a globular monomer (G-actin) and a filamentous polymer of globular

subunits (F-actin). G-actin is an asymmetric globular protein that contains an ATP-binding cleft which regulates polymerization^{1,30}. In the presence of ATP and above a critical concentration of G monomers, actin spontaneously forms a long helical polymer referred to as F-actin³¹. F-actin has inter-monomer interactions, resulting in significant tensile strength while remaining flexible. Because of the polarized structure of the G monomers in F-actin, the filament has structural polarity. This polarity impacts not only the bulk properties of actin such as limiting the polymerization and depolymerization of actin to the barbed (+) end of the filament but also establishes the directionality to actin binding proteins such as myosin¹.

In addition to F-actin's bulk properties, the filament often interacts with a large number of accessory proteins, which are referred to collectively as actin binding proteins (ABPs). These proteins modulate the mechanical and biochemical properties of the actin filament and play an important role in both the structure and function of the sarcomere. The ABPs found in the sarcomere include proteins such as Cap-Z which attaches the thin filament to the Z-disk through its interaction with the actin's barbed (+) end, nebulin, which controls the length of the thin filament, and tropomodulin, which stabilizes the actin filament by binding the (-) end of the actin core. While the majority of the ABPs in the sarcomere function to maintain and support the sarcomere, the thin filament also contains the tropomyosin - troponin complex (TmTn), which functions to regulate contraction.

TmTn (Figure 6) is composed of 1 tropomyosin (Tm) and 1 troponin (Tn) per complex. Tm is a rod-shaped coiled-coil protein that loosely binds along the length of the actin filament spanning seven actin monomers. Each Tm interacts with adjacent Tm

molecules in a “head-to-tail” fashion forming a continuous strand running along the thin filament. Tn is a complex of three subunits: TnC, TnI and TnT. The TnC subunit contains two globular domains located at the N and C terminals of the protein. Both globular domains contain two Ca^{2+} binding sites. The C-terminal domain binds Ca^{2+} at physiological $[\text{Ca}^{+2}]$ while the N-terminal Ca^{+2} binding sites are only occupied during the period of high Ca^{+2} during muscle activation. TnC’s ability to reversibly bind Ca^{2+} allows TnC to function as the calcium sensor for the complex. In addition to TnC’s ability to bind Ca^{2+} , the subunit strongly binds both TnI and TnT, however the strength of TnI’s interaction is dependent on Ca^{+2} binding and is the most likely mode of action for communicating the Ca^{2+} occupancy of TnC to the TnI subunit³². TnI in coordination with Tm inhibits the myosin ATPase by blocking strong binding when Ca^{+2} is not bound to the TnC subunit. TnT functions as a structural element binding strongly to Tm and is most likely responsible for attaching the troponin complex to the thin filament³².

Regulation of Contraction

The contraction of striated muscles is regulated at the level of the thin filament. This regulation is mediated through the TmTn complex. When the filament is inactive such as in the absence of Ca^{2+} , the TmTn complex blocks myosin strong binding between the thick and thin filaments³³. The process of thin filament activation modifies TmTn’s interaction with the thin filament permitting myosin strong binding and myosin ATPase function. Currently there are several competing possible mechanisms for thin filament activation.

The first model is the simple three state model^{32,34}. The three state model proposes the thin filament exists in three distinct states and that the thin filament is activated

through a systematic progression through these states. The activation starts from the 'blocked' state which occurs in the absence of free Ca^{2+} . In the blocked state TnI binds strongly to actin, holding Tm in a blocked position preventing myosin strong binding. The binding of Ca^{2+} by TnC transitions the thin filament to the 'closed' state. In this state the interaction between Tn and Tm is comparatively weak. The decreased strength of the TmTn interaction allows Tm to have more freedom of "position" on the thin filament than in the blocked state. The increased Tm flexibility enables weak interactions between myosin and the thin filament but still prevents myosin strong binding. While the thin filament is in the closed position, myosin heads can on rare occasion strongly bind to the thin filament. Upon myosin binding, Tm is further displaced, locally transitioning the thin filament from the blocked state to the open state. This transition allows additional myosin heads to bind the thin filament opening up more of the filament to further myosin binding. The 'opening' process continues to feed forward until the maximal head cycling and ATPase activities are achieved at which point the system is fully active.

The second model of thin filament activation is the Hill two state model³⁵. In this model TmTn and seven actin monomers form a regulatory unit, which exist in two states: an inactive state with low myosin binding affinity or an active state with high myosin binding affinity. Calcium binding to TnC shifts the population of the two states to the high myosin affinity state. Since the activation of a single regulatory unit results in multiple myosin binding sites becoming available, the process of activation is highly cooperative. Further cooperativity is possible in this model through the head to tail

interactions of the Tm which promotes activation of adjacent regulatory groups along the thin filament ³⁶.

Another proposed model of activation is the dynamic two state scheme ³². In this mechanism Tm is in equilibrium between two stable states; the “blocked” state where Tm prevents myosin strong binding and the “open” state where myosin strong binding can occur freely. In this model the effect of Tn in the absence of Ca^{2+} is to hold Tm in the “blocked” position shifting the thin filaments activation equilibrium towards the blocked state. Similarly myosin heads bound to the thin filament displace Tm holding the thin filament in the open state resulting in a shift in the activation equilibrium favoring the open state. Tn binding Ca^{2+} functions to reduce the number of Tm held in the “blocked position” shifting the activation equilibrium from the “blocked state”. The magnitude of the effect from myosin heads bound to the thin filament on the activation equilibrium would be dependent on the lifetime of the acto-myosin strongly bound state. Our group has demonstrated that factors that modulate myosin strong binding kinetics play a role in modulating the activation of thin filaments and therefore muscle contraction ^{37,33}.

Intracellular Transport

In addition to muscle contraction, a major function of molecular motors in cells is the transport and relocation of cellular components such as proteins, vesicles, and organelles over long distances using cytoskeletal tracks such as microtubules and molecular motors such as kinesin.

The interior of a cell contains many interacting compartments that are physically separated. While this internal compartmentalization is clearly beneficial to the cell it

requires the existence of a system of directed and rapid trafficking of cellular components between compartments. In response to the demands of compartmentalization a complex system of regulation has evolved. This system of regulation controls transport by specifying cargos, their destination, and speed of delivery.

Neurons and other polarized cells are of particular interest in the study of intracellular transport because these cells represent an extreme case. Neurons are highly polarized cells and specialized in communication. They are composed of a small central body, which contains the nucleus and protein synthesis machinery, and long projections (axons and dendrites), which facilitate long distance communication between cells. These axons and dendrites pose a particular challenge for the cellular transport machinery because not only are they significantly long often millimeters in length, but they have very little ability if any to synthesize proteins or create organelles such as mitochondria. These limitations require that all proteins and other compounds present in the axons and synapses must be created in the soma and then transported long distances through these structures^{38,39,40}.

In axons there are at least two major classes of anterograde transport: The fast class and the slow class. The fast class is composed mainly of membrane bound components, which travel over 100mm/day ($1\mu\text{m/s}$). The fast class is primarily composed of vesicles transported in a continuous manner towards the synapse. The slow class is a diverse class of different cargos composed mainly of cytosolic proteins, which move at 1mm/day ($0.1\mu\text{m/s}$). The field currently has a good understanding of which cargos are transported by fast transport, which motors are responsible for this

transport and how they are regulated⁴¹. The molecular identity of the components of slow axonal transport and how that this mode of transport is regulated have proven to be more elusive.

In 2000 however two different groups independently reported that GFP-labeled neurofilaments in cultured cells were “transported rapidly but intermittently”^{42,43}. These studies reported that small polymers of neurofilaments were able to move with speeds in excess of 3 $\mu\text{m/s}$ in a manner similar to what has been classified as typical behavior for the fast axonal mechanism. However unlike the fast transport mechanism, which moves cargo smoothly from soma to synapse, the movement of neurofilaments seems to be interrupted by prolonged periods with no directed motion. The implication of this observation was that at any given time the majority of the neurofilaments in the axon were stopped. The resulting low duty ratio (i.e., time spent moving as a ratio of total time) resulted in an overall low average speed^{38,42}. Further it has been proposed that slow axonal transport is powered by the same type of motors that power vesicle transport⁴⁴. This “Stop and Go” motility has been proposed to be a fundamental mechanism behind the apparent large differences seen in axonal transport rates³⁸. This theory proposes that all cellular transport happens at similar instantaneous velocities and that differences in cargo delivery rates are created by varying the frequency and duration of the moving and stopping periods⁴³. Further by better understanding the mechanism underlying intermittent stopping found in intracellular transport we might be able to better understand the mechanism of the marked disruption of axonal transport found in many neurodegenerative diseases such as Alzheimer’s disease⁴⁵. In chapter 2 we demonstrate the utility of our non-averaging

displacement analysis (NADA) to measure the duration of directed and paused periods from single particle trajectories obtained from GLUT4-GFP vesicle transport in 3T3-L1 adipose precursor cells.

Single Molecule Experiments

Our understanding of molecular motor function has benefited significantly from the development of single molecule methods and technologies that allow researchers to characterize the properties of individual and small collections of interacting molecules. These single molecule methods obtain information regarding the molecular dynamics and function of individual molecules with details that are obscured by the averaging inherent to traditional ensemble methods. While significant insights are still possible using ensemble methods, the ability of single molecule methods to measure the behaviors of individual molecules provide many advantages over more traditional ensemble methods that characterize the average behavior of on the order of 10^{12} molecules at once.

The advantages of single molecule methods include the ability to directly measure the distribution of behaviors of individual molecules including the direct observation of rare events and states that are often averaged out in ensemble experiments. Further, these methods can simplify the measurement of complex and dynamic systems by removing the need to synchronize molecules in the system into a single state, which is often a requirement of ensemble kinetic measurements and often difficult to perform when studying molecular motors⁴⁶. Several methods have been developed to measure the mechanical behaviors (forces and stroke size) produced by single molecules. These approaches in which single molecules and small assemblies can be spatially manipulated have proven critical to understanding motor systems.

***In Vitro* Motility**

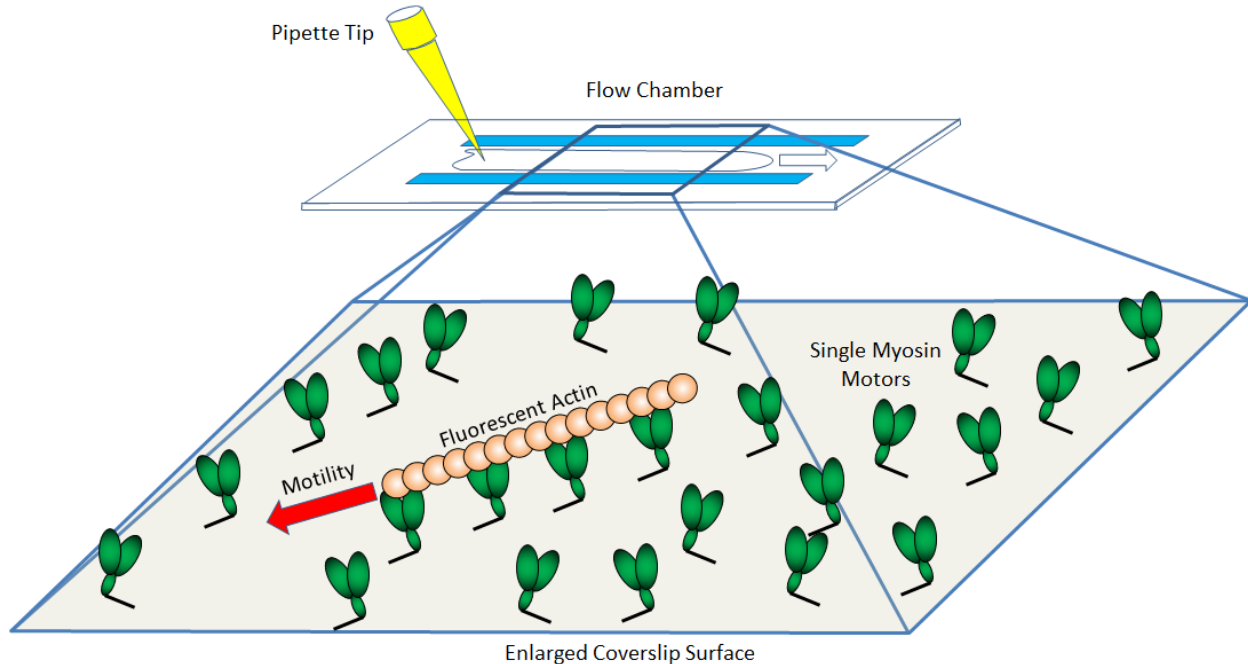


Figure 7 - An Illustration of the Geometry and Components of a Typical *In Vitro* Motility Assay.

The assay is performed on the surface of a cover slip attached to a glass slide. The cover slip is covered in nitrocellulose, which binds protein with high affinity. Myosin (Green) is flowed into the chamber followed by BSA (blocking agent, not shown). Fluorescently labeled F-actin is then flowed into the chamber and binds to the myosin heads. The chamber is flushed and the experimental buffer containing ATP is added. This allows myosin to cycle ATP and propel fluorescent actin across the cover slip. The movement of the actin filaments is recorded by video microscopy and processed through single particle tracking to obtain trajectories and velocities.

The development of the *in vitro* motility assay (Figure 7) was a major breakthrough in the study of myosin⁴⁷. It is a minimalistic experimental “movement” system that mimics the dynamics and kinetics of muscle contraction through the interaction of purified myosin molecules immobilized on a glass surface. These myosins propel purified F-actin filaments at speeds comparable to muscle contractions⁴⁸.

The current *in vitro* motility assay was the culmination of the work of several labs to create an *in vitro* biophysical assay for myosin movement as an analog of muscle contraction. The early methods utilized myosin-coated surfaces interacting with

phagocytic vesicles from Dictyostelium lysates, which are naturally coated in actin filaments emanating from the membrane surfaces⁴⁹ or using actin-coated beads⁵⁰. Ideally these actin-coated vesicles and beads could interact with myosin-coated coverslips to produce motility. These systems however were unable to generate the continuous directional movement seen in modern motility assays. This lack of motility was most likely the result of the experimental inability to create directionality in the actin filaments attached to the beads⁴⁹.

The first viable motility system with directionality was published by Sheetz and Spudich in 1983⁵¹. This method overcame the actin directionality problem by combining myosin-coated myosin filaments attached to polystyrene beads with skinned “giant intermodal” cells of *nitella axillaris*, giant-celled alga, attached to a cover slip. These cells exhibit strong directional cytoplasmic streaming along polar cables made of actin filaments on the cytoplasmic face of the organized chloroplasts. The highly polar nature of the actin cables conferred directionality to the motility system that was missing in the earlier methods.

The *Nitella*-based method was able to generate reproducible and quantitative velocity data, which led to several important insights regarding the function of the myosin system, such as establishing that only a single head domain of a myosin-II motor was required for motility⁵². However, it had several limitations. First, because these assays were dependent on skinned cells, researchers had no control and little knowledge of the biochemistry and composition of the actin cables being used and there was always a risk of contamination by residual components. Second, these assays used relatively large beads (~1 μm in diameter) to hold myosin. Not only did

these beads create significant viscous drag, compared to a 10 nm long actin filament, the beads moved relatively short distances when motile and often bound to actin without moving. The beads also complicated the analysis of movement because the bead did not necessarily report the position of the proteins of interest. Finally, the 3D structure of the actin cables significantly complicated the analysis of the moving beads by introducing a complex 3D surface for the bead to move over. While these cells were primarily flat, the 3D structure of the cables combined with the implied flatness of the microscope field complicates the calculation of velocities through an under estimation of the motion perpendicular to the 2D plane of the microscope ⁴⁸.

The modern form of the *in vitro* motility assay (Figure 7) was developed by Kron and Spudich in 1986 ⁴⁸. The assay consists of using glass surfaces coated with purified myosin to propel individual fluorescent F-actin filaments. The actin is polymerized from purified G-actin and labeled/stabilized by a fluorescent phalloidin-rhodamine conjugate method developed by Yanagida et al ⁵³. The major advance of this approach is the simplification it provides over the previous *Nitella*-based method. The use of purified proteins eliminates all of the complications mentioned above and reduces the number of components required to produce motility to only purified actin and myosin, ATP, and fluorescent phalloidin allowing the experimenter precise control of the motility assay composition.

Single Particle Tracking/Single Molecule Motion Analysis

Single molecules or small particles in biological systems often switch between different dynamic behaviors such as the transition from free diffusion to tethered diffusion associated with ligand binding ⁵⁴ or the stop-and-go motility of molecular

motors³⁸. Changes in particle dynamics are indicative of changes in physical states. To characterize the motion of molecules in single molecule assays, single particle tracking methods are used to measure the position of fluorescent molecules/particles in high spatial and temporal resolution. Single Particle Tracking (SPT) is a method that uses computer vision algorithms to measure the position of individual particles labeled with fluorescent markers in video microscopy. These observed positions are connected to form a trajectory that represents the motion of the particle overtime. SPT methods have provided tremendous insights into the mechanisms by which molecules bind, localize, and move both *in vivo* and *in vitro*.

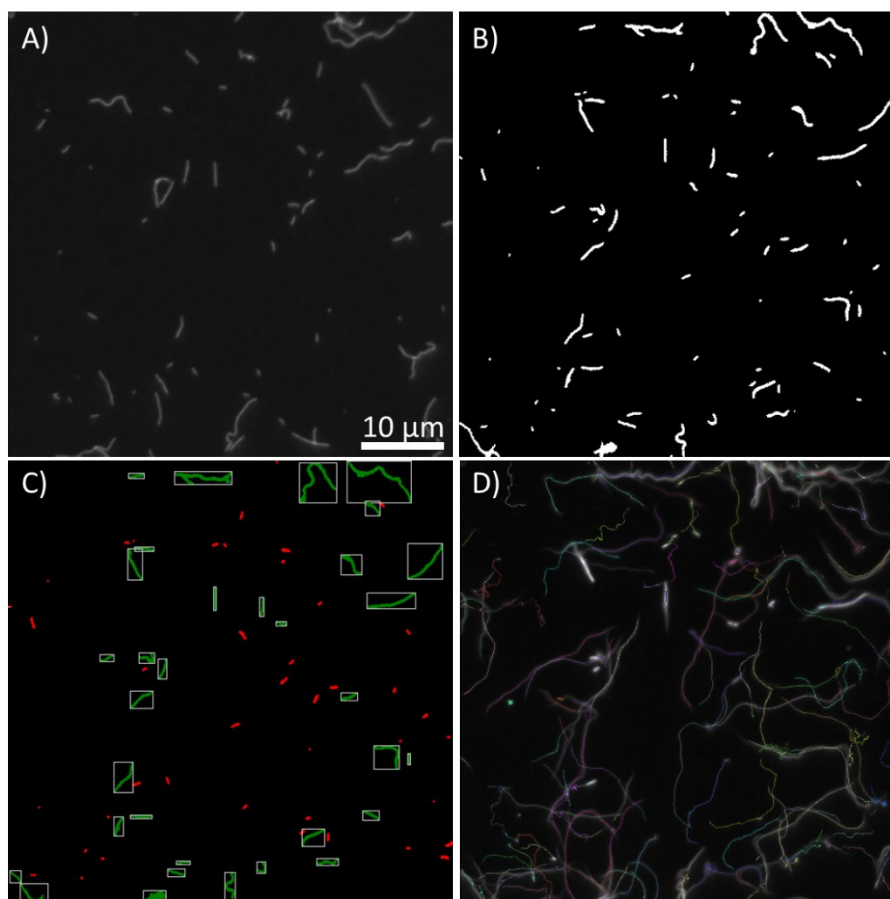


Figure 8 - Images Demonstrating the Application of Single Particle Tracking on *In Vitro* Motility

(A) Raw frame *in vitro motility* assay with tritc-phalloidin-labeled actin filaments (visible) sliding over a surface of skeletal myosin (invisible) in the presence of ATP. (B) Thresholded version of the image in A with white pixels representing the high intensity portions of the image and black pixels representing areas of low intensity. (C) Regions of intensity from (B) qualified based on apparent particle area with large objects colored green and small objects colored red. (D) Sum of intensity from 200 frames of *in vitro* motility, qualitatively illustrating motion of fluorescent actin, with the quantitative trajectories from the tracked single particles plotted as colored lines.

The process of determining the position of a single particle can be described as five steps: fluorescence labeling, segmentation, qualification, localization, and linking.

Labeling. There are many methods used to label individual molecules but the most common is extrinsic fluorescence labeling, which uses specialized small molecules that can be excited over a narrow band of wavelengths causing the molecule to emit light at

a longer wavelength. This Stokes shift between the excitation and emission light allows the excitation light to be optically blocked permitting only the emitted light to be observed, resulting in a high contrast image where only the light from the fluorescent molecules is visible. This fluorescence intensity indicates the location of the molecules of interest.

Segmentation. Once the particles are labeled within the image they must be isolated from the background through a process referred to as segmentation. The most direct method of segmentation is thresholding, a process which converts grey scale images to a binary image using a cutoff value. Pixels above the cutoff value are given a value of one (white) and pixels with intensity values less than the cutoff value are given a value of zero. When thresholding is used effectively the original image is reduced to a black background with clusters of adjacent pixels with the value of one. These groupings of adjacent high intensity pixels represent regions in the image that may contain labeled molecules. These clusters of high intensity pixels are grouped into possible objects.

Qualification. The collection of possible objects is then filtered by apparent area to eliminate sources of intensity other than the one of interest such as shot noise or large aggregations. This process of qualification determines if the region of intensity will be considered for further processing or ignored.

Localization. Following object qualification, the position of each object is determined in sub-pixel resolution by using either centroid-based methods or using an intensity profile analysis. In centroid-based methods the center of an object is determined by averaging the x and y position of each pixel belonging to that object. Centroid-based methods are computationally light and robust to noise but do not offer the level of

positional accuracy possible through intensity profile analysis-based methods. Intensity profile methods can localize the position of isolated fluorescent sources with higher accuracy than the Rayleigh limit. The improved accuracy is possible by fitting the distribution of intensities from the fluorescent object to a theoretical model of intensity distribution produced by a single sub-resolved source. Using numerical optimization such as non-linear curve fitting it is possible to estimate the best parameters for the model and determine the center position of the intensity distribution with higher accuracy than is possible using other methods. Conceptually the precision of these methods is limited by the signal-to-noise ratio in the intensity distribution, which increases with increasing photon counts ⁵⁵.

Linking. The final step in constructing single molecule trajectories is connecting the individual objects between frames. While this seems a simple task, the number of possible linkages grows geometrically with the number of objects that must be linked. The problem is further complicated by collisions of multiple objects and the appearance and disappearance of objects. The most common method of linking objects between frames calculates linkages one object at a time. This approach has the advantage of being computationally light while producing a minimum number of bad linkages. Recent revisions to linking algorithms use scoring matrices to consider better possible linkages and further reduce artifacts created by incorrect linking methods ⁵⁶.

Once the positions of the molecules of interest have been located using SPT, the traditional approach is to classify the trajectory into a distinct type of motion to determine the optimum equation to analyze the trajectory. The method of motion analysis is often arbitrarily chosen based on the quality of motion within the trajectory.

When a molecule's motion appears to be primarily diffusive, the mean-square displacement (MSD) of each trajectory is calculated at each window width (t) and the resulting mean squared displacements are fit to the diffusion equation ($MSD(t) = 4Dt^\alpha$) to obtain a single diffusion coefficient (D) and an α constant. When a molecule's motion appears to be primarily directed, trajectories are analyzed to obtain an average velocity V .

However as described in chapter 2, SPT data shows that single molecule trajectories are complex, consisting of different states of motion such as diffusive movement, directed transport⁵⁷ and periods of static binding or caging⁵⁸. In a MSD analysis, periods of short binding decrease the apparent diffusion coefficient, D , and the α -value. In measuring the speed of directed movement, short binding events result in an apparent average velocity that is less than the maximal velocity of movement. The assumption of homogeneous behavior by the standard motion analyses methods not only leads to a poor estimation of the parameters of motion but can also convolute important mechanistic details.

Previously Change-point analysis⁵⁹ and Hidden Markov models⁶⁰ have been used to extract multiple populations from single molecule trajectories. Both approaches attempt to identify specific points in a trajectory at which state transitions occur. However both methods are black-box approaches that are difficult to implement and require *a priori* definitions of states. In chapter 2, we develop and test a Non-Averaged Displacement Analysis (NADA) as an alternative lit-box approach that allows one to visualize trajectories in a single graphic from which multiple populations can be directly identified and characterized and their lifetimes determined. This technique is relatively

easy to implement and can be applied to a wide range of different systems which contain multiple dynamic behaviors.

References

1. Howard, J. *Mechanics of Motor Proteins & the Cytoskeleton. Mech. Mot. Proteins Cytoskelet.* (Sinauer Associates, 2001).
2. Mogilner, A. & Oster, G. Polymer Motors: Pushing out the Front and Pulling up the Back. *Curr. Biol.* **13**, R721–R733 (2003).
3. Theriot, J. a. The Polymerization Motor. *Traffic* **1**, 19–28 (2000).
4. Mattila, P. K. & Lappalainen, P. Filopodia: molecular architecture and cellular functions. *Nat. Rev. Mol. Cell Biol.* **9**, 446–54 (2008).
5. Seidel, R. & Dekker, C. Single-molecule studies of nucleic acid motors. *Curr. Opin. Struct. Biol.* **17**, 80–6 (2007).
6. Alberts, B. *et al. Molecular Biology of the Cell.* (Garland Science, 2008).
7. Muscle fiber image. at <http://www.ucl.ac.uk/~sjjgsca/muscleSlidingFilament.html>
8. Carabello, B. A. Concentric versus eccentric remodeling. *J. Card. Fail.* **8**, S258–63 (2002).
9. Wikipedia Image - Sacromere. at <http://en.wikipedia.org/wiki/Sarcomere>
10. Huxley, A. F. & Niedergerke, R. Structural Changes in Muscle During Contraction: Interference Microscopy of Living Muscle Fibres. *Nature* **173**, 971–973 (1954).
11. Luther, P. K. The vertebrate muscle Z-disc: sarcomere anchor for structure and signalling. *J. Muscle Res. Cell Motil.* **30**, 171–85 (2009).
12. Katzemich, A. *et al.* The function of the M-line protein obscurin in controlling the symmetry of the sarcomere in the flight muscle of *Drosophila*. *J. Cell Sci.* **125**, 4170–4170 (2012).
13. Sliding filament image. at <http://en.wikipedia.org/wiki/Sarcomere>
14. Huxley, H. & Hanson, J. Changes in the Cross-Striations of Muscle during Contraction and Stretch and their Structural Interpretation. *Nature* **173**, 973–976 (1954).
15. Clarke, M. Muscle: the sliding filament at 50. *Nature* **429**, 145 (2004).
16. Cartoon of a thick filament. at <http://www.ivy-rose.co.uk/Define/Myosin>

17. Image of a single myosin. at <<http://www.ivy-rose.co.uk/Define/Myosin>>
18. Craig, R. & Woodhead, J. L. Structure and function of myosin filaments. *Curr. Opin. Struct. Biol.* **16**, 204–12 (2006).
19. Xu, S. *et al.* The M.ADP.Pi state is required for helical order in the thick filaments of skeletal muscle. *Biophys. J.* **77**, 2665–76 (1999).
20. Lodish, H., Berk, A. & Zipursky, S. in *Mol. Cell Biol. 4th Ed.* (W. H. Freeman, 2000). at <<http://www.ncbi.nlm.nih.gov/books/NBK21724/>>
21. Cooke, R. The sliding filament model: 1972-2004. *J. Gen. Physiol.* **123**, 643–56 (2004).
22. Muthu, P. *et al.* Structural and functional aspects of the myosin essential light chain in cardiac muscle contraction. *FASEB J.* **25**, 4394–405 (2011).
23. Toepfer, C. *et al.* Myosin regulatory light chain (RLC) phosphorylation change as a modulator of cardiac muscle contraction in disease. *J. Biol. Chem.* **288**, 13446–54 (2013).
24. Mehta, A. D. Single-Molecule Biomechanics with Optical Methods. *Science* (80-). **283**, 1689–1695 (1999).
25. Nyitrai, M. & Geeves, M. a. Adenosine diphosphate and strain sensitivity in myosin motors. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.* **359**, 1867–77 (2004).
26. Taylor, E. W. & Trentham, D. R. Mechanism of Actomyosin Atpase and the Problem of Muscle Contractio. *Crit. Rev. Biochem. Mol. Biol.* **6**, 103–164 (1979).
27. Lymn, R. W. & Taylor, E. W. Mechanism of adenosine triphosphate hydrolysis by actomyosin. *Biochemistry* **10**, 4617–24 (1971).
28. Flashman, E., Redwood, C., Moolman-Smook, J. & Watkins, H. Cardiac myosin binding protein C: its role in physiology and disease. *Circ. Res.* **94**, 1279–89 (2004).
29. Structure of the muscle thin filament. at <http://jolisfukyu.tokai-sc.jaea.go.jp/fukyu/tayu/ACT04E/img/04IMG/04_16.jpg>
30. Kabsch, W., Mannherz, H. G., Suck, D., Pai, E. F. & Holmes, K. C. Atomic structure of the actin: DNase I complex. *Nature* **347**, 37–44 (1990).
31. Tobacman, L. S. & Korn, E. D. The kinetics of actin nucleation and polymerization. *J. Biol. Chem.* **258**, 3207–14 (1983).

32. Squire, J. M. & Morris, E. P. A new look at thin filament regulation in vertebrate skeletal muscle. *FASEB J.* **12**, 761–71 (1998).
33. Webb, M. *et al.* The myosin duty ratio tunes the calcium sensitivity and cooperative activation of the thin filament. *Biochemistry* **52**, 6437–44 (2013).
34. McKillop, D. F. & Geeves, M. a. Regulation of the interaction between actin and myosin subfragment 1: evidence for three states of the thin filament. *Biophys. J.* **65**, 693–701 (1993).
35. Minkoff, L. & Damadian, R. Biological ion exchanger resins: XI. Actin in *Escherichia coli*. *Physiol. Chem. Phys.* **9**, 399–403 (1977).
36. Mijailovich, S. M., Li, X., Griffiths, R. H. & Geeves, M. a. The Hill model for binding myosin S1 to regulated actin is not equivalent to the McKillop-Geeves model. *J. Mol. Biol.* **417**, 112–28 (2012).
37. Sich, N. M. *et al.* Effects of actin-myosin kinetics on the calcium sensitivity of regulated thin filaments. *J. Biol. Chem.* **285**, 39150–9 (2010).
38. Brown, a. Slow axonal transport: stop and go traffic in the axon. *Nat. Rev. Mol. Cell Biol.* **1**, 153–6 (2000).
39. Chevalier-Larsen, E. & Holzbaur, E. L. F. Axonal transport and neurodegenerative disease. *Biochim. Biophys. Acta* **1762**, 1094–108 (2006).
40. Shah, J. V & Cleveland, D. W. Slow axonal transport: fast motors in the slow lane. *Curr. Opin. Cell Biol.* **14**, 58–62 (2002).
41. Terada, S. Where does slow axonal transport go? *Neurosci. Res.* **47**, 367–372 (2003).
42. Wang, L., Ho, C. L., Sun, D., Liem, R. K. & Brown, a. Rapid movement of axonal neurofilaments interrupted by prolonged pauses. *Nat. Cell Biol.* **2**, 137–41 (2000).
43. Roy, S. *et al.* Neurofilaments are transported rapidly but intermittently in axons: implications for slow axonal transport. *J. Neurosci.* **20**, 6849–61 (2000).
44. Brady, S. T. Neurofilaments run sprints not marathons. *Nat. Cell Biol.* **2**, E43–5 (2000).
45. Lasek, R. J., Garner, J. a & Brady, S. T. Axonal transport of the cytoplasmic matrix. *J. Cell Biol.* **99**, 212s–221s (1984).
46. Deniz, A. a, Mukhopadhyay, S. & Lemke, E. a. Single-molecule biophysics: at the interface of biology, physics and chemistry. *J. R. Soc. Interface* **5**, 15–45 (2008).

47. Huxley, H. E. Fifty years of muscle and the sliding filament hypothesis. *Eur. J. Biochem.* **271**, 1403–15 (2004).
48. Kron, S. J. & Spudich, J. a. Fluorescent actin filaments move on myosin fixed to a glass surface. *Proc. Natl. Acad. Sci. U. S. A.* **83**, 6272–6 (1986).
49. Spudich, J. a. Molecular motors: forty years of interdisciplinary research. *Mol. Biol. Cell* **22**, 3936–9 (2011).
50. Brown, S. S. Nucleation of polar actin filament assembly by a positively charged surface. *J. Cell Biol.* **80**, 499–504 (1979).
51. Sheetz, M. P. & Spudich, J. A. Movement of myosin-coated fluorescent beads on actin cables in vitro. *Nature* **303**, 31–35 (1983).
52. Hynes, T. R., Block, S. M., White, B. T. & Spudich, J. a. Movement of myosin fragments in vitro: domains involved in force production. *Cell* **48**, 953–63 (1987).
53. Yanagida, T., Nakase, M., Nishiyama, K. & Oosawa, F. Direct observation of motion of single F-actin filaments in the presence of myosin. *Nature* **307**, 58–60 (1984).
54. Lu, H. P. Single-Molecule Enzymatic Dynamics. *Science (80-.)*. **282**, 1877–1882 (1998).
55. Thompson, R. E., Larson, D. R. & Webb, W. W. Precise nanometer localization analysis for individual fluorescent probes. *Biophys. J.* **82**, 2775–83 (2002).
56. Jaqaman, K. *et al.* Robust single-particle tracking in live-cell time-lapse sequences. **5**, (2008).
57. SAXTON, M. Single-particle tracking: models of directed transport. *Biophys. J.* **67**, 2110–2119 (1994).
58. DIETRICH, C. Relationship of Lipid Rafts to Transient Confinement Zones Detected by Single Particle Tracking. *Biophys. J.* **82**, 274–284 (2002).
59. Manzo, C. & Finzi, L. Quantitative analysis of DNA-looping kinetics from tethered particle motion experiments. *Methods Enzymol.* **475**, 199–220 (2010).
60. Das, R., Cairo, C. W. & Coombs, D. A hidden Markov model for single particle tracks quantifies dynamic interactions between LFA-1 and the actin cytoskeleton. *PLoS Comput. Biol.* **5**, e1000556 (2009).

Chapter 2 - A Non-Averaged Displacement Analysis to Determine Rates of Transition Among Multiple States of Motion Applied to Two Systems.

Michael Sean Carter, Del Jackson, Travis Phillips, Joe Muretta, Cynthia Mastick, and Josh E. Baker

Department of Biochemistry, University of Nevada School of Medicine, Reno Nevada 8955

Abstract

Microscopic motions of fluorescently-labeled single molecules or small particles switch between different dynamic states such as bound, freely diffusing, and directionally moving. In many systems determining the kinetics of transitions between these dynamic states is important for establishing basic molecular mechanisms of molecule interactions. Techniques such as mean squared displacement (MSD) analysis average displacements from different dynamic populations. In this study, we develop a Non-Averaging Displacement Analysis (NADA) for determining the kinetics of switching between multiple dynamic states. With this technique trajectory displacements at different window widths are plotted in a 2D histogram from which different states of motion can be visually segregated into pure populations and the decay in the intensity of a given population is used to obtain the transition rates out of that population. To verify the accuracy and precision of NADA, we simulated trajectories that stochastically switch between tethered and directed movement. The lifetimes of the tethered and directed states measured with NADA were consistent with simulation parameters. We

used NADA to analyze two experimental datasets. First, we analyzed trajectories of fluorescently-labeled actin fragments binding to and dissociating from skeletal myosin subfragment-1 (skS1) bound to a coverslip surface and determined rate constants for actin-skS1 attachment (k_{att}) and detachment (k_{det}) as a function of skS1 density. Our measured values are consistent with previously published values obtained from solution kinetics. Second, a NADA analysis of the trajectories associated with GLUT4- GFP vesicle transport in 3T3-L1 adipose precursor cells showed switching between three different dynamic states of motion (tethered, diffusing, and motile) with kinetic rates that vary with cell crowding. These results demonstrate that NADA is a tractable technique for visually segregating multiple dynamic populations and for determining the lifetimes of these dynamic states.

Introduction

Single molecules in biological systems often switch between different dynamic behaviors such as the transition from free diffusion to tethered diffusion associated with ligand binding ¹ or the stop-and-go motility of molecular motors ². Changes in particle dynamics are indicative of changes in physical states (Fig. 1A). Here we have developed an analysis method that allows us to readily identify different dynamic populations in a trajectory (Fig. 1B) and measure the kinetics of transitions between them.

Single Particle Tracking (SPT), which is the use of computer vision algorithms to record the position of single particles labeled with fluorescent markers over several seconds, has provided tremendous insights into the mechanisms by which molecules bind, and move both in vivo and in vitro. SPT data has shown that single particle trajectories are often complex, consisting of different states of motion such as diffusive movement, directed transport ³ and periods of static binding or caging ⁴ due to interactions with other molecules or particles. These mechanistic details are lost when standard averaging methods are used to analyze single particle trajectories. When a molecule's motion appears to be primarily diffusive, trajectories are typically analyzed to obtain a single diffusion coefficient. When a molecule's motion appears to be primarily directed, trajectories are often analyzed to obtain an average velocity. In a mean-square displacement (MSD) analysis, periods of short binding decrease the apparent diffusion coefficient, D , and the α -value. In measuring the speed of directed movement, short binding events result in an apparent average velocity that is less than the maximal velocity of movement.

Change-point analysis ⁵ and Hidden Markov models ⁶ are two approaches used to extract multiple populations from trajectories. The goal of both approaches is to identify specific points in a trajectory at which a state transition occurs. Both are black box approaches that are difficult to develop and implement and that require *a priori* definitions of states. Here we present an alternative lit box approach that allows one to visualize trajectories in a single graphic, from which multiple populations can be directly identified, characterized and their lifetimes determined. This technique is relatively easy to implement and can be applied to a wide range of different systems.

Here we describe, and test a novel approach we call Non-Averaged Displacement Analysis (NADA) that can resolve different dynamic populations within single particle trajectories. Similar to a MSD analysis, particle displacements are plotted at different time windows. However, rather than plotting MSD values averaged for each time window, in NADA all particle displacements within a given time window are plotted in a 2D histogram. In this plot, diffusive, static, and motile populations can be visually segregated. The lifetime of a given dynamic population is determined by fitting to a single exponential the decay in the intensity of a given population with increasing window width (similar to lifetime determination in a mean-variance analysis ⁷).

To test the ability of NADA to resolve multiple dynamic populations in a noisy system, we simulated single particle trajectories (with added positional error) that stochastically switched between bound and unidirectional states. Each state of motion was evident as a distinct population within the 2D histogram. We used NADA to measure the lifetime of the different states of motion and the measured lifetimes were similar to the inverse of the rate constants used in the kinetic simulations.

To demonstrate the utility of NADA to analyze both *in vitro* and *in vivo* experimental data, we used NADA to characterize multiple dynamic populations in two different experimental datasets. The first experimental system is an *in vitro* single molecule binding assay. In this system actin fragments bind to and dissociate from single myosin heads on a motility surface. The heads are separated spatially on the μm scale. This leads to a high probability that the behavior of the actin fragments reflects interactions with single myosin molecules. The actin fragments were tracked using traditional SPT methods to generate trajectories containing periods of free diffusion and periods of tethered diffusion. Using NADA to analyze these actin fragment trajectories, we were able to resolve two distinct populations characterized by free diffusion (when the actin was dissociated from skS1) and tethered diffusion (when the actin was bound to the skS1). Lifetimes for each population were determined from the decay in intensity of each population with increasing window width. Rate constants determined from these lifetimes are consistent with previously established values.

The second experimental system is GLUT4-GFP vesicle transport in 3T3-L1 adipose precursor cells. The transport of Glut4-GFP vesicles in these cells has been previously characterized as consisting of long distance directed motion interrupted by periods of pausing or diffusion⁸. The long distance transport observed in these cells is driven by microtubule motors. The tethered state presumably results from transient tethering or caging of the vesicle to intracellular components⁸. The diffusive state presumably occurs when vesicles are detached from the track. We used TIRF microscopy and SPT to obtain trajectories of GLUT4-GFP vesicles transported in 3T3-L1 cells; using NADA we resolved three distinct populations of motion: directed

transport over long distances, diffusive motion and tethered diffusion/bound. From the decay in intensity for each population with increasing window width, we determined the lifetime of all three populations. These experiments were repeated at different concentrations of external sucrose to determine the effects of crowding on these parameters. Our results show that sucrose slows the average vesicle velocity by decreasing the lifetime of the directed population and increasing the lifetime of the tethered population with no significant change to the lifetime of the diffusive state.

These results show that NADA can accurately separate multiple dynamic populations of motion from single particle trajectories and that using this method we can measure the lifetimes of these populations. The NADA method will prove an important tool in the future analyses of a wide range of different in vitro and in vivo systems.

Materials and Methods

Simulations

In order to test NADA we simulated single particle trajectories that switched between directed and tethered states of motion according to the kinetic scheme in Fig. 1A. The state of the particle was determined at each step by obtaining a random number between 0 and 1 then comparing that number to the transition probability $P_{\text{ChangeState}} = \Delta t \times K$ where Δt is the duration of the time step and K is the sum of rate constants leaving the current state. If the random number is less than the $P_{\text{ChangeState}}$ then the particle changes state.

The position (X_{i-1}, Y_{i-1}) of a simulated particle at time $t = i$ is calculated from the position at the previous time $t = i - 1$ as $X_i = X_{i-1} + d_i * \cos \theta_i$ and $Y_i = Y_{i-1} + d_i * \sin \theta_i$

where d_i and θ_i are the polar coordinates for a displacement vector, which are calculated differently depending on the state (free diffusion, tethered diffusion, or directed movement) of the particle.

When the particle is in the directed state, $d_i = V \cdot \Delta t$ and $\theta_i = \theta_{i-1} + RND1 \cdot 0.04 \frac{\text{radians}}{\text{s}} \cdot \Delta t$ where V is the average velocity (input parameter) and Δt is the change in time per step of the simulation. When in the diffusive state $d_i = \sqrt{4D\Delta t} \cdot RND1$ and, $\theta_i = 2\pi \cdot RND1$ where D is the coefficient of diffusion (input parameter) and $RND1$ is a random number between 0 and 1. When in the tethered diffusion state, d_i and θ_i are defined as was used for free diffusion with the added condition that any displacement vector that would result in a point with a distance greater than the tethered radius from the first point in the tethered state would be rejected and a new random displacement vector would be calculated using the same method as before until the resulting point fell within the tethered radius.

To better mimic experimental data and explore the role of positional precision, additional error was added to the recorded position of each time step after the position of the particle was passed to the next time step. This positional measurement error allowed us to decrease the precision with which the particle location was resolved without effecting the actual position for the next time step. This error was implemented by adding an additional displacement of magnitude, $d_{\text{err}} = P_{\text{error}} \cdot RND1$, at direction, $\omega = 2\pi \cdot RND1$ to the measured position without changing the real position of the particle. For the simulations used in this paper we chose a positional error that was

comparable to the per frame displacement of both directed and the diffusing populations.

Transport of GLUT4-GFP Vesicles in 3T3-L1 Fibroblasts

3T3-L1 fibroblasts (CCL 92.1; American Type Culture Collection) were maintained in DMEM (high glucose, with 2 mM L-glutamine, 50 units/ml penicillin, and 50 µg/ml streptomycin) supplemented with 10% calf serum (CS; Hyclone). For experiments, 40 mm diameter round borosilicate coverslips were coated with fibronectin, washed with PBS and placed into 60 mm tissue culture dishes. 3T3-L1 fibroblasts were plated into dishes at a cell density of 0.34×10^5 cells cm^{-2} . At confluence (3 days after plating), the cells were switched to DMEM supplemented with 10% fetal bovine serum (FBS; Hyclone; characterized) and 5 µM Troglitazone (Calbiochem). One day post confluence (day 0), the cells were induced to differentiate into adipocytes essentially as described⁹. Briefly, cells were incubated for 3 days in differentiation medium (DMEM, 10% FBS, 5 µM Troglitazone, dexamethasone, isomethylbutylxanthine, and insulin), 3 days in insulin medium (DMEM, 10% FBS, Troglitazone, and insulin), and then 4–5 days in DMEM, 10% FBS. Cells were used 8–11 days after the initiation of differentiation. 3T3-L1 fibroblasts were used 2–3 days post confluence, and differentiation media was replaced with Opti-MEM for microscopy analysis. Culture buffers and temperature was maintained during imaging using a FCS2 closed flow chamber system. Movies of GLUT4-GFP vesicles transport in 3T3-L1 fibroblasts were collected at an average exposure time of 0.05 seconds per frame using TIRF illumination to mitigate background fluorescence. Each cell movie was filtered using ImageJ's built-in rolling ball filter with a radius of 5 pixels to remove the large non-

moving fluorescent background present in these cells and to even the illumination of the image¹⁰. The contrast of each frame was adjusted for maximum contrast to compensate for photobleaching during the recording. The GLUT4-GFP vesicles in the resulting images were tracked using a custom centroid based tracking algorithm. Vesicle trajectories that traveled more than 2 μm and had durations of greater than 2.5 seconds were used for NADA analysis. This process was repeated for 4 cells per sucrose concentration and yielded an average of 75 trajectories per condition with an average duration of 8.8 s (175 points) per trajectory. The 4% sucrose has an average track length of 14.4 s (287 frames).

Protein Purification

Skeletal muscle myosin was prepared from rabbit psoas muscle as previously described and stored in 50% glycerol at -20°C ¹¹. Skeletal muscle myosin subfragment-1 (skS1) was prepared by papain digestion of myosin^{11,12}. Actin was isolated from rabbit psoas muscle and stored on ice at 4°C ¹³. Before use, actin was mixed with equimolar phalloidin-Alexa-488 (Invitrogen, Carlsbad, CA USA) and incubated overnight at 4°C .

Buffers

A myosin buffer of 300 mM KCl, 25 mM imidazole, 1 mM EGTA, and 4 mM MgCl_2 was used to dilute skS1 to experimental concentrations used in single molecule binding assays¹³. An actin buffer of 50 mM KCl, 50 mM imidazole, 2 mM EGTA, 8 mM MgCl_2 , 10 mM DTT, and an oxygen scavenger system (292 $\text{mg}\cdot\text{ml}^{-1}$ glucose, 1.63 $\text{mg}\cdot\text{ml}^{-1}$ glucose oxidase, and 2.25 $\text{mg}\cdot\text{ml}^{-1}$ catalase) was used to dilute actin for single molecule binding assays.

***In Vitro* Single Molecule Binding Assay**

The *in vitro single binding assay* is a modified landing assay^{14,15} developed to determine kinetics of A-M binding and dissociation in *in vitro motility* conditions. In this assay actin fragments bind and dissociate from skS1 bound to a coverslip creating both freely diffusing and tethered states of motion. We incubated glass cover slips with papain-digested skS1 in myosin buffer at each assay concentration on ice for 20 min. We then blocked the cover slips with bovine serum albumin at 1 mg·ml⁻¹ in actin buffer on ice for 20 min. Actin fragments were prepared by sonicating Alexa-488-labeled actin at a concentration of 10 nM in actin buffer containing 1% methylcellulose and no MgATP on ice until the resulting actin fragments were approximately 1 μm in length. These actin fragments were added to the flow chamber and were visualized using a 488 nm excitation laser on a Nikon TE2000 total internal reflection fluorescence microscope with a 100x oil immersion objective (Nikon plan Apochromat NA 1.45). Images of the actin fragments were collected at 10 frames s⁻¹ over a 3 min period using a Roper Cascade 512B (Princeton Instruments, Trenton, NJ) EMCCD digital camera. The resulting movies were then analyzed using our custom centroid tracking algorithm. Tracks with durations less than 2 s and any tracks that did not displace at least 1 μm were excluded from the analysis. An average of 256 tracks per condition with an average duration of 8 s (80 frames) per track were analyzed per condition. These trajectories were analyzed using NADA to obtain the rates of switching for the tethered and diffusing populations at each condition. The tethered population was chosen as the region of the NADA histogram with a displacement less than 0.1 μm, which corresponds to the horizontal population running parallel to the x axis (Figure 4 D

– F, green lines). The freely diffusing population was chosen as the region within $0.2 \mu\text{m}$ of the center of the distribution of displacements for the $0 \mu\text{g/ml}$ S1 condition at small window width (Figure D – F, cyan lines) and this definition of the diffusion population was used on all other conditions. No directed population was observed in these experiments.

Custom Centroid Tracking Algorithm

All experimental single particle tracks were constructed as follows. Individual regions of fluorescence were segmented from the background by intensity-based thresholding. The level of the threshold was manually determined for each movie such that the particles of interest were adequately represented without including excess noise from the background or joining of adjacent fluorescent regions. Any fluorescent region with an area less than a given constant was removed from the analysis and the centroid for each of the remaining regions was calculated as previously described¹⁶. Trajectories were created by linking the centroid positions through a simple displacement minimization algorithm. This linking algorithm calculates all of the possible ways of connecting the detected objects in one frame to the next frame and chooses linkages between objects that minimize the total displacement between frames. Trajectories with durations less than the user-provided minimum value were excluded from the analysis to ensure that the motion of any particle contained enough sampled points to properly describe its behavior. Similarly trajectories that did not displace a total minimum distance while observed were also excluded from consideration to prevent the analysis of non-moving objects that may be nonspecifically bound to the surface.

NADA Algorithm

The non-averaging displacement analysis (NADA) uses changes in the distribution of displacements within single particle trajectories to detect and measure the lifetimes of different states of motion present within the trajectory. The NADA method is performed by calculating the distance, $d(i, ww) = \sqrt{(X_{i+ww} - X_i)^2 + (Y_{i+ww} - Y_i)^2}$, from one point in the trajectory, (X_i, Y_i) , at time index i to a second position (X_{i+ww}, Y_{i+ww}) at a time index $i+ww$ for all possible i and ww combinations.

For data sets containing multiple trajectories, all of the trajectories for a given experimental condition were combined into a single trajectory by first translating the first trajectory in the data set so that its first point is located at 0,0. The trajectory was then rotated about its first point by a random angle. Each subsequent trajectory was added to the previous trajectory by translating the added trajectory so its first point overlaps with the last point of the combined trajectory. The overlapping point created by the translation was dropped and the trajectory being added was rotated a random angle about its first point. This process was repeated until all of the trajectories for the condition were combined into a single trajectory. The combined trajectory was then used in the NADA calculation as described above.

In order to visualize the distribution of displacements as a function of window width, we developed custom software that generates a NADA histogram, a 2D histogram with window width (ww) plotted on the x axis and displacements, $d(i, ww)$, plotted on the y axis with the intensity of each pixel being proportional to the number of counts in that bin.

Once the distribution of displacements from a collection of trajectories has been visualized as a 2D histogram, the individual states of motion contained within the trajectories segregate into different populations according to the mechanical properties of each state of motion and the resulting propagation of the population as a function of increasing window width. To enhance the separation of dynamic populations, we treated the 2D histogram as an image and used image enhancement techniques such as adjusting brightness and contrast, taking the log of the intensity (bin height), adjusting the vertical and horizontal binning of the 2D histogram, and maximizing the intensity gradient (e.g., Sobel edge detection). The lifetime of each population was determined by reading the intensity of the area within the 2D histogram as a function of window width, which corresponds to the displacements generated by the state of motion being measured. Intensities, I , were plotted at each window width and then fitted to the equation:

$$I(ww) = A * \exp\left(-\frac{ww}{\tau}\right) \quad \text{Eq. 1}$$

where I is the measured intensity of the population at each window width, A is the intensity of the population at a window width of 0, ww is the window width and τ is the lifetime of the measured population.

Results

Kinetics of Single Molecule Switching between Multiple Dynamic Populations.

Single particle trajectories often switch between different states of motion. Transitions between these states occur when interactions between the observed particle and its surroundings undergoes a physical change. Figure 1A is a kinetic scheme that describes the transition between three commonly observed states of motion: directed (movement in a single direction), free diffusion (movement with random direction) and tethered (movement in a random direction limited to a maximum distance). While different methods are used to analyze single molecule trajectories, the most tenable and commonly used are averaging techniques such as the Mean Squared Displacement analysis (MSD). Averaging methods accurately describe the physical state of a single molecule when there is only one dynamic state present, but these techniques cannot resolve multiple states of motion.

Figure 1B is a simulated trajectory for a single particle that switches between the three states shown in Fig. 1A. In the directed state (blue points) the particle moves at a velocity of $6 \mu\text{m/s}$ for an average lifetime of 25 ms. In the free diffusion state (cyan points) the particle moves with a diffusion coefficient of $0.20 \mu\text{m}^2 \text{s}^{-1}$ and an α value of 1 for an average lifetime of 25 ms. In the tethered diffusion state (green points) the particle moves with a diffusion coefficient of $0.20 \mu\text{m}^2 \text{s}^{-1}$ with a tether radius of $0.05 \mu\text{m}$ and with a lifetime of 25 ms. All three dynamic states are simulated with a positional measurement noise of 20 nm per point (1 point = 2.5 ms) as described in the methods section.

The NADA method visualizes multiple dynamic states within single particle trajectories by calculating the displacement from each point in the trajectory to a point a specified time interval, referred to as the window width, later in the trajectory. These calculations are repeated at all possible window widths and visualized using custom software as a NADA histogram, a 2D histogram of displacements (y axis) verses window widths (x-axis) with the intensity of each pixel being proportional to the number of counts in that bin. The distribution of intensities in the NADA histogram is dependent on the mechanical properties and rates of propagation for the state of motion which generated the displacement. Figure 1C is a composite of 3 different NADA histograms calculated from simulated trajectories exhibiting either directed, tethered or free diffusion with the same mechanical properties as the matching states of motion present in Figure 1B. Figure 1C clearly shows that the individual dynamic states are visually resolved in a NADA histogram. For the directed population, the average displacement at each window width is plotted (blue line), and a least squares linear fit of this data gives a slope of 6.02 $\mu\text{m/s}$, which is consistent with the simulated velocity. The average displacement of the tethered population (green line) varied from 0.038 μm to 0.045 μm . The linear fit of this data resulted in a slope of zero with a y off set of 0.042 μm . The average displacement for the diffusive population (cyan line) varies with window width approximately as $\bar{d}(ww) \approx \sqrt{4Dww^\alpha}$, where \bar{d} is the averaged displacement, D is the diffusion coefficient and t is the window width. When the means squared displacement of the simulated trajectory is calculated and fit to the standard MSD equation ($\langle r^2 \rangle = 4Dww^\alpha$), we obtain a D coefficient of 0.204 $\mu\text{m}^2 \text{s}^{-1}$ with an α of 1.

Figure 1C shows a combined NADA histogram obtained from homogeneous (no transitions) trajectories simulated from each of the three states in Fig. 1A. In contrast, Fig. 1D illustrates how the NADA histogram changes when we incorporate kinetic transitions between the same three dynamic states. Specifically, while the three populations in Fig. 1C are evident in Fig. 1D, in Fig. 1D the intensities of each state decay with increasing window width, and from the observed rate of decay we can determine the lifetimes of each state in Fig. 1A. The NADA histogram image contains areas of high intensity, each of these areas represent displacements created while the simulated particle was in only one state during the observational window. The solid lines in Fig. 1D are replotted from the average displacements of the pure populations in Fig. 1C, showing that at window widths less than the state lifetimes (25 ms), the three populations are visually resolvable, and from those populations we can accurately measure the properties of each state (e.g., D , α , V , and the tether length which is the maximum displacement of the particle while in the tethered state). In a NADA histogram, the decrease in the intensity of a given population with increasing window width can be fit to a single exponential to obtain a time constant that is equal to the inverse sum of rates leaving that population (see below). At window widths much larger than the time constants, the populations merge into a single blended population.

Using NADA to Determine Kinetics of Switching Between Multiple Dynamic States in Simulated Trajectories.

To test NADA's accuracy in determining the kinetics of switching between multiple dynamic states, we simulated trajectories that switch between states of directed ($V = 6 \mu\text{m/s}$) and tethered diffusion ($D = 0.2 \mu\text{m}^2 \text{s}^{-1}$; tether radius = $0.1 \mu\text{m}$) using

combinations of lifetimes (5, 10, 20, 40, 80, 160, 320, and 640 ms) for both states. Positional noise of 20 nm per point (1 point = 2.5 ms) was added in each simulated trajectory. Figure 2 (top) shows sample trajectories simulated with a directed lifetime of 40 ms and tethered lifetimes of 5 ms, (A), 80 ms, (B) and 640 ms (C). Figure 2 (bottom) are the resulting NADA histograms for the above trajectories. For simulations where the directed state is significantly longer than the tethered lifetime the trajectory (Fig. 2A) is dominated by long periods of directional movement punctuated by short periods of tethered diffusion, and the corresponding NADA histogram (Fig. 2D) is dominated by the directed population, where the displacement increases linearly with window width, having a slope equal to the simulated velocity. For simulations where the motile lifetime is much shorter than the tethered lifetime the resulting trajectory (Fig. 2C) is composed of large diffusive nodes with a radius equal to the tether length, interrupted by brief periods of directed movement. The corresponding 2D histogram (Fig. 2F) is composed primarily of a horizontal population of displacements that are less than the tether distance. For simulations with comparable directed and tethered lifetimes (Fig. 2B) the resulting trajectories are composed of equal periods of directed and tethered movements, exhibiting a “nodes on a string” appearance. The corresponding NADA histogram (Fig. 2E) contains both directed and tethered populations. At large window widths, the merged population resembles a single moving population with a broad distribution of velocities.

To test the accuracy of NADA in determining lifetimes for the motile and tethered states we used the simulated NADA histograms from Fig. 2. We measured and plotted the decay in intensity for the 2D histogram as a function of window width for both the

directed and tethered states. State lifetimes were obtained from single exponential fits to these intensity decays (See Equation 1 in the Methods section). The first and most critical step in this process is to identify populations in the NADA histogram. This is done through a combination of i) identifying peaks in the intensity distributions at low window widths, ii) performing NADA on pure populations, and iii) applying edge detection algorithms to NADA histograms (as was done in Fig. 2D – 2F). For the NADA histograms in Fig. 2, the directed state was defined by a line (blue) fitted to the peak of the directed population. The slope of this line is V , and the intensity, $I(ww)$, of the directed population at a given window width was calculated as the number of data points in the NADA histogram with a displacements greater than $V \cdot ww$, normalized to the total number of displacements observed at that window width. The tethered state was defined by a line (green) of slope zero fitted to the peak of the tethered population. The intensity, $I(ww)$, of the tethered state at a given window width was calculated as the number of data points having displacements below this line normalized to the total number of displacements observed at that window width. For the directed population, in Fig. 3A we plotted $I(ww)$ at each window width for simulations run with a tethered lifetime (τ_T) of 80 ms and directed lifetime (τ_D) values of 10 (red), 40 (blue), 160 (magenta) and 640 ms (yellow). For the tethered population, in Fig. 3B we plotted $I(ww)$ at each window width for simulations run with a τ_D of 80 ms and τ_T values of 10 (red), 40 (blue), 160 (magenta) and 640 ms (yellow).

We fitted the data in Figs. 3A and 3C to Eq. 1 to obtain lifetimes for the directed and tethered states respectively. These measured lifetimes are plotted in Figs. 3B and 3D against the corresponding simulated lifetimes for the directed and tethered states,

respectively. These plots are linear with slopes of 0.97 (Fig. 3B) and 0.94 (Fig. 3D), demonstrating that NADA accurately estimated the lifetimes of directed and tethered states in the simulated trajectories.

Using NADA to Determine the Kinetics of Experimentally Measured Single Actin Fragments Binding to (k_{DT}) and Dissociating from (k_{TD}) skS1 on a Coverslip Surface.

In order to test NADA's ability to experimentally determine the rate constants for transitions between dynamic populations in *in vitro* experimental systems, we conducted a single molecule binding assay using TIRF microscopy to track fluorescently labeled actin fragments freely diffusing in two dimensions over a coverslip coated with different densities of skeletal myosin S1 (M) in the presence of methylcellulose. In this system fragments can transition between two different biochemical states: bound to skS1 (tethered in Fig. 1A) and dissociated (diffusing in Fig. 1A). The lifetime of the diffusing population can be predictably changed by altering the myosin S1 density on the coverslip; specifically the time it takes for an actin filament to find a myosin S1 on the surface (i.e., τ_D) increases linearly with decreasing myosin density. Figure 4 (A – C) shows example trajectories obtained with SPT analysis of actin fragments diffusing over a coverslip surface having a myosin S1 density that was obtained by incubating the coverslip for two min with a solution of 0.00 (A), 1.25 (B), and 5.00 $\mu\text{g ml}^{-1}$ skS1 (C) in myosin buffer. Harris and Warshaw showed that for myosin concentrations less than 100 $\mu\text{g ml}^{-1}$, the myosin density on the coverslip surface increases linearly with the myosin concentration in the incubation buffer¹⁷. It is evident from these trajectories that the lifetime of the diffusing population decreases with increasing S1.

Figure 4 (D - F) shows NADA histograms obtained from actin fragment trajectories acquired under the conditions in Fig. 4 (A - C). The tethered population (A · M) was defined in the NADA histograms as the region of displacement with a magnitude less than $0.1 \mu\text{m}$ (area under green lines in Fig 4 D - F). The diffusing population (A + M) was defined by first identifying the peak intensity at low window widths in a NADA histogram obtained from a primarily diffusive population (no skS1, Figs. 4A and 4D). The center of the peak intensity was accurately fit by the diffusion equation (not shown) with a D coefficient of $0.20 \pm 0.01 \mu\text{m}^2/\text{s}$ and α value of 0.9 ± 0.1 with an R^2 of 0.999. The intensity for the diffusing population was then obtained by selecting all displacements within $0.2 \mu\text{m}$ of the peak intensity (between the cyan lines in Fig. 4D). In our analysis we assumed that the diffusion coefficient does not change with myosin density, and so we used the same diffusing population in all experiments (e.g., cyan lines in Figs. 4E and 4F). For the no-myosin experiment, a tethered population is observed, presumably resulting from non-specific binding of actin fragments to the coverslip. However, this population has a low occupancy (intensity) relative to the diffusing population because there are relatively few non-specific binding events.

We can determine from the NADA histograms in Fig. 4 the skS1-dependence of the A-M attachment ($1/\tau_D$) and detachment ($1/\tau_T$) rates. Figure 4G shows intensities for the diffusing population plotted as a function of window width for 0 (black), 0.625 (green), 2.50 (cyan), and 5.0 (magenta) $\mu\text{g/ml}$ skS1. Figure 4H contains example intensities for the tethered population plotted as a function of window width for 0 (black), 0.3125 (red), 0.625 (green), and 1.25 $\mu\text{g/ml}$ (blue) skS1. The data in Figs. 4G and 4H were fitted to single exponentials to determine τ_T and τ_D at different [skS1].

In Fig. 4I, the skS1-dependence of τ_T and τ_D was plotted, showing that $1/\tau_D$ increases linearly with skS1, consistent with a binding reaction, and that τ_T is independent of skS1 as expected for a dissociation reaction. In the absence of skS1, we observed a relatively low frequency of binding events with bound lifetimes 3-fold shorter than the bound lifetimes observed at high skS1. These low-frequency, short-lifetime events result from non-specific binding interactions between actin fragments and the coverslip surface. The frequency of actin-skS1 binding events increases with increasing skS1, resulting in a measured bound lifetime that is influenced more by actin-skS1 binding than by actin-surface binding. At high skS1, τ_T does not change with increasing skS1, suggesting that τ_T is dominated by actin-skS1 binding. At the lowest skS1 concentration used ($0.3 \mu\text{g ml}^{-1}$), τ_T is intermediate between our measured non-specific (no skS1) and specific (high skS1) lifetimes, indicating that the frequency of actin-skS1 binding is comparable to the frequency of actin-surface binding. The average bound lifetime, τ_T , measured at $[\text{skS1}] > 0.3 \mu\text{g ml}^{-1}$ is 2.5 s, giving a detachment rate constant, k_{TD} , of $0.4 \pm 0.1 \text{ s}^{-1}$ (green line), consistent with previous measurements of 1 s^{-1} ¹⁸. From the slope of a linear fit to the $1/\tau_D$ data (cyan line), we obtained an actin-skS1 attachment rate constant, k_{DT} , of $2.0 \pm 0.1 \text{ s}^{-1} \cdot (\mu\text{g ml}^{-1})^{-1}$. Here the concentration units come from the concentration of skS1 that was used to incubate the coverslip. Previous studies show a linear relationship between the incubation concentration of myosin and the resulting myosin density on the coverslip surface¹⁷.

Using NADA to Determine the Kinetics of Experimentally Measured Glut4-GFP Fusion Vesicle Transport within 3T3-L1 Cells.

We used NADA to analyze the behavior of single particles that exhibit all three dynamic states in Fig. 1. Specifically, we measured the trajectories of GLUT4-GFP fusion protein-containing vesicles in 3T3-L1 adipose precursor cells (Fig. 5A). These vesicles are actively transported through the cell over long distances, dynamically switching between the three described states of motion. Possible mechanisms for switching between these dynamic states include motor activation and deactivation, local environmental factors such as intracellular crowding, and regulatory factors such as the recruitment from or to storage populations. We observed that the addition of sucrose to experimental media significantly altered vesicle trajectories by decreasing vesicle run lengths and increasing tethered lifetimes. We obtained vesicle trajectories (Fig. 5A, red lines) in the presence of 0, 1, 2, and 4% sucrose, and performed on NADA using these trajectories. NADA histograms for 1% and 4% sucrose are shown in Figs. 5B and 5C.

In Fig. 5B all three states can be visually resolved. The tethered population appears as a horizontal yellow streak near $y = 0$. Specifically, we defined the tethered population as all intensities below $0.1 \mu\text{m}$ (green line). Harmonics of this population are observed as repeating horizontal lines at higher window widths and displacements. At low window widths and high displacements, overlaid trajectories of directed vesicles are observed as orange streaks with relatively high slopes (velocities) that intercept near the origin. In our NADA analysis, we chose the directed population to include all of these trajectory clusters that approach the origin (above the blue line). Harmonics of the directed population (trajectories with similar slopes but with non-zero intercepts) are

also observed. Finally, the diffusive population appears as a faint angled yellow streak (between cyan lines) defined for our NADA analysis as described in Fig. 4.

When we analyzed these trajectories assuming a single directed population, we obtain an average velocity of $0.5 \mu\text{m}\cdot\text{s}^{-1}$ that decreases 5-fold with addition of 4% sucrose. When we perform an MSD analysis of these trajectories assuming a single diffusive population, we obtain a D value of $0.08 \mu\text{m}^2\cdot\text{s}^{-1}$ that decreases ~8-fold with addition of 4% sucrose and an α of 0.62 that decreases to 0.46 with addition of 4% sucrose. But the NADA histograms above show that these are not single population trajectories, and when we use these histograms to characterize the three different dynamic states, we obtain values for V and D that are significantly different from those obtained from the averaging methods above.

From NADA histograms we calculated, for each sucrose concentration, the average velocity of the directed population from the slope of a linear fit (y-intercept of zero) to the displacement versus window width data for all displacements above the velocity cut-off (blue lines in Figs. 5B and 5C). These data are plotted in Fig. 5D. As expected the velocities calculated using NADA were significantly faster than the average velocities from MSD analysis, because the tethered and diffusing populations slow the average velocities. From the NADA histograms we also calculated for each sucrose concentration the diffusion coefficient and α 's for the diffusing population by fitting the center of the diffusing population (midway between the two cyan lines) to the equation $d = \sqrt{4Dww^\alpha}$. These values are plotted in Fig. 5E.

In addition to providing a visual separation of multiple dynamic populations (Fig. 5B) as well as a means to directly characterize these populations (Figs. 5D and 5E), NADA allows us to determine the lifetimes of these populations. By plotting the total intensities of the directed (above the blue line), tethered (below the green line) and freely diffusing (between cyan lines) populations as a function of window width and fitting these curves to single exponentials, we determined lifetimes for all three populations at each sucrose concentration. These data are plotted in Fig. 5F, showing that the lifetime of the tethered population increases with increasing sucrose, the lifetime of the directed population decreases with sucrose, and the lifetime of the diffusing population is unaffected.

These observations are all consistent with sucrose increasing cytosolic crowding through a hyper-osmotic effect. While a hyper-osmotic effect could result in a wide range of effects in the cell, the acute mechanism of water loss associated with hyper-osmotic shock would result in increased crowding within the cell. This crowding might slow V and D through increased viscosity; decrease the lifetime of the directed population by increasing opportunities for trapping/tethering vesicles; and increase the lifetime of the tethered population by decreasing untethering forces.

When analyzing single particle trajectories using NADA there are several properties of the analysis that need to be taken into account; first since the intensity of any population in the NADA histogram is dictated by $I(ww) = A * \exp\left(-\frac{ww}{\tau}\right)$ and at window widths longer than state lifetimes, populations merge, and states become unresolvable. This dictates that the trajectory must be sampled at a rate much faster

than one over the expected lifetime of the measured population to insure that the calculated NADA histogram will contain sufficient temporal resolution to adequately describe the decay in intensity versus window width for each population. When the trajectory is under-sampled it cannot be resolved and will instead influence the definition (V , D , α , etc.) of other populations. Also trajectories must have sufficient duration to contain several instances of switching between states to prevent the measured state lifetimes from being influenced by the average track length. Second, NADA uses differences in the rates of propagation to resolve the states of motion present in the 2D histogram. The implication of this is that an individual state of motion can only be resolved when the displacements generated by the state are different from other states present. Since different states of motion accumulate displacements as a function of window width at different rates there are often regions at low window widths where the population of displacements from different states of motion overlap. These regions can create an effective dead time where the intensity of individual populations cannot be accurately measured reducing precision with which short lifetimes can be obtained. Further any factors such as positional error or variance in V , D , or α can further obfuscate individual states at low window widths and increase the minimum window width required to distinguish multiple populations. To address these limitations, we are developing an explicit model to describe the interactions of different states of motion within the NADA histogram. This model will allow for a globalized fit of the NADA histogram utilizing not only the regions of the 2D histogram where the states of motion are resolved but from regions such as the mixed population which are currently not

used in the analysis. This will increase the precision of the analysis, allowing a wider range of populations to be analyzed.

Conclusion

In this article we have introduced a new method of analyzing complex single particle trajectories. This approach has several clear advantages over existing more black-box methods such as HMM: 1) it allows for the direct visualization of multiple dynamic states in a trajectory, 2) each state can be characterized to determine values for D , α , and V as well as the lifetimes for each state, 3) computational speed for NADA is a relatively fast, and 4) it is a white box approach that keeps the entire dataset of multiple trajectories in view at all times, allowing the user to visualize the data from which NADA outputs are determined in a manner not previously possible. Further we have demonstrated NADA's utility in analyzing multiple states of motion in a wide range of experimental systems including both *in vitro* and *in vivo* systems.

Figures

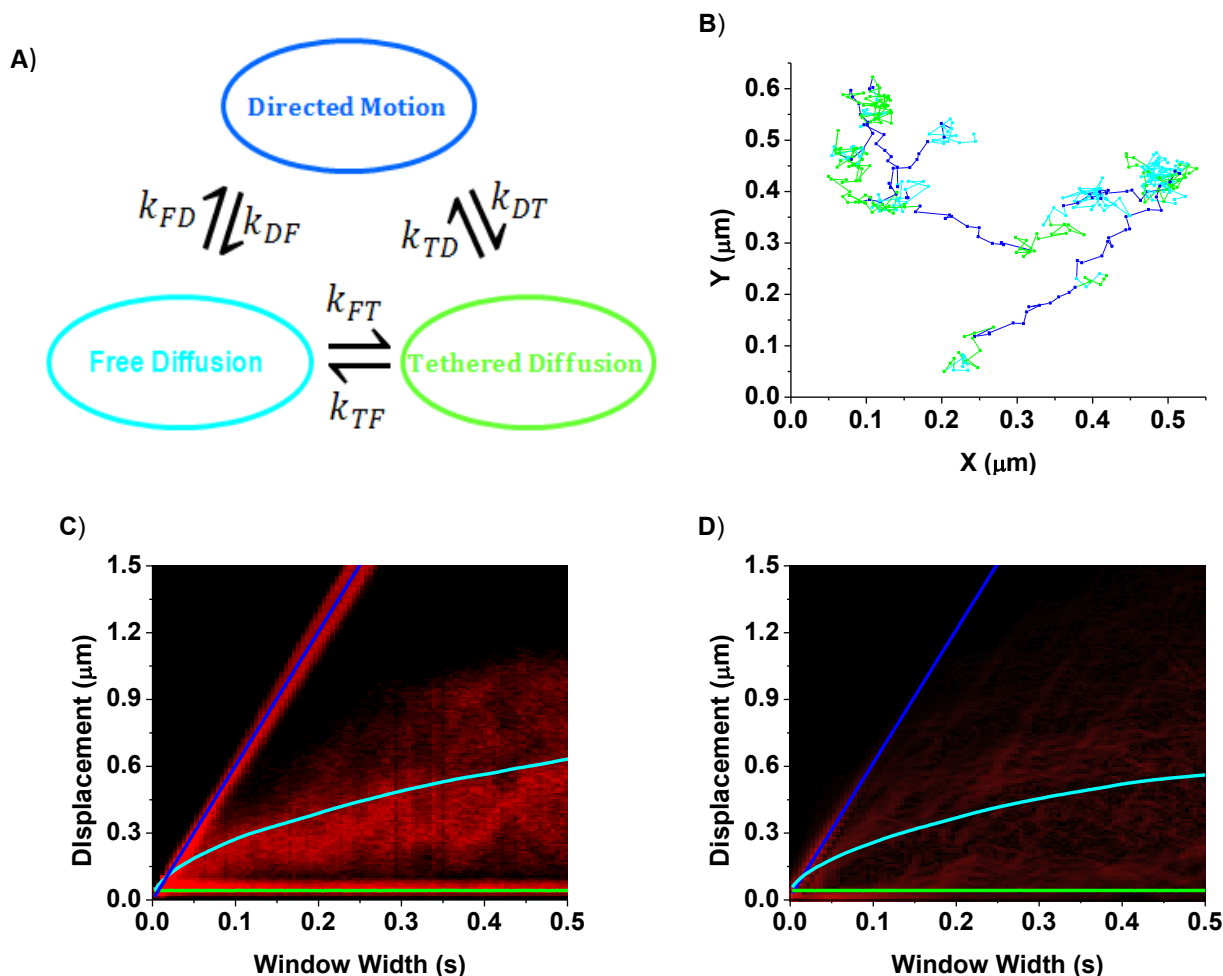


Figure 1 - Kinetics of Single Molecule Switching Between Multiple Dynamic Populations

A) A kinetic scheme showing the rates of switching between three states of motion: directed (moving in a single direction), free diffusion (movement with random direction), and tethered (movement in a random direction limited to a maximum distance). Arrows denote transitions between the states, with rate constants k that include the abbreviations T (tethered), F (freely diffusive), and D (directed). B) A one second sample of a simulated trajectory for a particle that switches between the three states of motion described in (A). The directed state (dark blue) is simulated with a constant V of $6 \mu\text{m/s}$ with an average lifetime ($1/(k_{DF} + k_{DT})$) of 20 ms. The free diffusion state (cyan) is simulated with a diffusion coefficient, D , of $0.2 \mu\text{m}^2 \cdot \text{s}^{-1}$ and a lifetime ($1/(k_{FD} + k_{FT})$) of 20 ms. The tethered diffusion state (green) is simulated with a diffusion coefficient, D , of $0.2 \mu\text{m}^2 \cdot \text{s}^{-1}$, a maximal tethered distance of $0.05 \mu\text{m}$, and an average lifetime ($1/(k_{TD} + k_{TF})$) of 20 ms. All three dynamic states are simulated with a positional measurement error of 20 nm per point (1 point = 2.5 ms). C) Combined NADA histogram of displacements versus window width obtained by summing NADA histograms of independently simulated trajectories with infinite lifetimes for each of the three states in (B). Solid lines show the average displacement for the homogeneous directed (blue), free diffusion (cyan), and tethered (green) states. D) A NADA histogram obtained from a single simulated trajectory using the same parameters as the trajectory in (B). The NADA histogram is edge enhanced with a Sobel filter. The blue, cyan, and green lines are copied from (C) for comparison.

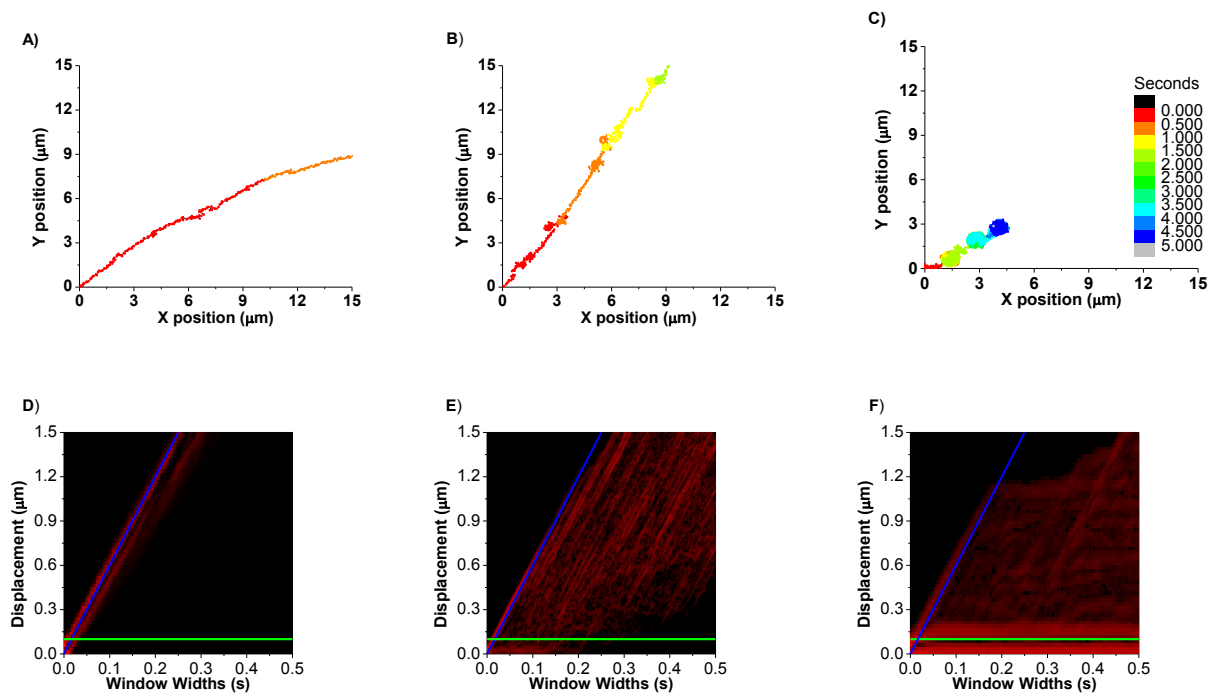


Figure 2 - NADA Applied to Simulated Directed-Tethered Trajectories.

Simulated trajectories for a single particle that transitions between a directed state with a lifetime, $1/k_{DT}$, of 40 ms and a tethered state with lifetimes, $1/k_{TD}$, of (A) 5, (B) 40, and (C) 640 ms. (D – F) NADA histograms of trajectories simulated in A – C. Histograms were edge enhanced with a Sobel filter. The blue line is the lower limit of the directed population (the set of directed trajectories with a zero intercept). The green line is the center of the tethered population.

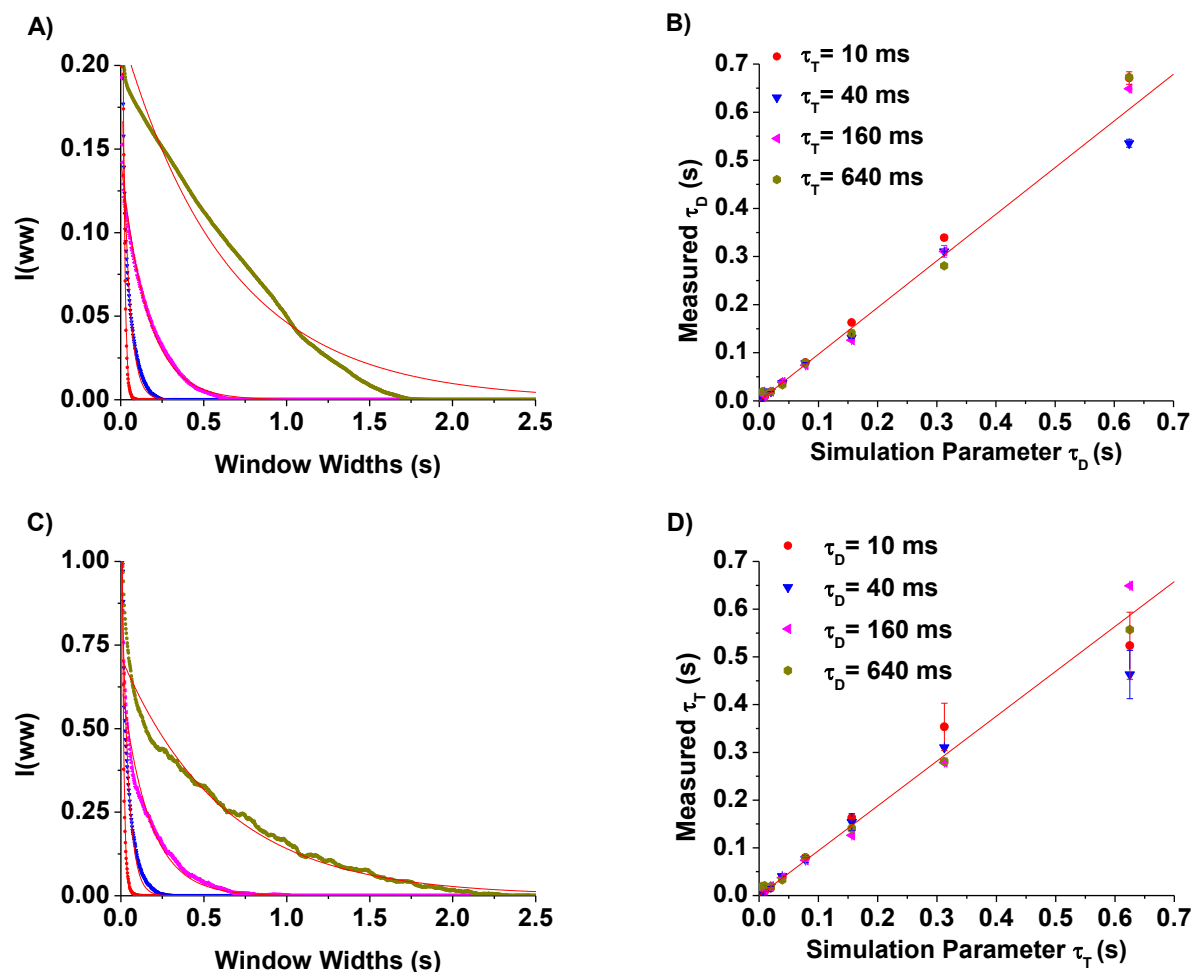


Figure 3- Determining the Lifetime of the Dynamic Populations Simulated in Figure 2.

(A) Plots of the intensity $I(ww)$ of the directed population from a NADA histogram obtained from directed-tethered trajectories simulated in Fig. 2 with tethered lifetimes (τ_T) of 80 ms and directed lifetimes (τ_D) of 10 (red), 40 (blue), 160 (pink), and 640 (yellow) ms. Intensity decays were fitted to a single exponential to obtain the lifetime ($\tau_d = 1/k_{DT}$) of the directed state at different simulated k_{DT} values. (B) Measured τ_D from simulations created with τ_T of 10 (red), 40 (blue), 160 (magenta) and 640 (yellow) and τ_D as indicated by the point's position on the x-axis. (C) Plots of the intensity of the tethered population from a NADA histogram obtained from directed-tethered trajectories simulated in Fig. 2 for with a τ_D of 80 ms and tethered lifetimes of 10 (red), 40 (blue), 160 (pink), 640 (yellow) ms. Intensity decays were fitted to single exponentials to obtain the lifetime ($\tau_T = 1/k_T$) of the tethered state at different simulated k_{TD} values. (D) Measured τ_T from simulations created with τ_D of 10 (red), 40 (blue), 160 (magenta) and 640 (yellow) and τ_T as indicated by the point's position on the x-axis. The linear relationship between simulated and measured τ values indicated by the linear fits (red lines) to the data in (B) and (D) demonstrates that NADA can accurately determine the lifetimes of different dynamic states in single particle trajectories.

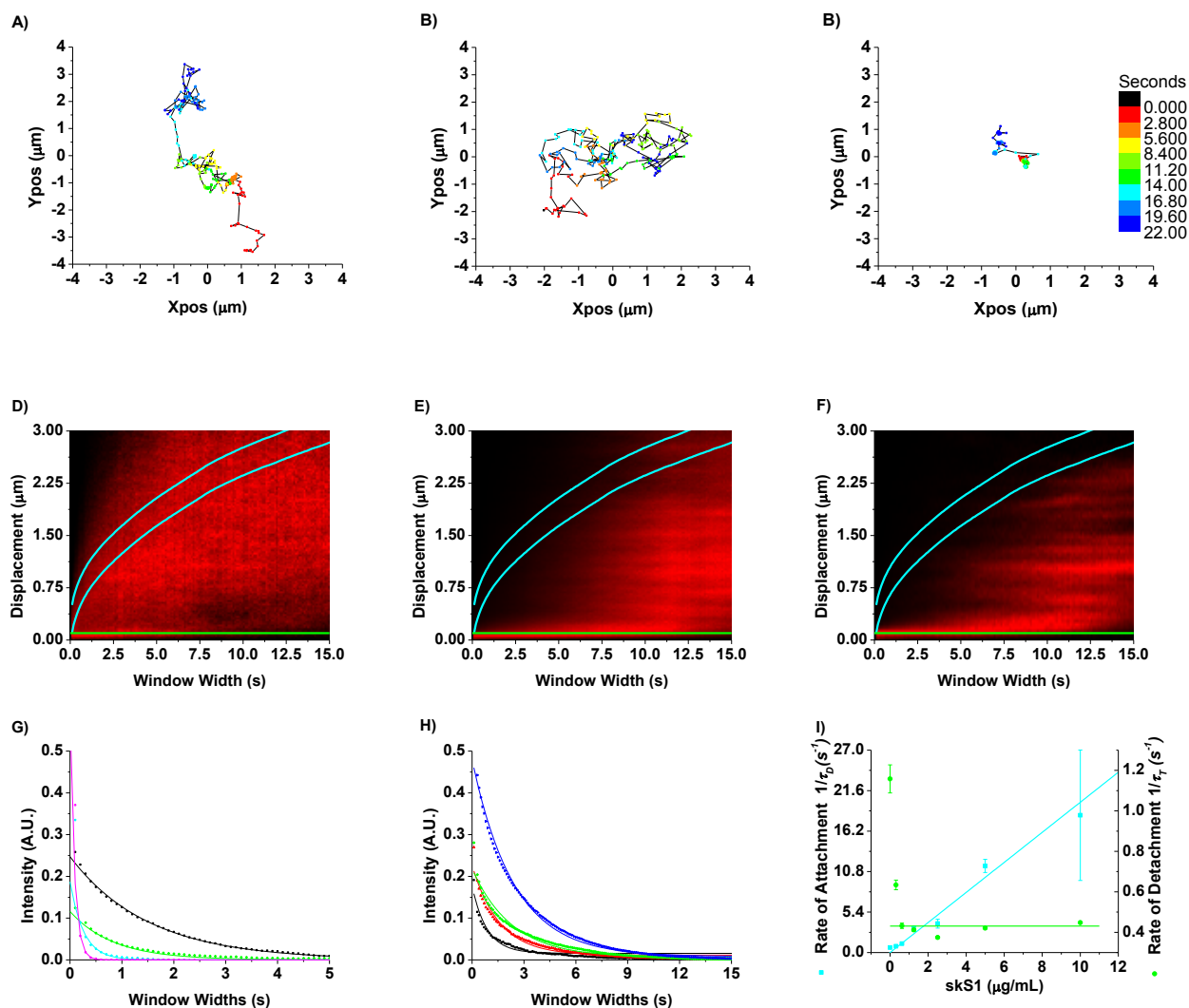


Figure 4 - Using NADA to Determine the Kinetics of Experimentally Measured Single Actin Fragments Binding to (k_{DT}) and Dissociating from (k_{TD}) skS1 on a Coverslip Surface.

(A-C) Sample trajectories of single actin fragments, A, binding to and dissociating from skS1 (M) on a coverslip surface obtained by incubating the surface for two minutes with (A) 0.00, (B) 1.25, and (C) 5.00 $\mu\text{g ml}^{-1}$ skS1. The bound actin fragment (A-M) is observed as a tethered state and the dissociated actin fragment (A + M) is observed as a diffusive state. (D-F) NADA histograms obtained from trajectories acquired under conditions of (A-C), respectively. The cyan lines are the limits for the selection of the diffusive population. Green lines are the center of the tethered population. (G) Plot of intensities for the diffusing population (all points between cyan lines) from NADA histogram obtained at 0 (black), 0.625 (green), 2.50 (cyan), and 5.0 (magenta) $\mu\text{g ml}^{-1}$ skS1. (H) Plot of intensities for the tethered population (all points below green line) from NADA histograms obtained at 0 (black), 0.3125 (red), 0.625 (green), 2.50 (cyan), and 1.25 (yellow) $\mu\text{g ml}^{-1}$ skS1. (I) Inverse lifetimes, $1/\tau_D$ (blue) and $1/\tau_T$ (green), obtained in G and H are plotted for different skS1 incubation concentrations. The average bound lifetime, τ_T , measured at $[\text{skS1}] > 0.3 \mu\text{g ml}^{-1}$ is 2.5 s (green line). A linear fit to the $1/\tau_D$ data (cyan line), gave a slope (k_{DT}) of $2.0 \pm 0.1 \text{ s}^{-1} \cdot (\mu\text{g ml}^{-1})^{-1}$.

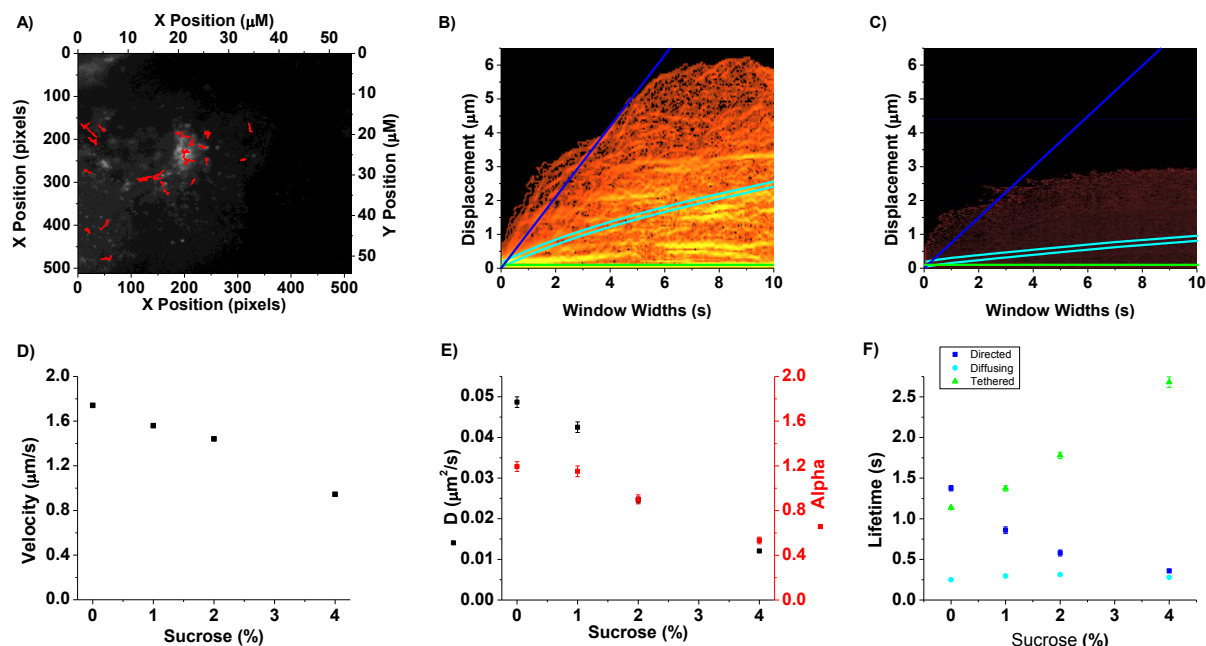


Figure 5 - Using NADA to Determine the Kinetics of Experimentally Measured Glut4-GFP Fusion Vesicle Transport within 3T3-L1 Cells.

(A) A TIRF image of 3T3-L1 fibroblasts with overlaid Glut4-GFP-vesicle trajectories (red lines). Trajectories were obtained from cells in media containing 0, 1, 2, and 4 % sucrose. (B-C) NADA histograms obtained from vesicle trajectories obtained with 0% (B) and 4% (C) added sucrose. NADA histogram images were edge enhanced with Sobel filtering and over-saturated to maximize image gradients. The blue lines are the lower limits of the directed population; the cyan lines bracket the center of the diffusing population; and the green lines are the center of the tethered population, as defined in the text. (D) Average velocities were determined from a linear fit to the displacements above the blue line for each NADA histogram. Velocities are plotted for trajectories obtained at 0, 1, 2, 3, and 4% sucrose. (E) D (black) and α (red) coefficients were determined by fitting the center of the diffusing population in each NADA histogram to the equation $d = \sqrt{4Dww^\alpha}$. These values are plotted for trajectories obtained at 0, 1, 2, 3, and 4% sucrose. (F) Lifetimes for the diffusing (cyan), tethered (green), and directed (blue) states were obtained from each NADA histogram and are plotted for trajectories obtained at 0, 1, 2, and 4 % sucrose.

References

1. Lu, H. P. Single-Molecule Enzymatic Dynamics. *Science (80-.)*. **282**, 1877–1882 (1998).
2. Brown, a. Slow axonal transport: stop and go traffic in the axon. *Nat. Rev. Mol. Cell Biol.* **1**, 153–6 (2000).
3. SAXTON, M. Single-particle tracking: models of directed transport. *Biophys. J.* **67**, 2110–2119 (1994).
4. DIETRICH, C. Relationship of Lipid Rafts to Transient Confinement Zones Detected by Single Particle Tracking. *Biophys. J.* **82**, 274–284 (2002).
5. Manzo, C. & Finzi, L. Quantitative analysis of DNA-looping kinetics from tethered particle motion experiments. *Methods Enzymol.* **475**, 199–220 (2010).
6. Das, R., Cairo, C. W. & Coombs, D. A hidden Markov model for single particle tracks quantifies dynamic interactions between LFA-1 and the actin cytoskeleton. *PLoS Comput. Biol.* **5**, e1000556 (2009).
7. Patlak, J. B. B. Measuring kinetics of complex single ion channel data using mean-variance histograms. *Biophys. J.* **65**, 29–42 (1993).
8. Lizunov, V. a, Matsumoto, H., Zimmerberg, J., Cushman, S. W. & Frolov, V. a. Insulin stimulates the halting, tethering, and fusion of mobile GLUT4 vesicles in rat adipose cells. *J. Cell Biol.* **169**, 481–9 (2005).

9. Frost, S. C. & Lane, M. D. Evidence for the involvement of vicinal sulfhydryl groups in insulin-activated hexose transport by 3T3-L1 adipocytes. *J. Biol. Chem.* **260**, 2646–52 (1985).
10. Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. *Nat. Methods* **9**, 671–675 (2012).
11. Margossian, S. S. & Lowey, S. Preparation of myosin and its subfragments from rabbit skeletal muscle. *Methods Enzymol.* **85 Pt B**, 55–71 (1982).
12. Weeds, A. G. & Taylor, R. S. Separation of subfragment-1 isoenzymes from rabbit skeletal muscle myosin. *Nature* **257**, 54–56 (1975).
13. Hooft, A. M., Maki, E. J., Cox, K. K. & Baker, J. E. An accelerated state of myosin-based actin motility. *Biochemistry* **46**, 3513–20 (2007).
14. Hancock, W. O. Processivity of the Motor Protein Kinesin Requires Two Heads. *J. Cell Biol.* **140**, 1395–1405 (1998).
15. Mizuno, N. *et al.* Dynein and kinesin share an overlapping microtubule-binding site. *EMBO J.* **23**, 2459–67 (2004).
16. Russ, J. C. . in *Image Process. Handbook, Fourth Ed.* 475–526 (CRC Press, 2002). doi:10.1201/9780203881095.ch8
17. Harris, D. E. & Warshaw, D. M. Smooth and skeletal muscle myosin both exhibit low duty cycles at zero load in vitro. *J. Biol. Chem.* **268**, 14764–8 (1993).

18. Howard, J. *Mechanics of Motor Proteins & the Cytoskeleton. Mech. Mot. Proteins Cytoskelet.* (Sinauer Associates, 2001).

Chapter 3: Non-Averaging Displacement Analysis of Dynamics of Regulated Thin Filament Motility

Michael Sean Carter, Sam Dugan, and Josh Baker

Department of Biochemistry and Molecular Biology, University of Nevada School of Medicine, Reno Nevada 89557

Abstract

Contraction of cardiac and skeletal muscle is regulated by the biochemical state of the tropomyosin-troponin complex (TmTn) on the thin filament. The TmTn complex functions as a switch that conditionally activates and inactivates the thin filament. In the absence of Ca^{2+} , the TmTn complex sterically blocks myosin strong binding sites, inactivating muscle mechanics. Upon addition of Ca^{2+} and myosin strong binding TmTn is displaced away from myosin binding sites on actin, activating the actin-myosin ATPase reaction and muscle contraction. It is well established that regulated thin filament (RTF) *in vitro* motility exhibits reduced average sliding velocities at sub-maximal activating conditions; however the mechanisms underlying these partially-activated RTF velocities are unclear. In this study we performed *in vitro* motility experiments in which fluorescent RTFs were imaged as they moved over a bed of skeletal muscle myosin. The resulting trajectories were analyzed using a Non-Averaging Displacement Analysis (NADA). We showed that with a decrease in $[\text{Ca}^{2+}]$ submaximal velocities result from a decrease in the fraction of time RTFs spend in a directed state of motion. In contrast, with a decrease in ATP, submaximal velocities result from a decrease in the average sliding velocity. These results imply that sub-maximal activation of RTFs result from modulating the dynamic switching of RTFs between

activated and inactivated states rather than a simple decrease in the level of RTF activation.

Introduction

The contraction of cardiac and skeletal muscle is regulated at the level of the thin filament. This regulation is mediated by the biochemical state of the tropomyosin-troponin complex (TmTn) on the thin filament. When the filament is in inactivating conditions such as the absence of free Ca^{2+} , TnI tightly binds Tm to the thin filament in a conformation that sterically blocks myosin strong binding, inactivating the thin filament and preventing contraction ¹.

The activation of the thin filament can occur through either calcium or myosin binding. During Ca^{2+} activation, TnC binds Ca^{2+} , which weakens the interaction between TnC and TnI, disrupting the Tm and TnI interaction and allowing Tm to be displaced from myosin strong binding sites ²⁻⁴. During myosin-based activation, a single myosin strongly binds an inactive region on the thin filament. The bound head induces a localized displacement of Tm opening several adjacent myosin binding sites to which additional myosin heads can bind. It has been proposed that the thin filament is activated as long as at least one myosin head is bound to it ². The thin filament returns to an inactivated state when no myosin heads are bound.

There are two different mechanisms for the velocities of partially activated regulated thin filaments, RTFs, measured in an *in vitro motility* assay. In one model (here referred to as the variable velocity model) partially-activated RTF velocities result from a decrease in the instantaneous velocity according to the equation

$$V = a \cdot V_{\max} \quad \text{Eq. 1}$$

where V is the average velocity, a is the fractional level of activation, and V_{\max} is the maximal velocity. In another model (here referred to as the dynamic switching model) partially-activated RTFs spend time in both an inactive state with a lifetime of τ_T and a fully active state with a lifetime of τ_D where calcium- and myosin-dependent activation modulate the time spent in these states according to the equation

$$V = f_T \cdot 0 + f_D \cdot V_{\max} \quad \text{Eq. 2}$$

where f_T is the fraction of time the RTF is in a tethered non-motile state defined as $f_T = \tau_T / (\tau_T + \tau_D)$, f_D is the fraction of time the RTF is in a directed motile state defined as $f_D = \tau_D / (\tau_T + \tau_D)$. Data from Marston and colleagues support the latter model, showing that in an *in vitro* motility assay the regulation of individual RTFs occurs in an all or none process where the entire filament is activated or deactivated as a single binary unit^{6,7}. Further laser trap data suggests that the normal TmTn inhibition reduces the probability of myosin's first encounter with actin 100-fold but after the first myosin head binding, the thin filament is activated 10-fold. This additional activation is most likely the result of the activation of adjacent regulatory units⁸.

In this study we test these hypotheses (Eqs. 1 and 2) using RTF *in vitro* motility performed under partially activating conditions. The RTF *in vitro* motility assay has been

used to study the biophysics of thin filament activation^{9,10}. This system approximates the regulation present in striated muscle by combining purified actin filaments, tropomyosin (Tm) and troponin (Tn) to reconstitute RTF providing a simplified mechanical system with which myosin cross bridge cycling/muscle contraction can be quantified through the motion of RTFs. In this system RTFs exhibit the same regulatory properties found in muscle particularly the inhibition of strong myosin binding and the strong cooperativity of activation.

The resulting motility trajectories were analyzed by Non-Averaging Displacement Analysis (NADA, Chapter 2) to characterize the kinetics underlying partially activated RTFs. The advantage of using NADA is that it creates a graphical representation of the individual displacements present in *in vitro* motility trajectories. This process sorts different states of motion into populations of intensities in a model-independent manner. The NADA histogram allows the user to quickly grasp all of the motion present within the system in a manner which is not possible by other methods. Further NADA can be used to measure the kinetics of transitions between states of motion identified in the NADA histogram.

We show the reduced RTF sliding velocities observed under partial activation of RTFs by both myosin and Ca^{2+} are produced by varying the fraction of time RTFs spend at V_{max} as described by Equation 2. Further we show that this mechanism is distinct from other mechanisms that generate reduced sliding velocities such as low [ATP] where the reduction in actin sliding velocities is generated by a reduction in V_{max} while the fraction of time moving remains constant.

Methods

Proteins

Skeletal muscle myosin was prepared from rabbit psoas muscle as previously described and stored in 50% glycerol at $-20\text{ }^{\circ}\text{C}$ ¹¹. Actin was purified from rabbit psoas and stored on ice at $4\text{ }^{\circ}\text{C}$ as previously described¹². Actin was labeled by mixing with equimolar concentration of either phalloidin-Alexa-488 for high frame rate imaging or phalloidin-tetramethylrhodamine (Invitrogen, Carlsbad, CA USA) for low frame rate imaging. Troponin and Tropomyosin (TmTn) was purified from skeletal muscle as previously described. Regulated thin filaments were reconstituted by incubating 20 nM actin with 200 nM TmTn for 20 minutes^{2,13,14}.

Speckled Actin

In order to maximize our positional resolution we refined the position of the RTFs used in the high frame rate motility assays by non-linear curve fitting the fluorescent intensity profile of the labeled RTFs with Gaussian curves in two dimensions. Because the fluorescent intensity profile of uniformly labeled actin is complex we created actin with small punctate labels separated by lengths of unlabeled actin. These punctate labels, speckles, simplify the intensity profile of the actin filament dividing the fluorescence into several sub-resolved regions which due to their small size and separation produce an intensity profile that effectively approximates a 2D Gaussian. The manipulation of the actins fluorescence significantly simplifies the refinement algorithm removing the need to determine the orientation of the filament and iteratively fit the actin filament through perpendicular slices as has been done previously¹⁵.

The speckled actin was prepared as follows: First regulated Alexa-488 labeled actin was fragmented using a Sonic Dismembrator Model 100 (Fisher Scientific, Hampton, New Hampshire) producing sub-micrometer long fragments of labeled F-actin. These actin fragments were flowed into the flow chamber as would be done in normal motility assays and allowed to bind to the myosin already present on the surface for two min. During the incubation, concentrated (2 μ M) unlabeled phalloidin stabilized F-actin was fragmented in the same manner as the labeled actin. The unlabeled F-Actin fragments were flowed into the chamber. Do to the high concentration, the unlabeled fragments annealed to the actin fragments attached to the surface. After a five min incubation, the flow chamber was washed two times with actin buffer to remove any un-annealed actin before proceeding with the rest of the assay. The construction of long speckled actin filaments was initially verified by using the same procedure described above using two different colors of labeled actin, which could be visualized on separate channels. These images were combined into a single image showing that the annealed filaments were continuous and several μ m in length (data not shown).

Buffers

The Myosin buffer contained 300 mM KCl, 25 mM imidazole, 1 mM EGTA, 4 mM MgCl_2 and 10 mM DTT. The Actin buffer contained 50 mM KCl, 50 mM imidazole, 2 mM EGTA, 8 mM MgCl_2 , and 10 mM DTT. Motility buffer contained 100 nM Tm-Tn , 50 mM KCl, 50 mM imidazole, 2 mM EGTA, 8 mM MgCl_2 , 10 mM DTT, and 0.5% methylcellulose. The concentration of ATP and Ca^{2+} was varied as specified. The pH of the motility buffer was adjusted to 7.4 using NaOH. All solutions except myosin buffer

contained oxygen scavengers (6 mg/ml glucose, 0.03 mg/ml glucose oxidase, 0.05 mg/ml catalase), which was added immediately prior to imaging to minimize pH changes due to the oxidation of glucose to gluconic acid ¹⁶. Calcium was buffered with EGTA, and even at pCa 3 calcium did not significantly (<4%) change the total MgATP concentration, as determined through a ligand binding algorithm ¹⁷.

In Vitro Motility Assay

In vitro motility assays were used to measure the level of activity of the fluorescently labeled regulated thin filaments. These assays were conducted by using flow chambers to incubate nitrocellulose-coated coverslips with small volumes (~ 80 μ L) of solution in a specific order: 100 μ g/mL myosin diluted in myosin buffer, 500 μ g/mL BSA diluted actin buffer (blocking solution), 20 nM fluorescently labeled RTF in actin buffer, actin buffer wash, and the motility buffer. The concentration of Ca^{2+} and ATP in the motility buffer was specific to each condition and is noted in the results section.

Analysis of *In Vitro* Motility Movies to Obtain Trajectories:

All imaging was done using a Nikon TE2000 epifluorescence microscope with a 100x Nikon Plain Apo TIRF objective with a NA of 1.45 using either a Roper Cascade 512B camera (Princeton Instruments, Trenton, NJ) in widefield epifluorescence mode for slow frame rate images or using a XR/turbo G intensified camera (Stanford Photonics Inc., Palo Alto, CA) in TIRF illumination mode for high speed imaging.

Low speed motility analysis.

Low speed motility images were analyzed using a custom centroid-based tracking algorithm that isolates regions of high intensity from the background within each frame of the microscope image by intensity thresholding. The level of the threshold

was determined for each movie such that the particles of interest were adequately represented without including excess noise from the background or joining of adjacent fluorescent regions. Any fluorescent region with an area less than a given constant was removed from the analysis and the centroid for each of the remaining regions was calculated as previously described¹⁸. Trajectories were created by linking the centroid positions through a simple displacement minimization algorithm. This linking algorithm calculates all of the possible ways of connecting the detected objects in one frame to the next frame and chooses linkages between objects that minimize the total displacement between frames. Trajectories with durations less than 2 s were excluded from the analysis to ensure that the motion of any particle contained enough sampled points to properly describe its behavior. Similarly trajectories that did not displace at least 1 μm while observed were also excluded from consideration to prevent the analysis of non-moving objects.

High Speed motility analysis

High speed motility images were analyzed in two passes. The first pass consisted of the same analysis used for the low speed images. The second pass refined the center positions of the fluorescent regions in each frame by fitting the intensity profiles to Gaussian curves in two dimensions. This additional refinement was possible because of the use of speckled actin. As discussed above the speckled actin has small punctate labels that have intensity profiles approximating Gaussian distributions allowing the center of these probes to be located in high precision using relatively simple non-linear curve fitting compared to existing methods to fit uniformly labeled filaments¹⁵.

Lifetime Analysis

In order to characterize the mechanical and kinetic properties of the different states of motion present within the *in vitro* motility trajectories obtained from partially active RTFs we used our Non-Averaged Displacement Analysis (NADA, chapter 2) to measure the lifetimes of the directed and tethered (non-motile populations of motion). The general theory of the NADA method is explained in depth in chapter 2. NADA is performed in a manner similar to calculating the standard mean squared displacement (MSD); where individual displacements are calculated from each point in the trajectory to a point an interval of time, referred to as a window width, later in the trajectory. NADA differs from the MSD analysis by not averaging the points at each window and instead uses a 2D histogram, referred to as an NADA histogram, to visualize the distribution of the displacements as a function of the window width. These individual displacements collect into regions of high intensity within the histogram according to the mechanical properties of the state of motion which generated them. For example the intensity representing tethered motion, is localized near the x axis because tethered particles do not increase their displacement as a function of time. Similarly the intensity from the directed state of motion appears as a diagonal region of high intensity with a slope equal to the velocity of the motion that generated it. (chapter 2) Since the probability of observing a single event of a fixed duration such as a molecule being in a defined state decreases exponentially as a function of the window width ¹⁹, the lifetime of each state of motion can be determined by measuring the sum of the intensities within the NADA histogram corresponding to the directed and tethered population as a function of window width. These intensities were fit to an exponential whose time constant equals the lifetime of the state of motion being measured. (chapter 2).

Analysis of NADA Histograms to Obtain V_{\max}

To quantify the range of displacements in the NADA histogram, a line was drawn parallel to the maximum observed displacement at each window width for the region of the NADA histogram where the maximum displacements at each window width followed a linear increase with increasing window widths. (i.e. where window width $< \tau_D$) The slope of this line is the maximum velocity of the directed population (V_{\max}).

Generation of Simulated Trajectories

Trajectories for the variable velocity model (Eq. 1) were simulated based on a single directed state of motion (Fig. 2A), in which a particle moved at velocities normally distributed about an average value $V = a \cdot V_{\max}$.

Trajectories for the dynamic switching model (Eq. 2) were simulated based on a biochemical scheme (Fig. 2A) in which a particle switches between a directed state (characterized by directed movement at an average constant velocity, V_{\max}) with a lifetime τ_D , and a tethered state (characterized by diffusion about a fixed point with no net velocity) with a lifetime τ_T (chapter 2).

Results

Average Velocities of Partially-Activated RTF Motility

Numerous studies suggest that RTFs become activated by strongly-bound myosin at low [ATP] even in the absence of Ca^{2+} ⁴. To measure myosin activation of RTFs, we use *in vitro* motility to measure RTF velocities, V , at different ATP concentrations. In Fig. 1A, V is plotted at different ATP concentrations. At high [ATP]

(~1 mM) and no added Ca^{2+} , V was approximately zero, consistent with low activation at low Ca^{2+} . When we decreased [ATP] from 1 mM to 40 μM , V increased from 0.10 to 0.55 $\mu\text{m/s}$, consistent with myosin-activation of the RTF^{2,5,7}. The peak velocity observed at 40 μM ATP is the point at which RTFs are fully activated by myosin. When we decreased [ATP] below 40 μM ATP, V decreased due to the ATP-dependence of fully-activated RTF velocities².

RTFs are also activated by calcium. Using *in vitro* motility we measured V at 1 mM ATP and different pCa. Consistent with previous studies, Fig. 1B shows that the calcium-dependence of V is sigmoidal. We fit these data to the Hill equation (red line) giving an upper plateau of $1.51 \pm 0.05 \mu\text{m s}^{-1}$ (SE), a pCa_{50} (the pCa at half maximal activation) of 5.94 ± 0.04 (SE), and a lower plateau of 0.10 ± 0.02 (SE). These results were similar to previously published data².

Non-Averaging Displacement Analysis

To determine whether the mechanism of partial activation of RTFs by myosin and calcium follow the model described by Eq. 1 or the model described by Eq. 2, we analyzed the data in Figs. 1 and 2 using a non-average displacement analysis (NADA). Simulated trajectories based on the two schemes in Fig. 2, demonstrates that NADA can distinguish between these two models (Fig. 2).

Myosin-induced Activation of RTF Motility in the Absence of Calcium

To determine the mechanism of myosin-induced activation of RTFs, we used NADA to analyze the RTF trajectories obtained in Fig. 1A. Figure 3 A-D are 20 s sample trajectories obtained by centroid tracking of RTF motility at 15, 30, 50, and 75 μM ATP respectively. These trajectories are color-coded to represent the time elapsed. The total

displacement of each trajectory increased with an increase in ATP from 15 (A) to 30 μM (B), consistent with the observed increase in V at the higher [ATP] (Figure 1A). With a further increase in [ATP] from 50 (C) to 75 μM (D), the total displacement decreased consistent with the observed decrease in V (Figure 1A).

Figures 3 E-H show NADA histograms for RTF trajectories obtained at 15 (E), 30 (F), 50 (G), and 75 (H) μM ATP. The NADA histogram obtained at 15 μM ATP (Fig. 3E) resembles the NADA histogram obtained from simulated trajectories based on the variable velocity model (Eq. 1, Fig. 2B). This is consistent with the proposal that RTFs are maximally activated below 30 μM ATP with an average velocity that varies with [ATP]. NADA histograms obtained at ATP concentrations of 30 μM and above (Figs. 3F – H) resemble NADA histograms obtained from simulated trajectories based on a switching velocity model (Eq. 2, Fig. 2A). According to a switching model, the observed decrease in V with increasing [ATP] above 30 μM results from a shift in the distribution of states (Fig. 2A, Eq. 2) from a directed population that moves at a fixed velocity V_{max} (Fig. 3E-H, blue lines) toward a non-moving population (Fig. 3E-H, green lines).

Also consistent with a switching model (Eq. 2), V_{max} determined from NADA histograms as described above at 30, 50 and 75 μM ATP increased with increasing ATP (0.90, 1.02, and 1.14 $\mu\text{m s}^{-1}$ respectively). According to Eq. 2, this means that the fraction of time, f_{D} , a filament spends moving at V_{max} must decrease with increasing [ATP]. To further test this hypothesis, we used NADA to determine the lifetimes of the directed (moving) and tethered states.

Lifetime Analysis of Myosin-induced Activation of RTF Motility in the Absence of Calcium

From the NADA histograms in Figure 3, the lifetimes of the directed and tethered populations were determined as described in the methods. The directed populations (the areas above the blue lines in Figs. 3E – H) were defined as the regions of the NADA histogram where the resolvable trajectories in the histogram had y-intercepts of zero. The tethered population was defined as the region of the NADA histogram below the major horizontal population parallel to the x-axis. Figures 4 A and B contain examples of the sum of intensities for the directed (Figure 4A) and tethered (Figure 4B) populations at each window width obtained at 40 (magenta), 50 (green) and 75 μM ATP (navy). The sum of intensities for the directed and tethered populations at different window widths were fit to single exponentials (matching colored lines) and the resulting lifetimes are plotted in Figure 4C.

Figure 4C shows that when we increased $[\text{ATP}]$ above 30 μM , the lifetime of the directed population, τ_D , decreased and the lifetime, τ_T , of the tethered population increased. Figure 4D clearly shows that the fraction of time spent in the directed population, $f_D = (\tau_D / (\tau_D + \tau_T))$, decreased when we increased $[\text{ATP}]$ above 30 μM , consistent with the switching velocity model (Eq. 1).

Our results show that the NADA histogram (Fig. 3E) obtained at 15 μM ATP is qualitatively similar to NADA histograms obtained from simulated variable velocity trajectories (Fig. 2F), and that V_{max} (yellow line) is less than that observed at 30 μM ATP. Both observations are consistent with a variable switching model (Eq. 1) for RTF velocities that are maximally activated by myosin (Fig. 1) ².

In contrast, our results indicate a switching model for RTF velocities partially activated by myosin ($[ATP] \geq 30 \mu\text{M}$). Specifically, NADA histograms obtained at or above $30 \mu\text{M}$ ATP (Fig. 3F-H) are qualitatively similar to those obtained from simulated switching trajectories (Fig. 2 E) and consistent with Eq. 2 fraction of time in the directed state, f_D , decreases (Fig. 4D) with decreasing V (Fig. 1).

The difference between the two mechanisms is evident by comparing the NADA histograms for RTF sliding obtained at 15 and $75 \mu\text{M}$ ATP. Even though the velocities obtained at these two ATP concentrations are similar (0.270 and $0.290 \mu\text{m/s}$ respectively, Fig. 1), the NADA histograms are fundamentally different.

In summary, partial activation of RTF velocities by myosin follow a switching model. Next, we determine if this is also true for partial activation of RTFs by calcium.

Calcium-Activation of Regulated Thin Filament Motility

To determine which model (variable or switching) most accurately describes calcium-based activation of RTFs, we conducted *in vitro motility* using RTFs at 1 mM ATP and at different Ca^{+2} concentrations. Figures 5 A-D are sample trajectories obtained by centroid tracking of RTF motility at $\text{pCa } 3$ (A), 5.6 (B), 6 (C), and 6.6 (D). These trajectories are color-coded to represent the time elapsed, and the calcium-dependence of the average velocities in these representative traces is consistent with the calcium-dependence of the velocities in Fig. 1B.

Figure 5 E – H are NADA histograms for RTF trajectories obtained at $\text{pCa } 3$ (E), 5.6 (F), 6 (G), and 6.6 (H). These NADA histograms resemble those obtained from simulations of the switching model (Eq. 2; Fig. 2F) for velocity. V_{max} (the slope of a line tangent to the histogram data that goes through the origin) decreases only 42% over the

range of complete inactivation (from pCa 3 and 6.6), whereas V decreases 92% over this same calcium range. This implies a hybrid between the switching and variable models, in which the directed state has a calcium-dependent V_{\max} and the fraction of time spent in the directed state is also calcium dependent. We determined the calcium-dependence of f_D from the NADA histogram.

Lifetime Analysis of Calcium-Activation of Regulated Thin Filament Motility

Using the NADA histograms in Figure 5, the lifetimes of the directed and tethered populations were determined as described in the methods. The directed population was defined as the region of the NADA histogram (Fig. 5 E – H) where the resolvable trajectories in the histogram had y-intercepts of zero (intensities above the blue line, Fig. 5 E - H), and the tethered population was defined as the region of the NADA histogram that includes the major horizontal populations running parallel to the x-axis (intensities below the green line, Fig. 5 E - H). Figure 6 contains examples of the summed intensities for the directed (A) and tethered (B) populations at each window width for experiments performed at pCa 3 (black), 5.6 (red), 6 (blue) and 6.6 (magenta). The data in Figure 6 was fit to single exponentials (matching colored lines) and the resulting lifetimes were plotted in Figure 6C. Figure 6C shows that the lifetime of the tethered population increases with increasing pCa. The saturation of tethered lifetimes at high pCa presumably results from an artifact of the analysis. Specifically, the relatively low tethered lifetime for a non-moving filament at pCa 10 could result from the increased diffusive behavior of RTFs present at high pCa.

Using the data from Figure 6C we calculated the fraction of time, f_D , a filament is in the directed state, which is plotted in Figure 6D. The resulting data was fit to the hill

equation (red line) to obtain a pCa_{50} of 5.98 ± 0.01 (SE), which is comparable to the pCa_{50} of 5.94 ± 0.04 (SE) measured in Fig. 1B. These results support the hypothesis that the calcium-dependence of the average velocity (Fig. 1B) is influenced by the calcium-dependence of the fraction of time RTF spend in a directed state (Fig. 6D).

The V_{max} values from the NADA histograms in Figs. 5 E - F suggest that the V_{max} of RTF motility is calcium-dependent. However, it is possible that this observation results from undersampling RTF motion. Specifically, if the image exposure time (minimum window width) is greater than the lifetime, T_D , of the directed population, then the apparent per-frame distance traveled will be influenced by both the tethered (non-moving) and directed (moving) states, and the apparent V_{max} will be less than the V_{max} of a purely directed state. Consistent with this undersampling argument, the apparent decrease in V_{max} with increasing pCa (Figs. 5 E – H) corresponds to a decrease in τ_D with increasing pCa (Fig. 6C). To test whether or not the observed decrease in V_{max} with increasing pCa (Fig. 5) results from undersampling, the experiments in Figures 5 and 6 were repeated at 400 fps using an intensified CCD camera.

Calcium Activation of Regulated Thin Filament Motility at 400 Frames per Second

Figure 7 A - D are sample trajectories of RTF motility measured at pCa 3(A), 5.6 (B), 6.2(C) 6.8(D). The observed decrease in trajectory lengths (each measured over the same period of time) with increasing pCa is consistent with the decrease in velocity observed with increasing pCa (Fig. 1B). Figures 7 E- F are the corresponding NADA histograms created from trajectories obtained at pCa 3 (E), 5.6 (F), 6.2 (G), 6.8 (H). These histograms show similar intensity distributions to that of the NADA histogram obtained from trajectories simulated using a switching model (Eq. 2, Fig. 2F). The

observed calcium-independence of V_{\max} obtained at 400 fps (Fig. 7 E – H), suggests that the calcium-dependence of V_{\max} estimated at slower frame rates results from the undersampling of the directed population.

Lifetime Analysis of Calcium Activation of Regulated Thin Filament Motility from High Speed Camera Trajectories

The lifetimes of the directed and tethered populations were determined using the NADA histograms in Figure 7. As before, the directed population was defined (Fig. 7 E – H) by the intensities above the blue line (Fig. 7 E - H), and the tethered population by the major horizontal populations running parallel to the x-axis (intensities below the green line, Fig. 7 E - H). Figure 8 A and B contain the summed intensities for the directed (Figure 8A) and tethered (Figure 8B) populations at each window width obtained at pCa 3 (black), 5.2 (red), 5.6 (blue), 6.2 (magenta) and 6.8 (green). The summed intensities for the directed and tethered lifetimes at each condition were fit to double exponentials (lines) and the slow rate was plotted in Figure 8C, showing that the lifetime of the directed population decreases with increasing pCa, similar to the trends observed in Fig. 6C, supporting the conclusion that the velocities of RTF observed at partial calcium activation result from modulation of the directed lifetime as a function of the level of activation of the RTF.

Discussion

As previously characterized^{2,3,5} RTF motility exhibits decreased actin sliding velocities under partially activating conditions; however, the mechanism responsible for the decrease in RTF sliding velocities (V) is not well understood. In this chapter we tested two hypotheses (Eqs. 1 and 2) that account for the decrease in actin sliding velocities at partially activating conditions. The first hypothesis (Eq. 1, variable velocity model) is that the decrease in V with decreasing calcium results from a decrease in the maximal velocity. The second hypothesis (Eq. 2, dynamic switching model) is that partially-activated RTFs dynamically switch between two different states of motion: an inactive non-directed or tethered state and an active, directed state, and that the decrease in V with decreasing calcium results from a decrease in the fraction of time, f_D , RTFs are in the directed state.

To test which of these two proposed mechanisms is responsible for the observed decrease in actin sliding velocities under partially activating conditions, we conducted *in vitro* motility using RTFs under partially activating conditions. The motion of the RTFs at partially activating conditions was measured using a non-averaging displacement analysis (NADA, chapter 2). This method creates a graphical representation of the individual displacements present in *in vitro* motility trajectories, segregating displacements into different states of motion. A NADA analysis of simulations based on the models in Eqs. 1 and 2 demonstrate our ability to use NADA to differentiate between the two models.

Our myosin-based activation studies (Fig. 3) show that NADA histograms of fully-activated RTF velocities obtained at low [ATP] ($< 30 \mu\text{M}$, Fig. 3E) are consistent with the variable velocity model (Eq. 1). However, NADA histograms of partially activated RTF at higher [ATP] ($> 30 \mu\text{M}$, Figs. 3F – 3H) are consistent with the dynamic switching model (Eq. 2). This transition from variable velocity to dynamic switching with increasing [ATP] results from a decrease in duty ratio corresponding to both increased [ATP] and decreased activation of the RTF. The decrease in V observed when [ATP] is increased above $30 \mu\text{M}$ ATP appears to follow the switching mechanism in Eq. 2. The lifetime analysis of the NADA histograms for these experiments (Figure 4) show that the fraction of time spent in the directed population decreases with increasing [ATP], whereas the maximum velocity appears to increase. These findings indicate that the observed decrease in the average velocity, V , when [ATP] is increased above $30 \mu\text{M}$ (Fig. 1A) results from a decrease in the fraction of time spent in the directed population and not from a decrease in the maximum velocity.

In the Ca^{2+} -based activation experiments (Fig. 5) we obtained NADA histograms from RTF trajectories obtained at each Ca^{2+} concentration. These histograms resemble those obtained from simulations of the switching model (Eq. 2; Fig. 2F). From a lifetime analysis of the directed and tethered states of motion in these histograms we measured a pCa-dependent decrease in the fraction of time spent in the directed state (Fig. 6D) that correlates with the observed decrease in V over the same range of pCa (Fig. 1B). These results suggest that the decrease in RTF sliding velocities at partial activation results from a decrease in the fraction of time RTFs spend in a directed state. From these histograms, it appears that V_{max} decreases by 42% over the same pCa range,

however, we suggest that this is an artifact of sampling. Specifically, in these experiments, as the lifetime of the directed population decreases and approaches the camera exposure time, the non-moving population contributes to and decreases the measured maximum velocity.

To test whether or not the above decrease in V_{\max} resulted from a sampling artifact, the above Ca^{2+} -based activation experiments were repeated at a camera frame rate of 400 fps. Like the NADA histograms obtained at slower frame rates, these experiments showed that the NADA histograms are consistent with those obtained from switching model simulations. Similarly a decrease in the fraction of time directed with increasing pCa was also observed for the high speed pCa experiments (Fig. 8 C and D). However the high frame rate NADA histograms demonstrate a calcium-independence of V_{\max} (Fig. 7 E – H). These results suggest that the calcium-dependence of V_{\max} observed at lower frame rates was the result of the sampling effects discussed above. As a result we conclude that the decreased actin sliding velocities observed at partial activation of RTF by Ca^{+2} is the result of a decrease in the fraction of the time the filaments spend in the directed state with increasing levels of inactivation as described by the dynamic switching model (Eq. 2).

Currently there are many competing models of thin filament activation. According to the Geeves three state model^{20,21} RTF activation involves three distinct structural states through which the system progresses. Initially, in the absence of free Ca^{2+} the thin filament is in the blocked state, TnI binds strongly to actin, holding Tm in a blocked position preventing myosin strong binding. The binding of Ca^{2+} by TnC transitions the thin filament to the ‘closed’ state in which the interaction between TnI and Tm is

comparatively weak. The decreased strength of the Tm and Tn interaction allows Tm to have more freedom of “position” on the thin filament than in the blocked state. The increased Tm flexibility enables weak interactions between myosin and the thin filament but still prevents myosin strong binding. While the thin filament is in the closed position, myosin heads can on rare occasion strongly bind to the thin filament. On myosin binding Tm is further displaced, locally transitioning the thin filament from the blocked to open state. This transition allows additional myosin heads to bind and open more of the thin filament. This ‘opening’ process continues to feed forward until the maximal head cycling and ATPase activities are achieved at which point the filament is fully active.

According to the Hill two-state equilibrium model²², TmTn and seven actin monomers form a single regulatory unit, which exists in two states: an inactive state with low myosin binding affinity and an activated state with high myosin binding affinity. Calcium binding to TnC shifts the population of the two states to the high myosin affinity state. Since the activation of a single regulatory unit results in multiple myosin binding sites becoming available, the process of activation is highly cooperative²³.

Recently Sich *et. al.* proposed a model of constitutive RTF activation²⁴ in which the RTF in inactivating conditions such as low Ca^{2+} , exhibit a low rate of myosin binding; on Ca^{+2} binding by TnI, the rate of myosin binding increases favoring myosin binding to the thin filament. When the first head binds to the thin filament, through increased myosin binding created by Ca^{+2} activation or through the inactivated binding rate, the thin filament is activated. Once activated the rate of myosin strong binding by subsequent myosin heads increases to its maximum rate. This level of maximal activation continues as long as at least one myosin head is strongly bound to the filament. When the thin

filament no longer has any heads bound, the filament inactivates and the rate of myosin strong binding reduces to the Ca^{+2} activated or inactivated binding rate depending on the Ca^{2+} occupancy of the TnI subunit ².

While all three models describe the state of regulated thin filaments under fully activating and inactivating conditions, neither the Hill nor Geeves models explicitly describe the mechanism underlying the decreased averaged sliding velocities observed at partially activating conditions in RTF *in vitro* motility. This lack of clarity particularly regarding the function of cooperativity with in these models complicates the interpretation of these experiments using the parameters (V_{directed} , T_{D} and T_{T}) measured by this study and as a result these models cannot be applied directly to the findings of this study without performing simulations to interpret these models in terms of the V_{directed} , T_{D} and T_{T} parameters.

The Sichi model in contrast describes different levels of activation through modulation of periods of directed motion with a lifetime of T_{D} and tethered motion with a lifetime of T_{T} . This model suggests that the decreased velocities observed at different levels of activation are generated by changes in T_{D} and T_{T} resulting in a decreased fraction of time in the directed state while V_{max} remains independent of the level of activation due to the high level of cooperativity that characterizes the model. Through the NADA analysis of the motion of RTFs at different levels of activation in this study we show that the activation dependent decrease in the fraction of time associated with a decrease in the observed velocity of the directed state of motion as predicted by the *Sichi et al.* model. Further we show that the mechanism for the decreased actin sliding velocities is distinct from other mechanisms which generate reduced sliding velocities

such as myosin ATPase. As a result we conclude that the mechanism which creates decreased velocities at partial activation by both Ca^{2+} and myosin is an activation dependent decrease in the fraction of time the RTF move in a directed state of motion.

Figures

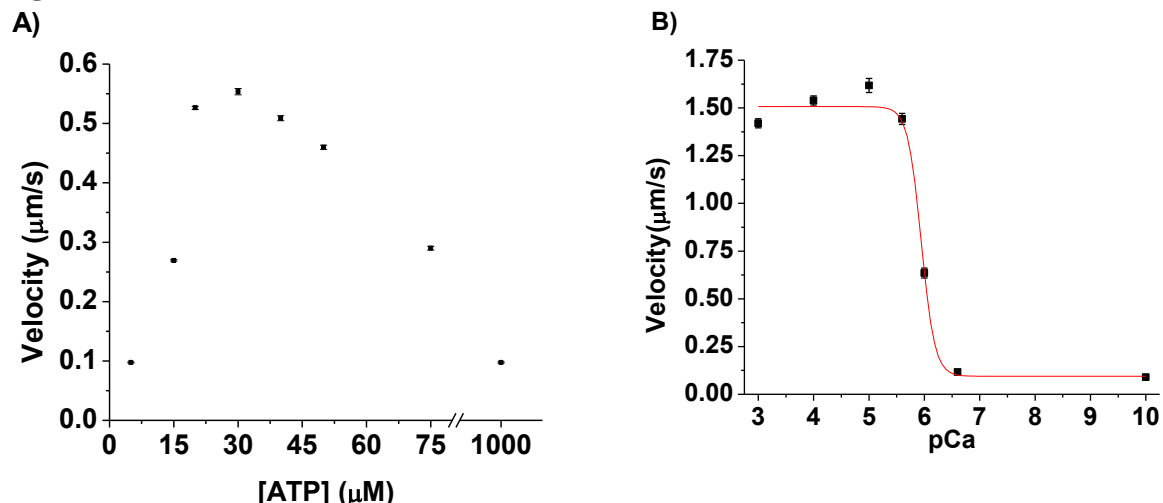


Figure 1 - Average Velocities of Partially-Activated RTF Motility.

Average velocities of myosin-activated RTF motility obtained from *in vitro* motility assays (A) different ATP concentrations in the absence of Ca^{2+} , and (B) at 1 mM ATP with different calcium concentrations. The data in B was fit to dose response curve resulting in an upper plateau of $1.51 \pm 0.05 \mu\text{m s}^{-1}$, a pCa_{50} of 5.94 ± 0.04 and a lower plateau of $0.095 \pm 0.016 \mu\text{m s}^{-1}$. (red line using the Hill equation)

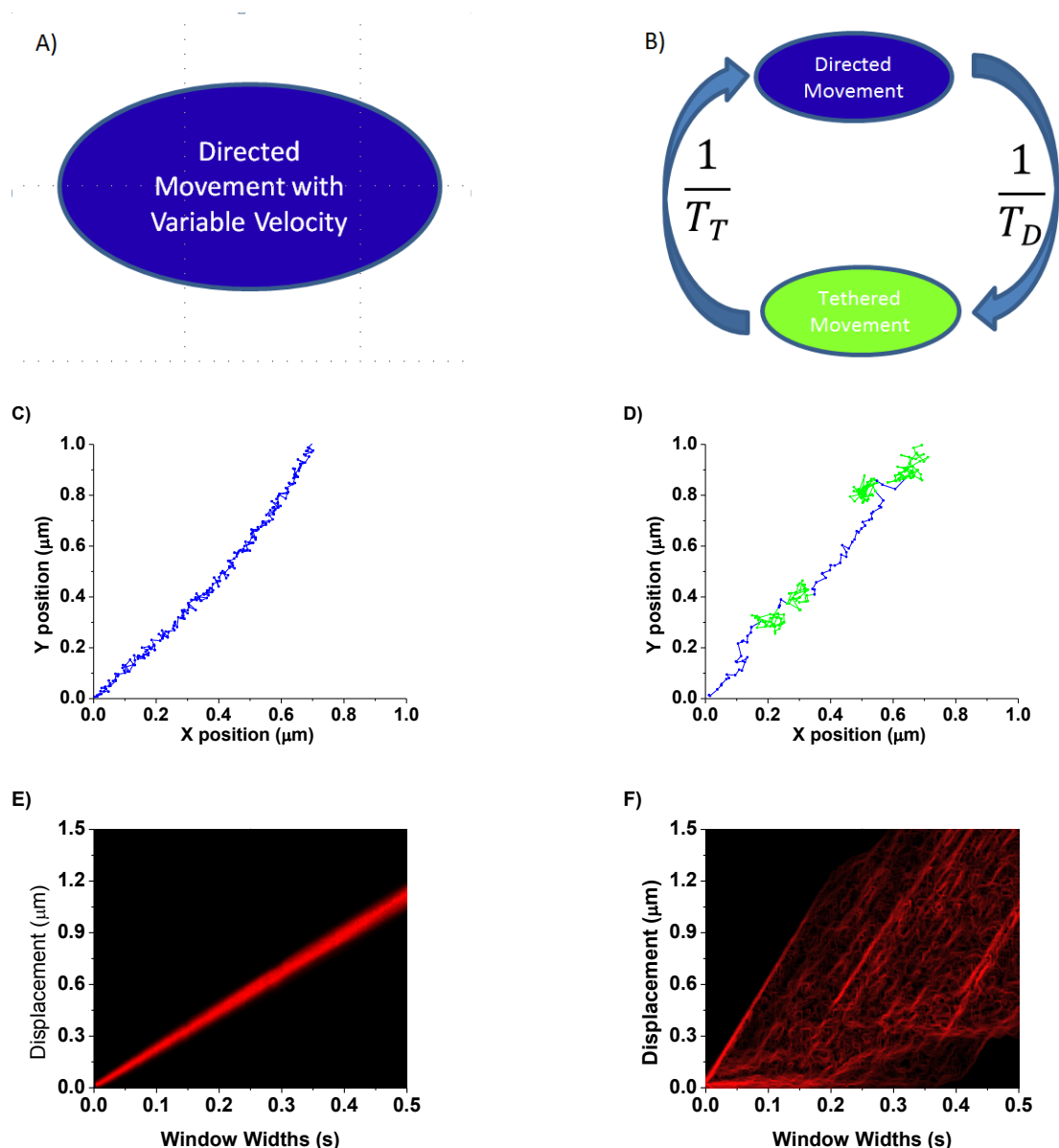


Figure 2 – Comparison of the Variable Velocity and Dynamic Switching Models for RTF Velocities at Partial Activation Analyzed by our Non-Averaging Displacement Analysis (A and C) For variable velocity model simulations, the particle exist in a single state with variable velocity, V , normally distributed at $2.23 \pm 1.30 \mu\text{m s}^{-1}$ (\pm standard deviation). (B and D) For the dynamic switching simulation, the particle dynamically switches between a directed state where the simulated particle moves with constant velocity ($6 \mu\text{m s}^{-1}$) with a lifetime ($T_D = 40$ ms, 16 frames) and a tethered state where the particle moves in a random direction about a fixed point with a lifetime ($T_T = 80$ ms, 32 frames) resulting in a V of $2.26 \mu\text{m s}^{-1}$. (C-D) $1 \mu\text{m} \times 1 \mu\text{m}$ samples of simulated trajectories for the dynamic switching (C) and the variable velocity (D) models with the particles state of motion is indicated by the color of the points in the trajectory; Blue indicates the directed state and green indicates the tethered state. (E and F) NADA histograms created from the simulated trajectories in C and D, respectively.

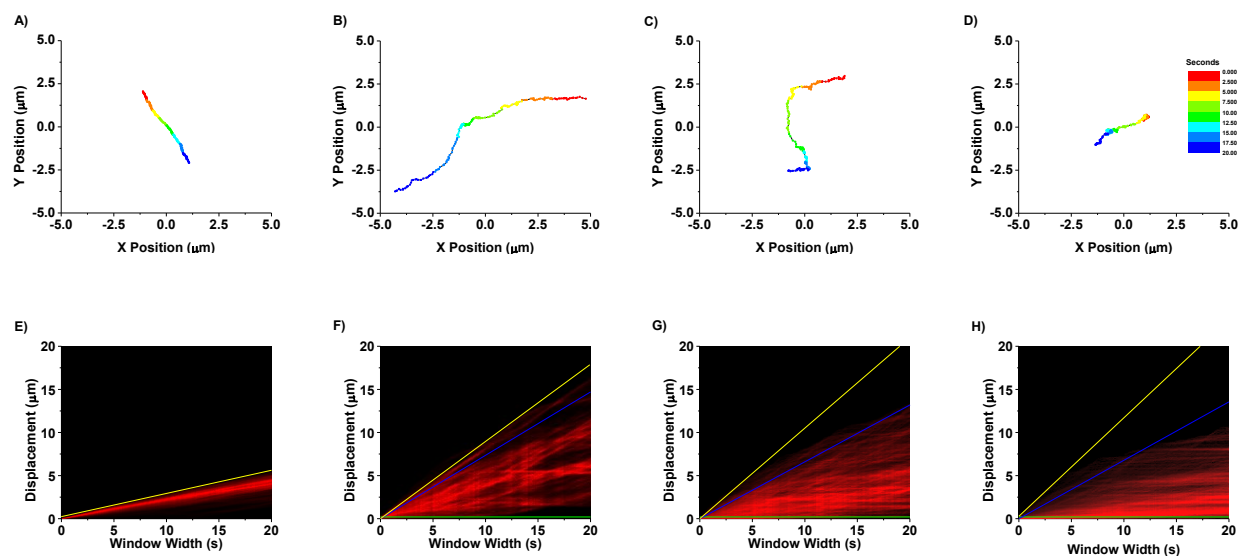


Figure 3 - Myosin-induced Activation of RTF Motility in the Absence of Calcium

(A-D) Examples of *in vitro* motility trajectories of myosin-activated RTF filaments in the absence of Ca^{2+} at (A) 15, (B) 30, (C) 50 and (D) 75 μM ATP. Trajectories were obtained by conducting *in vitro* motility using RTFs at $100 \mu\text{M ml}^{-1}$ myosin at the specified ATP concentration. All example trajectories are 20 s in duration (200 points), where time is represented by the color of the individual points (legend). (E-H) NADA histograms obtained from trajectories acquired under conditions of (A-D) respectively. The blue line is the lower limit of the directed population (the set of directed intensities with a zero intercept). The green line is the center of the tethered population which was chosen by observing RTF under conditions which do not support directed motility (ATP=0 μM). The yellow line is the V_{max} of the NADA histogram.

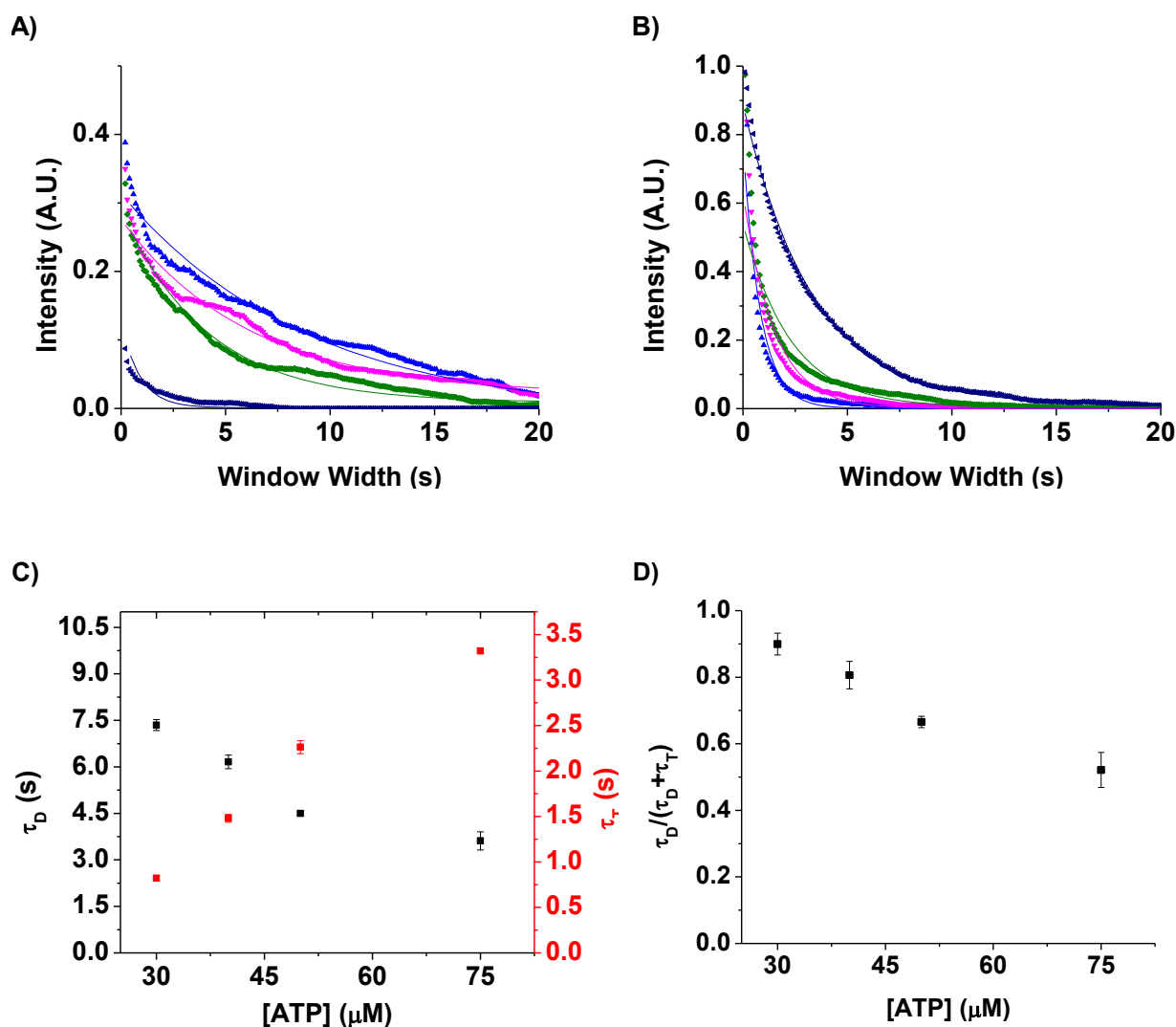


Figure 4 - Lifetime Analysis of Myosin-induced Activation of RTF Motility in the Absence of Calcium

(A - B) Plots of intensities for the directed population (A), the sum of intensities above the blue lines, and the tethered populations (B), the sum of intensities below the green line from the matching NADA histograms (Figure 3, bottom) obtained at 30 (blue), 40 (magenta), 50 (green), and 75 (navy) μ M ATP at each window width. Intensity decays in A and B were fitted to single exponentials (matching colored lines) to obtain population lifetimes. (C) Lifetimes of the directed population (black symbols) and the lifetime of the tethered population (red symbols) plotted as a function of ATP concentration. (D) The fraction of time in the directed state $f_D = (\tau_D / (\tau_D + \tau_T))$ calculated from measured lifetimes in C.

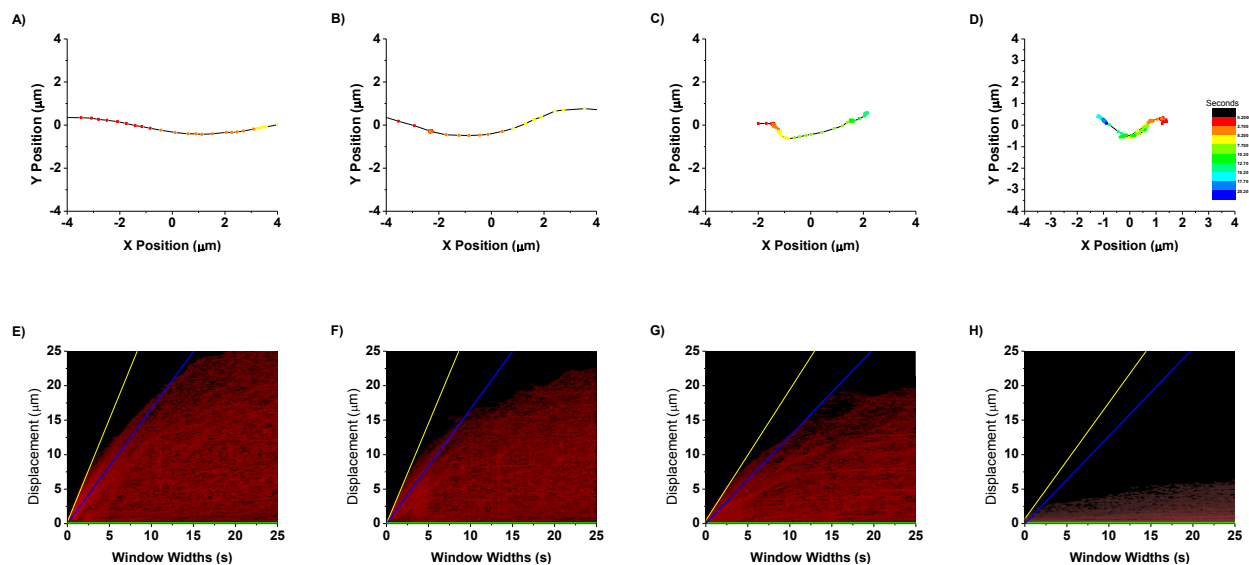


Figure 5 - Calcium Activation of Regulated Thin Filament Motility

(A-D) Example *in vitro* motility trajectories of activated RTF filaments at pCa 3 (A), 5.6 (B), 6 (C), and 6.6 (D). Trajectories were obtained by conducting *in vitro* motility assays using RTFs at $100 \mu\text{M ml}^{-1}$ myosin. The time since the example trajectory entered the plot is represented by the color of the individual points (legend). (E-H) NADA histograms obtained from trajectories acquired under conditions of (A-D) respectively. The blue line is the lower limit of the directed population, which was chosen to include (the set of directed intensities with a zero intercept). The green line is the center of the tethered population which was chosen to include the major horizontal intensity in the NADA histogram. The yellow line is the V_{max} of the NADA histogram.

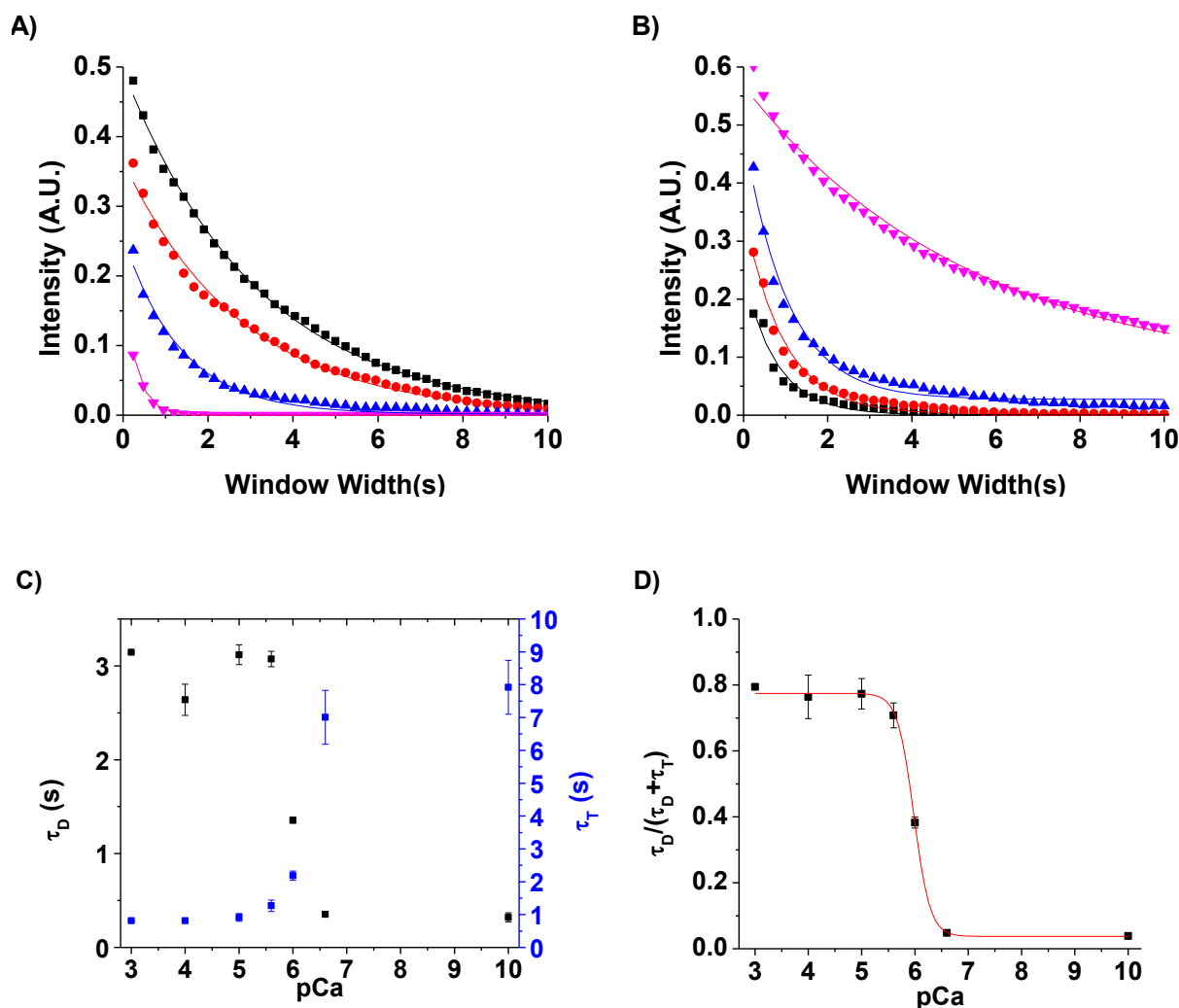


Figure 6 - Lifetime Analysis of Calcium Activation of Regulated Thin Filament Motility

(A-B) Plots of intensities for the directed population (A), the sum of intensities above the blue line, and tethered population, (B) the sum of intensities below the green line, from the matching NADA histograms (Figure 5 bottom) obtained at pCa 3 (black), 5.6 (red), 6 (blue), 6.6 (magenta) at each window width. Intensity decays in A and B were fitted to single exponentials (matching colored lines) to obtain population lifetimes. (C) Plot of measured lifetimes for the tethered population (black points) and the directed population (blue points). (D) Fraction of the directed population ($\tau_D / (\tau_D + \tau_T)$) calculated from the lifetimes in C and fit to a the Hill equation resulting in a pCa_{50} of 5.98 ± 0.01 (SE).

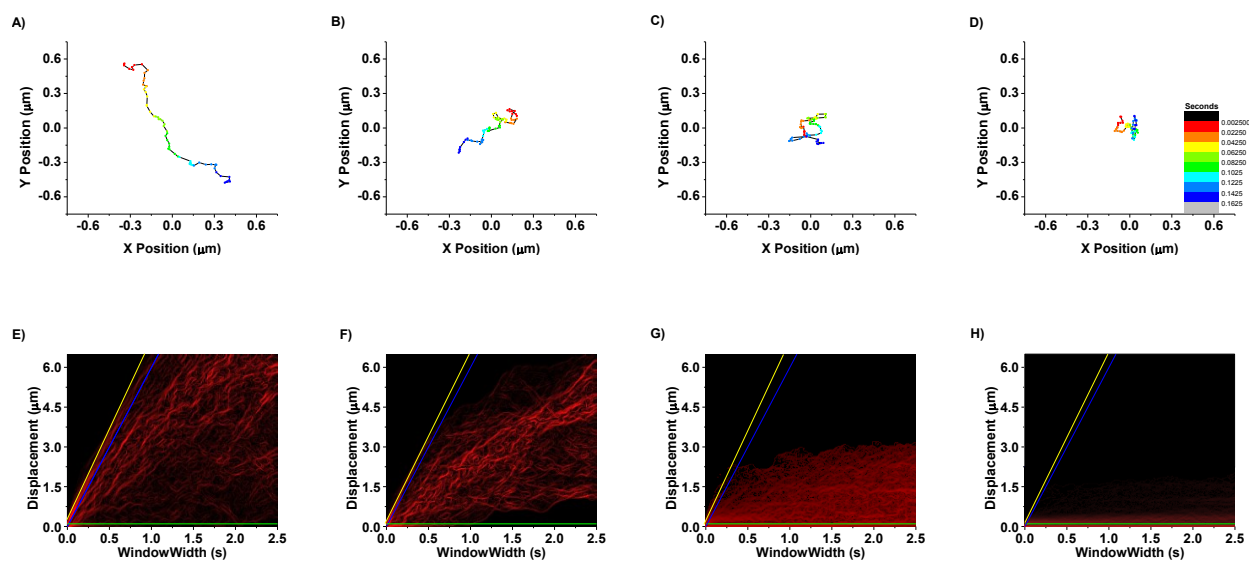


Figure 7 - Calcium Activation of Regulated Thin Filament Motility at 400 Frames per Second

(A-D) Sample *in vitro* motility trajectories of myosin-activated RTF filaments captured at 400 frames per s at pCa (A) 3, (B) 5.6, (C) 6.2 and (D) 6.8. Each example trajectory consists of 65 frames (162 ms) where time is represented by the color of the individual points (legend). (E-H) NADA histograms obtained from trajectories acquired under conditions of A-D respectively. The blue line is the lower limit of the directed population (the set of directed intensities with a zero intercept). The green line is the center of the tethered population which was chosen to include the major horizontal populations. The yellow line is the V_{max} of the NADA histogram.

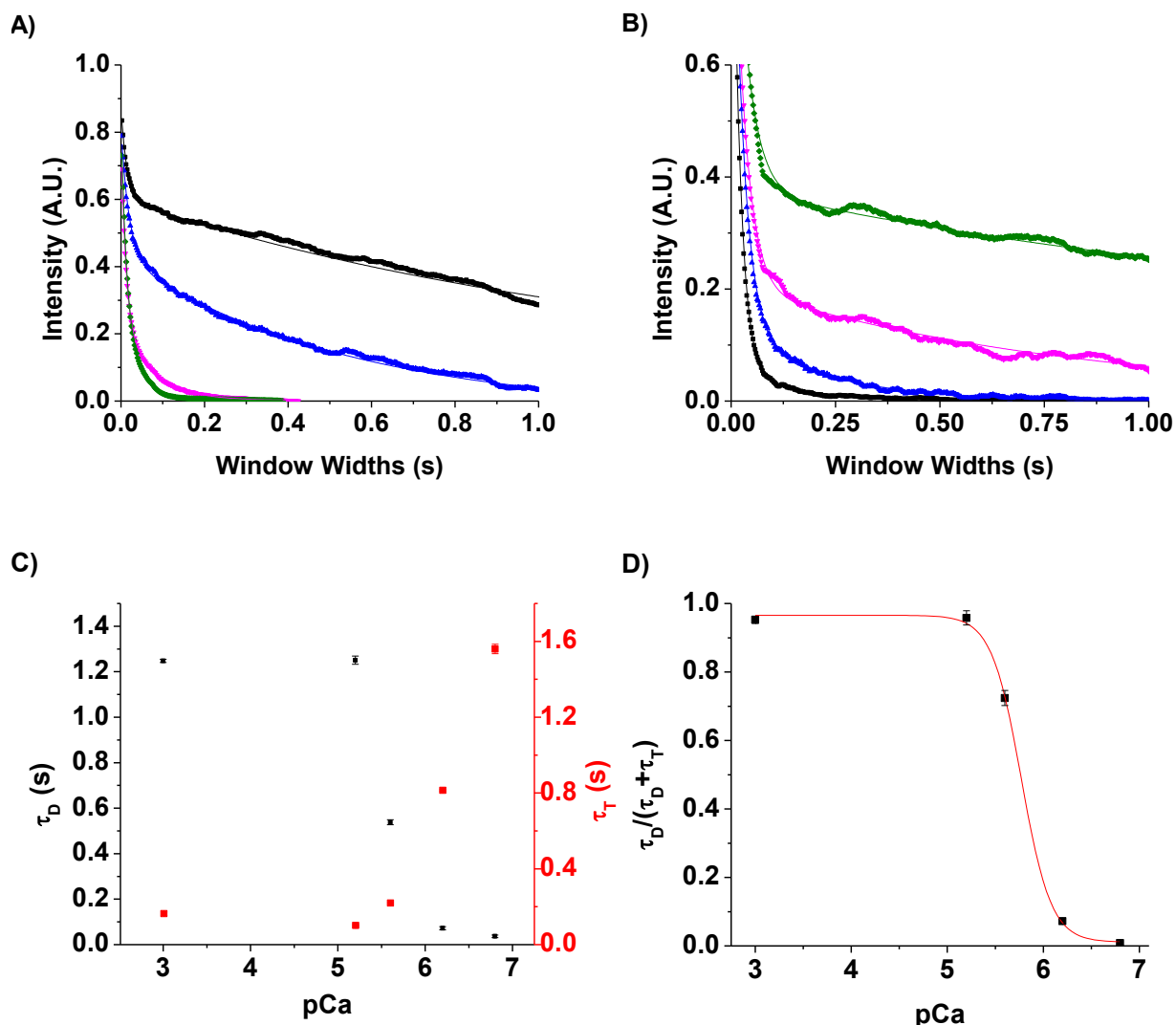


Figure 8 - Lifetime Analysis of Calcium Activation of Regulated Thin Filament Motility from High Speed Camera Trajectories

(A - B) Plots of intensities for the directed population (A), the sum of intensities above the blue line, and tethered populations (B), the sum of intensities below the green line, from NADA histograms obtained at pCa 3 (black), 5.6 (blue), 6.2 (magenta), and 6.8 (green) at each window width (Figure 7, bottom). The intensity decays in A and B were fitted to double exponentials (matching colored lines). The larger magnitude time constants are plotted in C. (C) Measured lifetimes of the directed population (black symbols) and the longer duration lifetime of the tethered population (red symbols) plotted at different pCa values. (D) Fraction of the directed population ($\tau_D/(\tau_D + \tau_T)$) calculated from the lifetimes in C and fit to a dose response curve resulting in a pCa_{50} of 5.77 ± 0.017 (SE).

References

1. Webb, M. *et al.* The myosin duty ratio tunes the calcium sensitivity and cooperative activation of the thin filament. *Biochemistry* **52**, 6437–44 (2013).
2. Sich, N. M. *et al.* Effects of actin-myosin kinetics on the calcium sensitivity of regulated thin filaments. *J. Biol. Chem.* **285**, 39150–9 (2010).
3. Gorga, J. a, Fishbaugher, D. E. & VanBuren, P. Activation of the calcium-regulated thin filament by myosin strong binding. *Biophys. J.* **85**, 2484–91 (2003).
4. BREMEL, R. & WEBER, A. Cooperation within actin filament in vertebrate skeletal muscle. *Nature* **238**, 97–101 (1972).
5. Webb, M. *et al.* The myosin duty ratio tunes the calcium sensitivity and cooperative activation of the thin filament. *Biochemistry* **52**, 6437–44 (2013).
6. Marston, S. B. In Vitro Motility Analysis of Actin-Tropomyosin Regulation by Troponin and Calcium. *J. Biol. Chem.* **270**, 7836–7841 (1995).
7. Sich, N. M. *et al.* Effects of Actin-Myosin Kinetics on the Calcium Sensitivity of Regulated Thin Filaments. *J. Biol. Chem.* **285**, 39150–9 (2010).
8. Kad, N. M., Kim, S., Warshaw, D. M., VanBuren, P. & Baker, J. E. Single-myosin crossbridge interactions with actin filaments regulated by troponin-tropomyosin. *Proc. Natl. Acad. Sci. U. S. A.* **102**, 16990–5 (2005).
9. Kron, S. J. & Spudich, J. a. Fluorescent actin filaments move on myosin fixed to a glass surface. *Proc. Natl. Acad. Sci. U. S. A.* **83**, 6272–6 (1986).
10. Yanagida, T., Nakase, M., Nishiyama, K. & Oosawa, F. Direct observation of motion of single F-actin filaments in the presence of myosin. *Nature* **307**, 58–60 (1984).
11. Margossian, S. S. & S, L. in (Academic Press, 1982).
12. Pardee, J. D. & Spudich, J. A. Purification of muscle actin. *Methods Enzymol.* **85 Pt B**, 164–81 (1982).
13. Potter, J. D. Preparation of troponin and its subunits. *Methods Enzymol.* **85 Pt B**, 241–63 (1982).
14. Smillie, L. B. Preparation and identification of alpha- and beta-tropomyosins. *Methods Enzymol.* **85 Pt B**, 234–41 (1982).
15. Ruhnnow, F., Zwicker, D. & Diez, S. Tracking single particles and elongated filaments with nanometer precision. *Biophys. J.* **100**, 2820–8 (2011).

16. Shi, X., Lim, J. & Ha, T. Acidification of the oxygen scavenging system in single-molecule fluorescence studies: in situ sensing with a ratiometric dual-emission probe. *Anal. Chem.* **82**, 6132–8 (2010).
17. Fabiato, A. & Fabiato, F. Calculator programs for computing the composition of the solutions containing multiple metals and ligands used for experiments in skinned muscle cells. *J. Physiol. (Paris)*. **75**, 463–505 (1979).
18. Russ, J. C. . in *Image Process. Handbook, Fourth Ed.* 475–526 (CRC Press, 2002). doi:10.1201/9780203881095.ch8
19. Patlak, J. B. B. Measuring kinetics of complex single ion channel data using mean-variance histograms. *Biophys. J.* **65**, 29–42 (1993).
20. Squire, J. M. & Morris, E. P. A new look at thin filament regulation in vertebrate skeletal muscle. *FASEB J.* **12**, 761–71 (1998).
21. McKillop, D. F. & Geeves, M. a. Regulation of the interaction between actin and myosin subfragment 1: evidence for three states of the thin filament. *Biophys. J.* **65**, 693–701 (1993).
22. Minkoff, L. & Damadian, R. Biological ion exchanger resins: XI. Actin in Escherichia coli. *Physiol. Chem. Phys.* **9**, 399–403 (1977).
23. Mijailovich, S. M., Li, X., Griffiths, R. H. & Geeves, M. a. The Hill model for binding myosin S1 to regulated actin is not equivalent to the McKillop-Geeves model. *J. Mol. Biol.* **417**, 112–28 (2012).
24. Sich, N. M. *et al.* Effects of Actin-Myosin Kinetics on the Calcium Sensitivity of Regulated Thin Filaments. *J. Biol. Chem.* **285**, 39150–9 (2010).

Chapter 4: Single Particle Tracking Software

Michael Sean Carter, Josh Baker

Department of Biochemistry, University of Nevada School of Medicine, Reno Nevada 89557

Abstract

Single particle tracking is a powerful tool for examining the dynamics of single molecule motion. In the previous chapters, single particle tracking was used in conjunction with our Non Averaging Displacement Analysis (NADA) to characterize the dynamic switching of single molecules between different states of motion present within single particle trajectories to infer the rates of biochemical transitions. In this chapter we provide a manual for the software that was developed for the particle tracking and NADA analysis used in these studies. The analysis software is composed of three complimentary programs; Particle Tracker that generates single molecular trajectories from video images of fluorescent molecules, Trajectory Viewer that allows the user to curate single molecular trajectories and perform the calculations necessary for generating the NADA Histogram, and NADA Histogram Viewer that generates the NADA Histogram image and allows the user to measure the lifetimes of the states of motion present within single molecule trajectories. This chapter will guide the user through the basics of single particle tracking and how to use each program to collect and analyze trajectories to perform the NADA analysis.

Introduction

In this dissertation we have used single particle tracking to measure the position and describe the motion of individual fluorescently-labeled molecules both *in vivo* and *in vitro*. In this chapter we will discuss the principles of single particle tracking, the user interface of our NADA analysis software, and the detailed work flow of a typical NADA analysis of single particle dynamics.

Single Particle Tracking (SPT), is a collection of methods that use computer vision algorithms to measure the position at sub-pixel resolution of individual particles, molecules, or biological structures labeled with high contrast markers from video microscopy images. In general an SPT analysis involves three steps: The collection of high quality video images of the particles of interest; the computational processing of these images and finally the interpretation of the resulting trajectories to produce biologically relevant information.

Collection of High Quality Images for Single Particle Imaging

The first step in conducting SPT experiments is the gathering of high quality, high contrast images. This requires optimizing image acquisition to maximize the signal to noise of the experiment by considering both the imaging method and the labels used to identify the particles of interest.

Widefield Imaging. In this dissertation we primarily imaged *in vitro* motility systems. In this system the fluorescently labeled actin is propelled in a directed manner by unlabeled myosin which is immobilized by nitrocellulose to the coverslip. The movement of actin filaments occurs parallel to the plane of the cover slip and the out of focus fluorophores can be minimized by flushing the chamber before imaging. This geometry

allows *in vitro* motility systems to be well imaged using widefield epifluorescence microscopy with a high NA objective and a sensitive camera (such as an EMCCD). In general widefield epifluorescence imaging methods allows for high frame rate, high contrast, and simultaneous images (where all pixels are collected at the same time unlike scanning systems).

TIRF Imaging. However widefield illumination approaches such as epifluorescence are less than optimal for imaging systems with fluorophores located outside of the focal plane such as the tracking of GFP vesicles within 3T3-L1 adipose precursor cells as performed in chapter 2. In widefield illumination configurations the objective emits a column of excitation light which travels the full thickness of the sample stimulating out of focus fluorophores and creating high background intensities. While it is still possible to track particles moving over high intensity backgrounds, the background intensity limits the confidence of the particles position and often requires additional filtering to mitigate. In chapter 2 the 3T3-L1 adipose cells contained sufficient out of focus fluorescence to obscure the small GFP labeled vesicles when observed using a widefield approach. To accommodate the out of focus fluorescence, we used TIRF illumination. This illumination method uses the property of total internal reflection to limit the excitation of fluorophores to a narrow epifluorescent field within ~ 100 nm of the cover slip. This structured illumination reduces the out of focus fluorescence reducing the background and allows the vesicles to be visualized with high contrast while still allowing simultaneous imaging of the field.

Another important consideration when conducting SPT experimentation is the selection of probes used to label the molecules of interest. Currently there are many

different methods which can be used to label molecules for SPT experimentation, the most commonly used is fluorescence labeling. In fluorescent labeling specialized small molecules and proteins are attached to the particles of interest. When these fluorophores are exposed to specific wavelengths of light the molecule becomes excited and on relaxing emits light at a longer wavelength than it absorbed. This Stokes shift between the excitation and emission light allows the excitation light to be optically blocked permitting only the emitted light to be observed. In order to create high quality images for SPT it is important to choose probes with excitation and emission wavelengths which limit the observation of sources of background / autofluorescence and with sufficient photostability to be visible over the time course of the observation.

Computation of Images to Obtain SPT Trajectories

Once high quality images of the labeled molecules have been collected, the images are computationally processed to localize the particles to a position with sub-pixel resolution. These positions are used to construct trajectories representing the motion of the particle. This analysis is composed of four steps: segmentation, qualification, localization, and linking.

Segmentation. The first step in single particle tracking is dividing the image into regions which may contain fluorescent objects and separating these regions from the background. There are many methods which can be used to segment the image into regions of localized intensity each with their own unique advantages and disadvantages. The most direct method of segmentation is thresholding. This function when applied to the image converts the grey scale intensities in the image to a binary image consisting of black pixels (0) where the grey value is less than the cutoff value

and white pixels (1) where the grey value is greater than or equal to the cutoff value. When thresholding is used effectively the original image is reduced to a black background with clusters of adjacent white pixels. These groupings of adjacent high intensity pixels represent the fluorescence of the labeled molecules of interest. These clusters of high intensity pixels are grouped into potential objects by using a local fill algorithm which identifies independent clusters of white pixels within the binary image generating a collection of possible objects.

Qualification. The collection of possible objects is then filtered by apparent area to eliminate sources of intensity other than the molecules of interest such as shot noise or large aggregations. This process of qualification determines if the region of intensity will be considered for further processing or ignored.

Localization. Following object qualification, the position of each object is determined in sub-pixel resolution by using either centroid-based methods or using an intensity profile analysis. In centroid-based methods the center of an object is determined by averaging the x and y position of each pixel belonging to that object. Centroid-based methods are computationally light and robust to noise but do not offer the level of positional accuracy possible through intensity profile analysis based methods. Intensity profile methods can localize the position of isolated fluorescent sources with higher accuracy than the Rayleigh limit. The improved accuracy is possible by fitting the distribution of intensities from the fluorescent object to a theoretical model of intensity distribution produced by a single sub resolved source. Using numerical optimization such as non-linear curve fitting it is possible to estimate the best parameters for the model and determine the center position of the intensity distribution with higher

accuracy than is possible using other methods. Conceptually the precision of these methods is limited by the signal-to-noise in the intensity distribution, which increases with increasing photon counts ¹.

Linking. The final step in constructing single molecule trajectories is connecting the individual objects between frames. While this seems a simple task, the number of possible linkages grows geometrically with the number of objects that must be linked. The problem is further complicated by collisions of multiple objects and the appearance and disappearance of individual objects. The most common method of linking objects between frames calculates linkages one object at a time. This approach has the advantage of being computationally light while producing a minimum number of bad linkages. Recent revisions to linking algorithms use scoring matrices to consider better possible linkages and further reduce artifacts created by incorrect linking methods.²

Interpretation of Trajectories

Once quality single particle trajectories have been obtained they are routinely analyzed by selecting a physical model to describe the motion of the particles being observed. This allows researchers to obtain descriptive parameters from the measured motion and compare results between conditions. Most current approaches are model dependent, which is to say that the number of dynamic states (e.g., diffusion, directed motion, static) must be defined *a priori*.

In chapter 2 we demonstrated that analyzing single particle trajectories using an incorrect model can result in significant error. However, defining the correct model is often difficult because different dynamic states often cannot be resolved in a trajectory.

In the same chapter we developed a non-averaging displacement analysis (NADA) that allows for the characterization of multiple states of motion within the same trajectory in a model independent manner. The NADA method measures the displacement of single particles as a function of window width in a manner similar to traditional means squared displacement analysis methods. However NADA does not average the displacements for each window width, rather the individual displacements generated by different states of motion segregate into independent populations based on their unique mechanical properties. This allows not only the characterization of the mechanical properties of the different states of motion within the trajectory but permits the measurement of the rates of transition between the different states.

Analysis Software

To facilitate the quantification of SPT experiments I have developed a software suite to measure the motion of fluorescently labeled molecules *in vitro* and *in vivo* from video microscopy of fluorescently labeled particles. The analysis suite is composed of three programs which provide specific functions in the processing of SPT. The first program in the suite is Particle Tracker. Particle Tracker contains a wide range of tools designed to isolate labeled fluorescent objects from video micrographs, localize these fluorescent particles in high spatial and temporal resolution, construct trajectories to represent the motion of these particles between frames, and provide output for further quantitative analysis. The second program in the suite is Trajectory Viewer. Trajectory Viewer is designed to provide a user interface for the rapid review and analysis of SPT trajectories generated by Particle Tracker (or any other software package that can export trajectories as a list of points) and to transform datasets into the format needed

to use in subsequent analyses methods such as MSD and NADA. The final program in the analysis suite is the “NADA Histogram Fitter.” This program provides the tools to visualize NADA histograms, to measure lifetimes of populations within the NADA histogram, and to export images for publication.

Particle Tracker

Like all of the programs in this software suite, Particle Tracker includes many functions which are beyond the scope of this dissertation. Many of these functions were designed to meet specific aspects of collaborator’s projects and have been maintained within the software’s interface to preserve the existing workflows and documentation for these projects. However since these functions are not relevant to my dissertation these aspects of the software’s functionality will not be addressed in this chapter.

The Particle Tracker program is composed of two windows. The Main window (Figure 1) functions as the primary interface for the software giving the user access to the software’s image processing tools. The second window functions primarily to extend the user interface across multiple monitors, and enlarge the views of the images shown in the screen to assist in the annotation of images.

Particle Tracker’s Main window (Figure 1) is composed of several interacting components that provide tools for the user to annotate images, analyze changes in intensity, detect and measure motion, refine the position of individual particles in nanometer resolution, and output the results for scientific interpretation or further processing in additional software. While the program is complex it was built to provide maximum usability and functionality for novice users by grouping like tasks into menus and tabs to simplify the workflow as much as possible. This design strategy has resulted

in a simple straight-forward design which needs little description beyond the summary in Figure 1 - 3 and Tables 1 - 7. However since Particle Tracker uses a series of custom controls to display and interact with images (Figure 1) the interaction and use of these controls needs to be explained in more detail.

The Screen and Playback Controls (Figure 2) are a set of custom controls that display images and provide a graphical interface for tools that allow the user to interact with the image by clicking on the image directly. This control is composed of many simple components like the “Frame Selection Bar” (Figure 2E) and the Frame Range Selection (Figure 2E) which have functions described in Table 5.

The primary function of the screen control is to display an image composed of the active layers selected in the “Image Interaction Tool Bar” (Figure 1D). Each layer as described in Table 4 is used to deliver specific information to the user such as the ROIs and the results of the tracking methods and can be enabled and disabled by clicking the layers name in the Drawing Layer Tool Bar (Figure 1E). In this way Particle Tracker can be customized to display only the relevant data for the specific tasks being performed. Beyond the screen’s function as a display, it functions as a data input method allowing the user to choose a tool from the “Image Interaction Tool Bar” (Figure 1D) and rapidly add regions of interest and annotations to the image, perform measurements both automated and manual, and interact directly with the displayed data by choosing a tool and clicking on the screen. The functions of these tools have been summarized in Table 3.

Loading Images

The first step for any workflow in Particle Tracker is loading images into the software. Currently Particle Tracker is designed to accept images in two different image formats, tiff stacks which are composed of a series of images within a single file and directories of numbered tiff files where the individual frames of the movie are present as individual numbered tiff files contained within a directory. In addition to the image file's format, images loaded into particle tracker must be encoded as 8 bit grey scale images using either raw uncompressed formats or the lossless "LZW Compression" scheme. This will ensure maximum compatibility across different systems and ensure that no information is compromised by the image compression. If an image will not open in Particle Tracker I suggest using either the NIH's ImageJ or IrfanView by Irfan Skiljan³ to convert your files into a compatible format before using them in Particle Tracker.

The process of loading images in Particle Tracker is performed by two different menu options. The first option is the "Open>Tiff" menu option under the File menu. This option is designed to open tiff stacks containing multiple frames of a single movie. Once the user selects the "Open>Tiff" menu option Particle Tracker creates a standard open file dialog allowing the user to select a single stacked tiff file. After the file has been loaded the software will prompt the user to establish the lateral calibration of the image in units of $\mu\text{m}/\text{pixel}$. The second option is the "Open>StanfordCam" menu option under the File menu. This option is designed to load movies as a series of numbered tiff images contained within a directory and its subdirectories. When the user selects this option Trajectory Viewer will generate a standard window folder browser dialog and allow the user to select the directory containing tiff files for the movie. After Trajectory

Viewer has indexed the files contained within the selected directory and subdirectories the software will prompt the user to establish the lateral calibration of the image in units of $\mu\text{m}/\text{pixel}$. After the user has successfully loaded the movie image into Particle Tracker the user then can use the software's other tools regardless of which loading option was used.

Loading and Saving Program Data

Unlike other software packages such as ImageJ ⁴ or FIESTA ⁵, Particle Tracker can save the progress of any analysis by storing the memory data used by the software to the hard drive allowing the user to reload the data behind the analysis at any point. This not only allows the user to save their progress but allows the user to audit problematic trajectories back to the fluorescent objects they represent. Currently Particle tracker uses two different methods of saving and loading images. The first format uses Microsoft's built in serialization API to copy a portion of Particle Trackers memory to a single file. This is automatically performed by several of the functions within Particle Tracker and can be manually performed by using the "Save>DataFile" function under the File menu (Figure 1B). When this option is selected from the menu Particle Tracker will create a save file dialog allowing the user to set the path of the saved *.dat file. These ".dat" files can be loaded into Particle Tracker by first starting a new instance of Particle Tracker and selecting the "Open>DataFile" option under the file menu. This option will then copy the saved memory value back into Particle Tracker and load the associated image files.

The single file method works well for most users of Particle Tracker particularly when the analyzed files have less than 1000 frames. However since the Microsoft serialization

API requires that the entire data file be loaded into memory at once, the load time and system resources required to load a large data file increased proportionately with the amount of data in memory. Because the high frame rate data movies regularly contained 48k~50k frames the single file save method was too slow and memory intensive for these observations. In order to support saving data from these movies it was clear that a more advance save method was needed. However in order to continue support of the original save method for Particle Tracker's user base, a second more efficient method of memory serialization was implemented parallel to the original one.

The second save method "Multipart Data File" was designed to overcome the memory limitations and long loading time for large datasets described above by splitting the serialization of the data into multiple parts. This allows the software to load the data as a series of small parts eliminating the need to load the entire data file into memory at once. To save a file using this method the user can select the "Save>Multipart Data File" option under the main menu (Figure 1B). This option will generate a standard save dialog that will allow the user to select the path for the multipart save file which by default will have the extension ".mps". Once the save path has been set Particle Tracker will generate a subdirectory with the name of the ".mps" file in the save paths directory. This directory is then filled with a series of small numbered files with file names beginning with pound followed by 8 numbers and the extension ".framedata." Each of these files contains the memory data for the individual frame indicated by the file number. Once Particle Tracker has saved the data for each frame it will generate the ".mps" file which contains the other data necessary to reload this saved data. This data can be reloaded by starting a fresh instance of Particle Tracker and selecting the

“Open>Open Multipart Save” menu option from the main menu (Figure 1B). This option will create an open file dialog allowing the user to select the “.mps” file. Once the “.mps” file is selected Particle Tracker will load the base file and then each of the individual “.framedata” files will be loaded in succession. Once the deserialization is complete via either method Particle Tracker will prompt the user for the lateral calibration of the image and the user can use Particle Tracker from the point where the save file(s) were generated.

Centroid Based Single Particle Tracking Implementation.

In order to understand the behavior of centroid-based single particle tracking, in this section we will discuss the specifics of our implementation of the centroid tracking algorithm in Particle Tracker. As discussed in the introduction the process of tracking single particles involves four steps: Segmentation, Qualification, Localization and Linking. Particle Tracker contains a custom centroid-based particle tracking algorithm which implements each of these steps to allow the software to identify single molecules and quantify their motion.

The first step in this process is segmenting the fluorescent signal of the particles of interest from the background. In Particle Tracker segmentation is performed via an intensity-based thresholding method that accepts the 8-bit grey scale frame from the tiff stack and a cutoff value to produce a binary image. The binary image is created by comparing the intensity value of each pixel in the 8-bit image to the cutoff value; if the intensity value of the pixel is equal to or greater than the cutoff value, the pixel in the tested image is set to 1 (white) otherwise the pixel is set to 0 (black). This results in a

black and white image with large white “blobs” representing regions of fluorescent intensity above the cutoff value and several smaller blobs that represent other sources of intensity such as shot noise or out of focus light.

After segmentation the individual binary blobs in the image must be qualified to remove the blobs originating from sources of intensity other than the particles of interest. Particle Tracker qualifies the blobs within the binary image by area. The process of qualification requires the use of three complimentary functions. The first function detects the presence of each unique binary blob. This function discovers blobs within the binary image by scanning each pixel one at a time. When a pixel with the value of 1 (white) is located, the location of the pixel is then passed to the second function.

The second function locates the connected pixels for the detected blob by using a connected component analysis. This analysis starts by adding the pixel located by the previous function to a queue of possible connected pixels. The function then removes the last pixel added to the queue. If the pixel has a value of 1, the pixel is added to the list of component pixels belonging to the newly discovered object and the pixel value is set to zero to prevent the point from being counted twice. Then each of the 8 adjacent pixels is then added to the connectivity test queue. The analysis then repeats this process until the queue is emptied and all of the connected points for the new blob have been identified. The resulting collection of points is then passed to the third function for qualification.

The third function evaluates if the pixels discovered in the previous two steps are representative of a fluorescent object or the result of noise within the image. In its

current implementation, Particle Tracker assumes that the majority of tracking based noise is the result of individual pixels randomly exceeding the threshold value resulting in small binary blobs with areas less than the particles of interest. In order to remove the blobs generated by this shot noise each detected object is qualified by the area of the object in pixels. If the measured area of the detected object exceeds the minimum value, the collection of pixels is qualified as a fluorescent object and is stored in the programs memory for later. After the third function is complete Particle Tracker continues scanning the image for unique binary blobs within the image until the next possible object is detected or until the first function has tested all of the remaining pixels within the image.

Using the fluorescent objects qualified in the previous step, Particle Tracker determines the location of each fluorescent object in sub-pixel resolution using a centroid based localization method. Centroid based localization is a non-iterative computationally simple method that calculates the center of mass for the particle based on the position of the component pixels belonging to the object being located. In Particle Tracker's implementation of centroid based localization, the center of mass of each object is calculated by averaging the X and Y components of each pixel belonging to the object using the equations $C_x = \frac{\sum_i^N X_i}{N}$ and $C_y = \frac{\sum_i^N Y_i}{N}$ where C_x and C_y are the components of the particles location, X_i and Y_i are the X and Y components of the object's i th pixel, N is the number of pixels belonging to the object and i is the index of the summation. This method results in a location of the particle with an accuracy within $1/10^{\text{th}}$ of a pixel of the object's true center of intensity for images with good signal to noise ratios ⁶.

The final step in single particle tracking is trajectory building and is performed after all other frames in the tiff stack have been processed as described above. The trajectory building process is commonly referred to as linking as it calculates linkages between the individual observed fluorescent objects detected in each frame to create trajectories. In Particle Tracker linking is performed by using processive velocity minimization. The algorithm functions by building a table of all possible displacements between the objects in the current frame and the objects contained in the next frame. This scoring table allows for Particle Tracker to determine “the best linkage” I.E. the linkage with the smallest displacement between two objects in different frames. Once the best linkage has been determined, the linkage is saved and all other linkages involving these two objects are removed from the table. The next best linkage is then chosen and any linkages involving these objects are also removed from the table. The process of choosing the “best linkage” is repeated until the table is either empty or the smallest displacement in between objects in the table exceeds 10 pixels in length. The process of choosing the “best linkage” is repeated until the table is either empty or the smallest displacement in between objects in the table exceeds 10 pixels in length. Once all possible linkages have been made Particle Tracker then indexes forward one frame and calculates the linkages between the new current frame and its following frame. The process of linking is then continued as described for each frame in the video. The resulting trajectories are then available for use in other functions within Particle Tracker or can be exported to Trajectory Viewer as described below.

Performing Centroid-Based Particle Tracking Analysis for Use in Trajectory

Viewer

The semi-automated centroid based tracking functions within Particle Tracker can be accessed by the “Qdot Controls”, “Better Qdot Tracking” and the “NADA Tracking” tabs contained within the Task Tabs control (Figure 1F). Each of these tabs contains the tools necessary for performing centroid based tracking as well as for conducting the task specific workflows described in Table 6. The “Qdot Controls” and “Better Qdot Tracking” workflows are discussed in the “Performing Super Resolution Measurements using Intensity Profile Refinement” section below and the “NADA Tracking” tab and its work flow will be discussed in this section.

The “NADA Tracking” tab within Particle Tracker’s Task Tabs control (Figure 1F) implements the tools necessary to perform the centroid based tracking of single particles and the tools required to export the resulting trajectories as files compatible with Trajectory Viewer. The first step in this work flow is to load a fresh instance of Particle Tracker. Once loaded the user then can load a tiff stack as described in the “Loading Images” section. Once the tiff stack is loaded in Particle Tracker the user should use the Frame Selection Bar (Figure 2B) to review the tiff stack for defocus events, changes in intensity and any other aberrations that may effect the quality of the tracking process. If aberrations are present in the image they can be address by either pre-filtering the tiff images in software packages such as ImageJ or avoided by using the Frame Range Selection (Figure 2E) buttons to limit the analysis of the tiff stack around these aberrations. If the Frame Selection Bar does not change the image within

the Screen (Figure 2A) the user should check that the “RawImage” drawing layer is enabled and the “Process” layer is disabled in the Drawing Layer Tool Bar.

After the tiff files have been loaded and reviewed the user can select the “NADA Tracking Tab” from the Task Tabs control (Figure 1F) and enable the “Process” Drawing Layer from the Drawing Layer Tool Bar (Figure 1E). This will give the user access to the workflow’s functions and allow them to view the setting previews generated by the following steps and the intermediate images generated by the analysis described above. The user should then click the “Run First” button (Table 6 “NADA Tracking Tab” H). This tool creates the required data structures to hold data resulting from the centroid tracking algorithm. The “Run First” button also calculates the average and standard deviation of the pixel intensity within each frame in the tiff stack. This calculated value is then used to set an initial guess for the segmentation step of the analysis.

When the “Run First” buttons action is complete the user can set the threshold values for the segmentation step of the analysis by moving the Threshold Tracking Bar (Table 6 “NADA Tracking Tab” B). As the user moves the Threshold Tracking Bar, Particle Tracker will preview the segmentation step of the tracking algorithm on the current frame selected by the Frame Selection Bar (Figure 2B). This preview is visible on the “Process” drawing layer. The best practice for setting the threshold value is to adjust the value to remove as much of the background as possible without eroding the silhouette of the particles of interest. In addition the threshold value should be tested on several different frames within the tiff stack to verify the settings performance throughout the video.

After the threshold value has been set the minimum size for qualification can be established. This value is set by moving the “Minimum Object Size Track Bar” (Table 6 “NADA Tracking” D). As the user moves the Minimum Object Size Track Bar, Particle Tracker generates a preview of the qualification process visible on the “Process” drawing layer. In this preview the green blobs passed the qualification process while the red objects fail. The orange numbers adjacent to each object indicates that objects area in pixels. The minimum size qualification value should be set to a value that excludes (sets red) any blobs created by the shot noise but keeps the particles of interest within the image. When the user has determined the optimum setting the set button (Table 6 “NADA Tracking” F) can be clicked to “set” the threshold constant and minimum object size values for the tracking algorithm in the next step.

At this point the user can either choose to perform the NADA histogram workflow directly or perform the centroid based tracking of the particles first. Performing the tracking step first is recommended because it allows the user to review the results of the centroid tracking before any other steps are performed. Once the user is satisfied with the threshold and minimum object size setting described above, the centroid tracking process can be started by clicking on the “Track” button (Table 6 “NADA Tracking” G). This will perform the centroid tracking algorithm on the range of frames set in the Frame Range Selection Buttons (Figure 2E) using the threshold and size qualification value set above.

While the tracking algorithm is performed the intermediate images from the tracking process are visible in the process tracking layer and the estimated time till completion is displayed in the Status Bar (Figure 1H). It is suggested that the user enables the

“RawImage” and “Fobjs” drawing layers and disable the “Process” layer using the Drawing Layer Tool Bar (Figure 1E). These changes to the active drawing layers allow the user to view the result of the tracking algorithm projected on top of the raw tiff image as a green boarder drawn around the edge of each detected fluorescent object in the field. If the “DrawObject#” option is checked, numbers will appear over each of the detected objects. However until the object linking process is complete at the end of the tracking algorithm, these numbers will correspond to the order each object was detected and not correspond to individual trajectories.

If the tracking algorithm fails to properly identify the fluorescent objects in the field, the user can click the stop button at any point to stop the tracking process (Figure 1G “Stop” button). After the algorithm stops, the quality of the tracking can be reviewed using the Frame Selection Bar (Figure 2B). If changes to the frame range or the threshold and size qualification values are needed they can be performed as described above. The tracking algorithm can then be restarted by clicking the set and then track buttons. Any frames that are reprocessed will replace the previous frames. If changes are made the user must rerun the linking function again at the end of the process for the entire range of analyzed frames by setting the first and last frames using the frame range selection buttons (Figure 2E) and selecting the “Sliding Motility>Link Fobjects” menu option. This will repeat the linking step of the tracking algorithm ensuring that the generated trajectories will include sections of the tiff stack that were analyzed separately.

If the user chose to skip manually running the tracking algorithm or after the tracking algorithm has completed, the user can proceed to run the NADA work flow. This

collection of functions will export the centroid based trajectories from Particle Tracker as trajectory files compatible with Trajectory Viewer.

Before performing the NADA work flow the user should ensure that the selected frame range (Figure 2E) includes all of the frames within the tiff stack to be processed and if the user has not already performed the tracking algorithm that the threshold and size qualification values are set as described above. The work flow is started by clicking the “NADA Work Flow” button (Table 6 “NADA Tracking” I). After the user starts the work flow, Particle Tracker will generate several prompts for the user before executing the required calculations.

The first four prompts in the NADA work flow establish the minimum distance and duration filters used to prevent the export of trajectories with total traveled distances and durations less than the set minimums. The first two prompts ask for the exposure time in units of seconds and the x/y calibration factor in units of μm per pixel for the loaded tiff stack. This establishes the calibration for the filtering functions. The next prompts establish the minimum total displacement in units of μm and the minimum time in seconds for the filters. The minimum distance filter should be configured based on the mechanics of the system being measured and the minimum duration filter should be set to a value no shorter than 3 frames to prevent the exported data from containing an excess of short lived trajectories.

The remaining three prompts will establish the behavior of Particle Tracker during the remainder of the NADA work flow. The first of these prompts will ask the user if Particle Tracker should close after the work flow is complete. Closing Particle Tracker on completion is particularly useful when working with multiple instances of Particle

Tracker in parallel because as each instance of Particle Tracker completes the NADA workflow, Particle Tracker will close freeing system resources. The second prompt asks the user if Particle Tracker should save when the NADA work flow is complete. Selecting “Yes” will generate a save file before exporting the trajectory data and selecting “No” will not. The final prompt asks the user if they wish to perform centroid based tracking on the tiff stack. If the user selects “Yes”, Trajectory Viewer will perform the centroid based tracking described above. If the user selects “No”, Particle Tracker will use the existing centroid based tracking data within the programs memory to build trajectories for export. After the user has answered the prompts, Particle Tracker will perform the required calculations for the NADA work flow.

The NADA workflow calculations starts by performing centroid based tracking using the setting on the “NADA Tracking” tab (Figure 1F) on the frame range within the tiff stack set by the Frame Range Selection Buttons (Figure 2E). If the user selected “No” to the “perform centroid based tracking” prompt above, the tracking step will be skipped. Following the tracking step Particle Tracker will save the analysis progress using the standard save method described above. This step will be skipped if the user indicated that they did not wish to save during the initial prompts.

After the previous steps are complete, the results of the centroid based tracking algorithm will be internally converted into trajectories compatible with Trajectory Viewer. These trajectories will then be filtered for duration and displacement using settings established by the first four prompts. Particle Tracker will then create a new subdirectory with the same name as the tiff stack to hold the exported trajectory files. Each trajectory remaining after the filtering process will be written to the directory as

individual files. Each of the resulting files will have a filename indicating the tiff file's name, the time and duration filter settings, and a track number corresponding to the fluorescent object in Particle Tracker.

As the final step of the NADA workflow within Particle Tracker, if the user selected the option to close on completion, Particle Tracker will exit increasing the available system resources.

Performing Super Resolution Measurements using Intensity Profile Refinement

Due to the diffraction of light, the maximum resolution achievable by an imaging system such as a high magnification light microscope is limited by the wavelength of the light observed and the numerical aperture of the objective. This diffraction limit can be approximated by the equation $d = \frac{\lambda}{2NA}$ where d is the minimum resolvable spot size (Abbe Resolution Limit), λ is the wave length and NA is the numerical aperture of the objective. This resolution limit significantly complicates the visualization of small features and structures within the image requiring the implementation of complex super resolution methods to deconvolve the effect of diffraction present within the image before these structures can be visualized in detail. These same properties of light that convolute small details within the image result in the intensity profile of any isolated sub-resolved point source within the microscope field to appear as a diffraction pattern known as an Airy disk. Since the intensity profile of the Airy disk is well understood, the center position of the point source can be determined in high resolution by iteratively fitting the intensity profile of the sub-resolved particles to a Gaussian equation. Methods based on this principle have been documented achieving positional resolutions of fluorescent sources within an error of single nanometers⁷. However more critical

reviews place the practical limitations of these methods within a 10 nm effective resolution or 0.02pixels⁶.

The process of obtaining super resolution positions for individual sub-resolved objects within Particle Tracker is composed of two separate steps. In the first step the user performs centroid based tracking of the fluorescent objects within the tiff stack. In the second step the objects detected by centroid tracking are refined by iteratively fitting the intensity profile of each object. The tools required to perform intensity based positional refinements are implemented on the “Qdot Controls” and the “Better Qdot Tracking” tabs within the Task Tabs control. (Figure 1F) Both tabs provide access to the same super resolution methods but the “Better Qdot Tracking” tab includes additional modifications to the centroid based tracking algorithms. These modifications implement additional filters that allow the user to increase the effective exposure of an image and average adjacent pixels to reduce random noise within the image. These filters increase Particle Trackers ability to perform centroid based tracking of fluorescent objects in images with low signal and high noise which are typical of images produced by high speed microscopy.

The first step in performing intensity profile based refinement of single particles is loading and reviewing the images in the tiff stack as discussed in the previous section. After the tiff stack has been loaded and reviewed, the user can perform centroid based tracking on the loaded tiff stack using the tools located in the “Qdot Controls” or “Better Qdot Tracking” task tabs. As described in the previous section, centroid tracking requires the user to establish both a threshold value and a minimum object size. The threshold value is set by enabling the “Process” drawing layer and then adjusting the

Threshold Tracking Bar (Table 6 “Better Qdot Tracking” C) until the preview on the Process layer shows that the current threshold value removes as much of the background as possible without eroding the silhouette of the particles of interest.

After the user has established the threshold value the “Set” button (Table 6 “Better Qdot Tracking” H) should be clicked to build the necessary data structures before proceeding to set the minimum object size. The minimum object size for the qualification step of the centroid tracking algorithm is set by moving the “Minimum Object Size Track Bar” (Table 6 “Better Qdot Tracking” F). As the user moves the Minimum Object Size Track Bar, Particle Tracker generates a preview of the qualification process visible on the “Process” drawing layer. In this preview the green blobs passed the qualification process while the red objects fail. The orange numbers adjacent to each object indicates that objects area in pixels. The minimum size qualification value should be set to a value that excludes (sets red) any blobs created by the shot noise but keeps the particles of interest within the image. When the user has determined the optimum settings, the tracking process can be started by clicking the “Set” button (Table 6 “Better Qdot Tracking” H) and then the “Track” button (Table 6 “Better Qdot Tracking” I). This will pass the threshold value and minimum object size values to the tracking algorithm and start the tracking process.

As Particle Tracker is running the centroid tracking algorithm, the intermediate images from the tracking process are visible in the “Process” drawing layer and the estimated time till completion is displayed in the Status Bar (Figure 1H). It is suggested that the user enables the “RawImage” and ”Fobjs” drawing layers and disable the “Process” layer using the Drawing Layer Tool Bar (Figure 1E) to allow the user to view

the result of the tracking algorithm projected on top of the raw tiff image. If the tracking algorithm is not functioning as expected, the user can repeat any portion of the centroid tracking process using the process described above in the Performing Centroid-Based Particle Tracking Analysis for Use in Trajectory Viewer section.

After the centroid tracking algorithm has been completed the user has two options to perform the intensity profile refinement algorithm on the resulting detected fluorescent objects. The user can either perform intensity profile refinement on all of the detected objects using the “FIONA ALL” button located on the “Better Qdot Tracking tab” or perform the same algorithm individually on objects selected by the user. The first step in either process is to configure the drawing layers. It is recommended that the user enable the “RawImage”, “ROI” , ”Fobjjs” and “FionaDots” drawing layers and disable the “Process” layer. This selection of active drawing layers will allow the user to visualize regions of fluorescence detected by the centroid tracking algorithm as green outlines and the result of the intensity profile refinement process as tan crosses projected on top of the raw tiff image. After setting the drawing layers the user can then proceed to use the “FIONA ALL” tool or perform the intensity profile refinement process on only selected objects.

To perform the intensity based profile refinement algorithm on all of the fluorescent objects detected by the centroid tracking algorithm the user can use the “FIONA ALL” tool. This process is performed by first setting the frames to be processed using the “Frame Range Selection Buttons” (Figure 2E). Next the user should ensure that the drawing layers mentioned above are enabled and the “Process” layer is disabled. These drawing layers settings will allow the user to observe the progress of the intensity based

profile refinement algorithm while the process is running. Finally the user can then start the intensity profile based refinement process by clicking on the “FIONA ALL” button located on the “Better Qdot Tracking” task tab. (Table 6 “Better Qdot Tracking” J)

The “FIONA ALL” tool will then index through each of the selected frames, measure the intensity profile of each object and use the non-linear curve fitter to fit a Gaussian equation to the intensity profile. The resulting X and Y center parameters will then be used as the best center position for the detected object. The resulting fit parameters are then saved. While the “FIONA ALL” tool is running, Particle Tracker will display each of the resolved positions on the “FionaDots” layer as they are available. In addition Particle tracker will display the ETA for the entire “FIONA ALL” process in the Status Bar (Figure 1H) and the progress and ETA for processing the current frame in Particle Trackers Title Bar. When the process is complete the user can export the resulting super resolution trajectories using the “Save All” button (Table 6 “Qdot Controls” K) or the “Save All Qdot Data” button (Table 6 “Better Qdot Tracking” K) buttons as described below.

In order to select individual objects for intensity profile refinement the user must specify the object numbers for the objects to be processed. This can be performed by either manually entering object numbers delimited by commas into the FIONA textbox manually or by using the “Select Object” tool.

To manually specify the object numbers to be processed the user must first enable the drawing of the object numbers for the centroid tracked objects by checking the “DrawObject#” checkbox located under the Drawing Layer Tool Bar (Figure 1 E) and enabling the “Fobj” drawing layer. The object numbers for each object will be drawn in

red above each object allowing the user to manually select each of the objects by copying each objects object number into the FIONA textbox. Alternatively, the user can use the “Select Object” tool in the Image Interaction Tool Bar (Figure 1 D). The Select Object tool function is enabled by clicking on its name within the Image Interaction Tool Bar (Figure 1D). When the tool is selected its background will darken indicating that it is currently selected. The “Select Object” tool allows the user to select a single object by clicking on the fluorescent object directly or to select multiple objects by holding select. When the user selects an object or objects, the object(s) the trajectory is calculated and displayed as a red line on the ROI drawing layer. The selected object(s) object number(s) is/are automatically entered in the FIONA textbox during the selection process.

Once the user has selected all of the objects they wish to refine the user can start the process by clicking the FIONA button (Table 6 “Qdot Controls” H). The user will be prompted to save as described above in the “Loading and Saving Program Data” section. After Particle Tracker has saved, the intensity profile refinement process as described above will begin. While the process is running the individually refined positions will appear on the “FionaDots” layer as each fluorescent position is refined and the ETA to complete the process will be displayed in the Status Bar (Figure 1H).

After the user has successfully completed the intensity profile refinement process as described above, the resulting super resolution trajectories can be exported as individual trajectory files compatible with Trajectory Viewer. The exportation process is performed by clicking on either the “Save All” button (Table 6 “Qdot Controls” K) or the “Save All Qdot Data” button (Table 6 “Better Qdot Tracking” K). Particle Tracker will

then prompt the user to set the minimum duration of the exported trajectories in units of frames. This filter removes trajectories from the exportation process with durations less than the specified value in a manner similar to the export filters in the previous section. It is recommended that the user set a value of at least 3 frames to ensure that the resulting trajectories contain enough points to prevent objects resulting from noise from being exported. After the prompt Particle Tracker will create a folder based on the tiff stacks file name and generate an individual trajectory file for each remaining trajectory. Each of the resulting file names will include the tiff stacks name without the extension, the “Qdot#”, and the extension “.csv”. After the export is complete the resulting files can be processed in trajectory viewer.

Trajectory Viewer

The purpose of the Trajectory Viewer program is to assist the user in the rapid evaluation of single particle trajectories and to perform the calculations needed to conduct advance analysis methods such as MSD and NADA on single particle trajectories obtained from Particle Tracker or other single particle tracking suites. Trajectory Viewer’s interface is composed of a single window containing a menu bar (Figure 4B) that provides access to a wide range of functions to perform tasks such as loading data, a control cluster (Figure 4C) containing tools to interact with the plot controls as well as for manually qualifying trajectories and four data plots (Figure 4 D,E,G, and H). The four plots are dynamically linked and display different visualizations of the motion represented by the loaded trajectory. The first plot (Figure 4G) displays raw points within the trajectory. The second plot is a plot of the instantaneous velocity calculated at an interval of one frame as a function of time (Figure 4H). The third plot is

a histogram of the instantaneous velocities (Figure 4E) and the final plot is a plot of the distance from the first point as a function of time (Figure 4D). In addition to displaying trajectory data, the plots in conjunction with the Plot Tools (Figure 5C), allow the user to directly interact with the data by left clicking in the plots to perform the selected function on the data or by right clicking, copy the data from the plot to the clipboard for use in additional analysis packages such as origin or excel. The individual functions of these controls and menus are described in greater detail in Figures 4 - 6 and Tables 8 -10.

Opening Trajectory Viewer and Setting Calibration

When opening Trajectory Viewer, the software will locate and load the last saved settings from the “Settings.csv” file in the executable’s folder. In Trajectory Viewer the settings file preserves the calibration and viewing settings between instances of the software. By automatically loading the last settings, Trajectory Viewer minimizes data entry errors and speeds up the process of analyzing multiple datasets. If the “Settings.csv” file cannot be found the program will close and regenerate the missing settings file from the internal defaults.

After the software has successfully loaded the settings file. The user is prompted by two input boxes to set the calibration for the software. These calibration constants will allow the software to convert the trajectories to the correct units as needed. The first prompt asks the user to enter the frame rate of the trajectory in the units of frames per second. The second prompt asks for the lateral calibration of the trajectories in units of pixels per μm . By default the input boxes are automatically populated with the calibration factors loaded from the settings file allowing the user to reload the program after initially setting the calibration for the dataset.

Loading Trajectory Files

After the user has entered the calibration factors for the trajectories Trajectory Viewer can load single or multiple “.CSV” files containing trajectory data generated by Particle Tracker or by third party software packages. These trajectory files must contain the following columns delimited by commas to be loaded into Trajectory Viewer. The first column contains the “ObjectNumber” which is a positive number from 0 to integer max. This column functions to identify the object responsible for a trajectory within Trajectory Viewer which can be used to backtrack each trajectory to the originating fluorescent object within Particle Tracker’s data files. The second, third, and fourth, columns contain the time component and the x and y positions in units of frames and pixels respectively for each point within the trajectory. The last two columns are only used when the position of the particle has been refined by intensity profile analysis using non-linear curve fitting, for all other cases these columns should have a value of “1.” When the trajectory has been refined by intensity profile refinement, these columns contain the adjusted R squared value from the intensity profile refinement fit for the X and Y positions respectively. These numbers are not used directly by the software but are useful when identifying outliers in the trajectory refinement process since low adjusted R squared values are associated with low quality fits to the intensity profile.

Trajectory Viewer has two different options for loading trajectory files: “Open Single Trajectories” and “Open Directory of Trajectories.” Both of these menu options are located under the “Open” menu in Trajectory Viewer’s main menu (Figure 4B). These two menu options function identically except for the dialog used to select the trajectories. When the user selects “Open Single Trajectories” a standard open file

dialog is generated allowing the user to select individual or multiple trajectories to load. When “Open Directory of Trajectories” is selected a folder browser dialog is created allowing the user to select a directory. When the user chooses a directory the software will search the selected directory and its subdirectories for trajectory files and produce a list of “selected” trajectories similar to the list produced by the open file dialog. For both menu options after the user clicks ok Trajectory Viewer will load the selected trajectory file paths into an internal list. The user is then asked if they wish to view all the files at once. If the user selects “Yes” the software will load all of the files simultaneously. It is suggested that before viewing a large collection of trajectories that the user uncheck the “Draw Dots” option under the view option menu. This will greatly speed the rendering of the plots within trajectory viewer and prevent a possible GDI+ overflow error on machines with limited ram. If the user selects “No” the software will load the first trajectory in the list. Regardless of the option the user selects they can view each trajectory individually by using the “GoNext” (Figure 5D) and “GoBack” (Figure 5E) buttons in the control cluster to index through the list of trajectories.

Interacting with Data Plots

In addition to visualizing the trajectory and derived values such as the instantaneous velocity (Figure 4H) and total displacement (Figure 4D), the plot controls (Figure 4 D,E,G, and H) in trajectory viewer’s window allow the user to interact with the data contained within the plot controls intuitively by using the mouse. The user’s interaction with the plot controls is mediated through the Plot Tools (Figure 5C) located within the Control Cluster (Figure 4C). The plot tools control functions to select the action performed when the user clicks on the plot control. When an item in this list is

highlighted the tool is active and the plot controls will respond to left clicks using the behaviors described below.

Using the “Move” Tool

When the “Move” tool is selected from the plot tools (Figure 5C) the plot controls will respond to left mouse clicks by centering the plot on the click’s position while keeping the same range of x and y values visible. This tool allows the user to quickly move through the data and examine any features of interest. The move tool is particularly useful when used in combination with the Zoom tool allowing the user to quickly inspect the data set while zoomed in without changing the scale of the plots. This tool can be used on all of the plot controls and does not affect the trajectory data.

Using the “Zoom” Tool

When the “Zoom” tool is selected from the plot tools (Figure 5C), the mouse click behavior for the plot controls changes to allow the user to redefine the plot’s extents allowing the user to quickly “zoom in” on regions of interest within the plot. The process of selecting new extents for a plot control is performed by first ensuring that the “zoom” tool is selected from the plot tools control and then clicking in one corner of the region of the plot that the user wants to enlarge. This will define the first corner of the plot’s new extents. After the first point has been selected the interface will indicate the selected region of the plot by drawing a rectangle between the location of the first click and the current position of the mouse. This assists the user in visualizing the area that will be contained within the resized plot. On the second click the plot’s extents will be resized to match the area of the defined rectangle. Alternatively this tools action can be canceled by clicking on any other control within trajectory Viewer or selecting a different tool from

the plot tools (Figure 5C). The extents of the plot can be reset by using the “Resize All” (Figure 5A) or “512x512” (Figure 6B) buttons. These tools reset the extents for all of the derived value plot controls and either adjust the extents of the trajectory plot to include all of the available data or to an area from 0,0 to 512x512 pixels respectively.

Selecting and Copying Trajectory Data from the Plot Controls

The “Select Data” tool in the plot tools control (Figure 5C) allows the user to select data from any plot control except for the velocity histogram and copy the matching positional data from the trajectory to the system clipboard. This function is useful for identifying behaviors in the trajectory based on calculated data such as total displacement and instantaneous velocity. This allows the user to rapidly identify and export regions of the trajectory that exhibit specific behaviors such as binding that may be difficult to determine from the Trajectory Graph itself. This tool uses the same two click interaction mechanism described above for the Zoom tool. The user first ensures that the “Select” tool option is highlighted under the plot tools (Figure 5C) and then clicks in one corner of the region they want to select. The user interface will then draw a rectangle between this first point and the current mouse position giving feedback as to the region being selected. On the second click the software will identify the points within the selected region on the plot. If the user has selected a plot other than the Trajectory Graph (Figure 4G) the software will then determine the points within the original trajectory that correspond to the selection and copy these points to the clipboard. If the user selected points from the trajectory plot the selected points will be copied to the clipboard. Finally Trajectory Viewer will prompt the user with the number of points in the original trajectory that were selected by this function.

In addition to the “select” tool, the data within any plot within Trajectory Viewer can be copied to the clipboard by using the right click menu. This menu is accessed by right clicking the plot containing the desired data and choosing the “Copy 1 Data set From Plot” or “Copy All Data from Plot” option from the menu. Both options will copy the data used to create the plot into the clipboard in a tab delimited format allowing the data to be copied into any spreadsheet software. It is important to note that currently the “Copy All Data from Plot” option will contain two copies of the data in the plot to the clipboard. The “Copy 1 Data set From Plot” option was created to address this problem and will send one copy of the data to the clipboard. However this option does not work correctly when multiple trajectories are viewed at the same time and should not be used under these conditions.

Delete Selected Data from the Plot Controls

The “Delete Selected” tool in the plot tools control (Figure 5C) allows the user to remove outliers in the trajectory data by selecting data from any of the plots except the Velocity Histogram (Figure 4E). This tool uses the same two click mechanic used above to select data. The user first ensures that the “Delete Selected” tool is highlighted under the plot tools (Figure 6C) and then clicks in one corner of the data they want to delete. The user interface will then draw a rectangle between this first point and the current mouse position giving feedback as to the region being selected. On the second click the software will identify the points within the selected region, these points are then used to identify and delete the matching points in the trajectory. As with any change to the trajectory data, the deletion of points triggers all of the plots to update reflecting the change and providing instantaneous feedback to the user. Any changes made to the

trajectory are automatically saved and require the user to use the “Save Track” button (Figure 5H) to record the changes. This tool prompts the user for the name of the modified trajectory and then saves the resulting file into the same path.

Qualifying Trajectories

Because the quality of single particle trajectories is dependent on many factors it is often necessary to review trajectories before they are used in further calculations. The original purpose of Trajectory Viewer was to provide the user the ability to rapidly visualize individual trajectories and their derived properties allowing the user to quickly evaluate the quality of individual trajectories, rapidly screen for desired behaviors within trajectories and identify and remove outliers within the trajectory.

The process of reviewing trajectories within Trajectory Viewer is performed by first selecting a list of trajectories to be reviewed as described in the Loading Trajectory Files section above. Once the user has successfully loaded the list of trajectories, the user can index through the list of trajectories using the “GoNext” (Figure 5D) and “GoBack” (Figure 5E) buttons in the control cluster. As the user reviews the trajectory files each trajectory can either be marked as good or bad by clicking the “Mark Good” (Figure 5F) or “Mark Bad” (Figure 5G) buttons. These buttons will move the trajectory into a sub folder labeled “Good” or “Bad” respectively and display the next file in the list of trajectories. This function allows the user to rapidly remove poorly tracked trajectories from the dataset before additional calculations are performed.

In addition to simply labeling trajectories good or bad Trajectory Viewer also includes tools to remove individual outliers present within the trajectory. As described above the “Delete Selected” tool allows the user to remove outliers by selecting data from any plot

in the software allowing the user to remove bad points reducing the level of noise in the dataset. This is particularly useful when reviewing trajectories that have had their trajectory further refined by intensity profile analysis methods. These trajectories often contain a few large outlier points resulting from bad fits to the intensity distributions of the tracked fluorescent object.

Performing Advance Calculations on Trajectories

In addition to Viewing and manipulating trajectories, Trajectory Viewer contains tools designed to assist the user in performing advance calculations on single particle trajectories. Currently Trajectory Viewer contains tools that assist the user in calculating the Mean Squared Displacement (MSD) and generating the NADA Displacement files used in combination with NADA Histogram Fitter to perform the NADA histogram analysis described in chapters 2 and 3.

Calculating Mean Squared Displacement

The most common method of determining mechanistic details from single particle motion trajectories is to determine the diffusion coefficient and alpha constant of the observed motion. This measurement is performed by calculating the mean squared displacement (MSD) as a function of window width for each trajectory in the dataset. The resulting MSD verses window width curves are fit to the equation $\langle r^2 \rangle = 4Dt^\alpha$ where $\langle r^2 \rangle$ is the measured mean squared displacement, D is the diffusion coefficient, t is the window width and α is the alpha constant^{8,9}. The parameters resulting from the fit can then be used to make a mechanistic interpretation of the data using a system specific model.

The calculation of the Mean Squared Displacement from individual trajectories is simply not intuitive to most scientists and is the most labor intensive step in determining the diffusion coefficient and alpha constant for individual trajectories within a dataset. In order to simplify and automate the calculation of the MSD from individual trajectories Trajectory Viewer includes functions to largely automate this process.

Calculating the MSD from a collection of trajectories using Trajectory Viewer is performed by loading a fresh instance of Trajectory Viewer and entering the calibration settings for the files that will be processed. The user then selects the “MSD Calculator Step 1” option from the Tools menu on the main window. This will generate a folder browser dialog allowing the user to select the directory containing the trajectories to be processed. This menu option generates a file containing all of the displacements between each point in the trajectory and a point a specified window width later in the trajectory measured at every possible window width ¹⁰. This is the same calculation performed to generate the NADA displacement files except in this case the calculation has been applied to individual trajectories instead of the whole data set as detailed below.

After the user has generated the NADA displacement files using the previous step. The NADA displacement tables need to be converted into Mean Squared Displacements. This is performed by selecting the “MSD Calculator Step 2” option from the Tools menu on the main window. This option will then prompt the user to select the directory containing the NADA displacement files generated in the previous step. After selecting the “MSD Calculator Step 2” option, Trajectory Viewer will generate a save file dialog so the user can specify the location of the output table. The software will then

open each NADA displacement file and calculate the Mean Squared Displacement as a function of window width. The resulting MSD versus window width data is then collected into a single table where the first column contains the window width in units of seconds and each successive column in the file contains the calculated MSD for an individual trajectory in the units of μm^2 . The file path of the NADA Displacement file used to calculate the MSD is noted in the column header for the purpose of bookkeeping. The resulting csv file then can be loaded into the analysis software of the user's choice and each column can be fit individually to the equation above to determine the D and α for each trajectory.

Generating NADA Displacement Files for use in NADA Histogram Fitter

In chapter 2 we developed a Non-Averaging Displacement Analysis (NADA) for determining the kinetics of switching between multiple dynamic states of motion for single particle trajectories. NADA measures the displacements within the trajectory at different window widths in a manner similar to the MSD method discussed above. Unlike the MSD calculation the measured displacements are not averaged and are instead plotted as a 2D histogram referred to as an NADA Histogram. This allows the different mechanical properties of the various states of motion to visually segregate into pure populations as discussed in chapter 2. Further by measuring the change in intensity of these pure populations as a function of window width then fitting the resulting measurement to an exponential, the lifetime of the individual states of motion within the dataset can be determined allowing us to extract mechanistic details from the measurement.

The process of performing NADA is split between two programs, Trajectory Viewer which produces NADA displacement files and NADA Histogram Fitter (discussed below) which uses the displacement files to visualize the NADA histogram and measure the change in intensity as a function of window width for the populations in the histogram. This measurement then can be used to determine the lifetime of the individual states of motion within the trajectories.

The first step in generating a NADA displacement file is loading a fresh instance of Trajectory Viewer and setting the calibration of the software to match the dataset being loaded. The user can then use either file opening method discussed above to load all the trajectories within the dataset simultaneously. This requires that the user select yes when prompted to "View all the data at once?" This loads all of the selected trajectories simultaneously providing Trajectory Viewer access to the whole dataset. Because loading large data sets can be both processor and memory intensive, it is suggested that the user deselect the "Draw Dots" option under the "View Options" menu before loading the files. Disabling this option will prevent the software from rendering individual dots reducing the number of GDI+ draws required to render each plot control. This can significantly increase the performance of Trajectory Viewer on systems with limited memory and CPU power.

Once all of the trajectories in the data set have been loaded the individual trajectories must be appended end to end (I.E. first point to last point) to create a single continuous trajectory. This is performed by selecting "Append Trajectories w/ Random Rotation" from the Tools menu. This menu option translates the first trajectory in the data set so that its first point is located at 0,0. This trajectory is then randomly rotated

around its first point. The second trajectory is then translated so that its first point is located at the same position as the last point in the first trajectory. The second trajectory is then randomly rotated about its first point and the overlapping point is removed. This process of appending trajectories is then continued with each loaded trajectory until the software has created a single effective trajectory for the whole dataset.

Once the trajectories have been appended the user can then process the whole data set at once. This is performed by selecting the "Build NADA Displacement File" menu option under the Tool menu. This option will generate a standard save file dialog allowing the user to set the file name of the generated NADA Displacement File. By default the displacement file's filename should end with "_lag.csv." The user will then be prompted to establish limits to the amount of data used in the calculation of the displacement file. These limitations enable the files to be calculated on computers with limited ram. In order to accommodate these limitation methods the generation of the NADA displacement file was designed specifically to minimize any possible effect these limits would have on the NADA Histogram and the measurement of state lifetimes.

The first prompt asks the user to set the maximum window size in units of seconds. This prompt is prepopulated with the maximum possible window width calculated from the loaded dataset and allows the user to limit the maximum window width in the generated NADA displacement file. Because the change in intensity verses window width used to measure the lifetime of the states of motion present within the NADA histogram occurs at small window widths, limiting the maximum window width in the NADA displacement file should have little to no effect on the measured state lifetimes provided that the user chooses a maximum window value larger than the time required

for the intensity of the pure populations to completely decay. The second prompt asks the user to set the maximum number of data points in the NADA displacement file. This prompt establishes an upper limit for total number of points processed into the NADA displacement file. This modification was necessary to prevent the software from crashing when processing extremely large datasets on computers with limited ram. Because the NADA displacement file is generated from small to large window widths any effect on the measurement of lifetimes will be limited to the last window width of the NADA displacement file.

After the user has answered these prompts Trajectory Viewer will then perform the calculations necessary to generate the NADA displacement file. Because NADA displacement files can take several minutes to generate the progress and estimated time till completion is indicated in the status bar (Figure 4F). Once Trajectory Viewer has completed generating the NADA displacement file, the file can be opened in NADA Histogram Fitter as described below. The NADA Histogram Fitter program allows the user to visualize the NADA histogram and perform the measurements needed to calculate the lifetimes of the states of motion within the NADA histogram.

NADA Histogram Fitter

The NADA Histogram Fitter program is designed to provide a graphical interface for visualizing NADA histograms, tools for the measurement of lifetimes from populations of intensities within the NADA histogram, and the generation of graphics for publication. The interface for the NADA Histogram Fitter consists of a single window (Figure 6) which is divided into several functional components. The functions of most of these components are straight forward and summarized in Table 11 and 12. However the

NADA Histogram Control (Figure 6C) is complex and requires a more detailed discussion.

The NADA histogram control (Figure 6C) is used to visualize the NADA histogram and provides tools for the user to interact with data. This control is composed of a drawing surface (Figure 7A), vertical and horizontal scroll bars (Figure 7B and C), a magnification control (Figure 7D) and a “Click to Read” button (Figure 7E). The most prominent feature in this control is the drawing surface (Figure 7A) which is used to display a graphical representation of the NADA histogram (Figure 7F) as well as various other indicators (Figure 7G-I).

The primary function of this control is displaying the NADA histogram data as an image. In this image each bin of the NADA histogram is represented as a pixel with window widths increasing from left to right and displacement increasing from bottom to top at the rate of one pixel per bin. The red channel intensity of each pixel is proportional to the number of observations within that bin as described in chapter 2. The scroll bars (Figure 7B and C) and magnification control (Figure 7D) control the size and position of the histogram image within the drawing surface allowing the user to zoom in and observe small features within the histogram. In addition to the NADA histogram this plot contains other information for the user such as the Edge of Data Indicator or (green dots, Figure 7 G) which indicate the upper edge of the distribution of displacements at each window width or the reading lines (Figure 7H-I) which indicate the regions within the NADA histogram which are “read” to determine the lifetime of the directed (blue lines, Figure 7H) and tethered populations (green lines, Figure 7I) within the NADA histogram. The drawing surface also responds to mouse movements as well

as left and right click events. As the user moves the mouse over the drawing surface, the mouse's position is displayed in the taskbar in both pixels and scaled units. Left mouse clicks can be recorded by clicking the "Click to Read" button (Figure 7E) located at the bottom of the NADA histogram control. After clicking the button the user can left click directly on the image to record the location of the click to the textbox in the Log tab. A second click on the drawing surface will record the position of the click and report the slope and y axis offset of a line connecting both clicks. This allows the user to directly measure the position of populations within the NADA histogram expediting the process of measuring lifetimes from the NADA histograms.

Opening NADA Histogram Fitter

When opening NADA Histogram Fitter, the software will locate and load the last saved settings from the "Settings.csv" file in the executable's folder. This includes values such as the last folder opened and the settings for the NADA histogram image binning. If the user is having trouble loading the software it is recommended to delete this file. When the program is reloaded the software will not be able to locate the settings file and will regenerate the setting file using the default settings hard coded into the software. This is helpful if corrupt or invalid settings are preventing the software from loading. It is also possible to change the maximum size of the generated NADA histogram image by changing "MaxCellsX" and "MaxCellsY" settings to better accommodate the generation of images on systems with limited ram.

Loading NADA Displacement Files

The first step in analyzing data using the NADA Histogram Fitter software is loading data from the NADA displacement file that was generated by Trajectory Viewer (see

above). This operation is performed by selecting the “File->Open File” option from the menu bar (Figure 6B). This menu item creates a standard windows open file dialog that allows the user to select the desired NADA displacement file. After the user selects the NADA displacement file and clicks ok. The software reads the file and converts the text values into doubles for the internal data table. For large datasets the conversion process can take a significant amount of time. Because of the long loading time the title bar is updated every 10^4 lines to reflect the ongoing loading progress to the user. The loading progress is also indicated in the status bar at the bottom of the form.

Following the parsing of the NADA displacement file the individual displacements are compiled into an NADA histogram and the filename will appear in the status bar's pull down menu (Figure 6E). The NADA histogram is then visualized as an image with each pixel in the image representing a single bin with increasing window width along the x axes from left to right and the displacement values on the y axis increasing in value from the bottom of the image to the top. The intensity of each pixel from 0 to 255 on the red channel represents the number of counts in each bin normalized at each window width (vertical columns). Bins which contain no counts are represented as black and the blue regions indicate the edge of the data. (chapter 2) Once the data file has been loaded then the software can be used to analyze the NADA histogram using the other work flows described below.

Adjusting the Binning Settings of the NADA Histogram Image

After the NADA displacement file has been loaded, (See “***Loading NADA Displacement File***” above) it is often necessary to optimize the binning of the NADA histogram. This is performed using the "Create Histogram Image" option under the

Redraw menu. This tool adjusts the binning parameters of the NADA histograms and allows the user to choose between two different drawing methods. When the user selects the menu option they are prompted to select a new “Window Width bin Size (s)” and a “Displacement bin size (μm)” which corresponds to the bin interval for the window width in seconds on the x axis of the image and the displacement binning interval in μm on the Y axis of the image.

The final prompt asks whether to “Use Subpixel Rendering” to build the histogram or use direct bin to pixel mapping. Selecting “yes” enables the subpixel rendering method, where each data point being binned splits its contribution to the final histogram between four adjacent bins based on the point’s proximity to the centers of each bin. The subpixel rendering method reduces the dependence of the final image on the specific binning properties and significantly speeds up the calculation of the NADA histogram. If the user selects “No” the bins are calculated using hard cut offs and each single point is counted in a single bin regardless of the proximity of the bin to the edge conditions.

When adjusting the window width binning interval, the user must be aware that a binning interval less than the sampling rate of the displacement file can result in empty bins which appear as vertical black bars in the histogram image. Empty bins in the histogram slow the performance of the software and complicate later steps in the process such as image filtering. If these bars are present in the image the user should increase the window width binning interval until they disappear. Ideally the window width interval should equal the inverse of the frame rate ($1/\text{FPS}$) used to generate the displacement file however I have found that it is often necessary to slightly increase the window width binning interval to prevent a banding artifact. It may also be helpful to turn

of subpixel rendering while adjusting the window width binning interval because this rendering method can obscure the presence of empty bins within the image.

The displacement binning interval ideally should be adjusted as a function of your expected positional resolution. Ideally trajectories captured using centroid tracking methods should have a resolution of approximately 1/10 the pixel size. Using our equipment (100x objective with the 1.5x multiplier and the Roper camera), I would recommend using a displacement bin size of 0.1 μm per bin. For data which has had additional enhancements performed on it, such as intensity profile refinement by non-linear curve fitting, smaller bin sizes such as 0.05 μm -0.01 μm per bin can be used.

In general the user should use their own judgment when choosing the bin intervals and select bin intervals small enough to prevent the consolidation of multiple real populations yet large enough to minimize the effect of error in the measurement on the NADA histogram. It is important to remember all measurements are taken from the NADA displacement file directly and so binning and other modifications to the histogram image do not affect the lifetime measurements.

Filtering the NADA Histogram Image to Increase the Contrast of the Populations within the NADA Histogram

In addition to adjusting the binning settings of the NADA Histogram (see above) it is possible to use external imaging software such as IrfanView³, ImageJ⁴ or Adobe Photoshop to adjust the contrast and brightness of the image or to apply filtering to increase the contrast of any populations present in the NADA histogram image.

The NADA histogram image can be externally filtered after the image has been generated. This is performed by copying the image from the NADA Histogram Control

(Figure 6C) by right clicking the Histogram Control and choosing "Copy whole image" from the right click menu. This option sends the current NADA histogram image to the system clipboard allowing the user to paste the image inside of the imaging software of their choice. Once the image has been filtered it can be transferred back to the NADA Histogram Control by copying the filtered image to the system clipboard and then right clicking the Histogram Control in NADA Histogram Fitter and selecting the "Paste Image" option from the right click menu. This will replace the existing NADA image in the NADA Histogram Control but will not modify the underlying data used to determine population lifetimes. Please note that the pasted image may not be visible until the histogram control is updated by clicking one of the scrollbars or making the window lose focus. The original NADA Histogram can be reloaded by clicking "Revert Image" in the NADA Histogram Control's right click menu or using the "Create Histogram Image" option in the Redraw menu.

When filtering the NADA Histogram image it's recommended for the user to begin by adjusting brightness and contrast levels to maximize the visibility of the populations in the image. This is particularly useful in histograms where a single type of motion dominates the histogram resulting in the intensity of less populated states of motion to become dim. Alternatively more advance methods like edge detection filters such as ImageJ's "Find Edges" filter work well for detecting Directed and Tethered populations. This increased detection of Directed and Tethered populations is because the ImageJ's "Find Edges" filter, a 3x3 Sobel image filter, increases the brightness of transitions within the image and so highlights the transition between the pure population and mixed populations in the NADA histogram.

Obtaining Lifetimes from the NADA Histogram

As described in chapter 2, the NADA method can measure the lifetime of the individual states of motion within the NADA histogram by identifying the populations of intensity generated by the states of motion within the trajectory, measuring the change in intensity of these populations as a function of window width, and fitting the resulting measurement to an exponential decay. The time constant from the fit is equal to the lifetime of the population. NADA Histogram Fitter facilitates this measurement by providing tools to define populations within the NADA histogram and measure the intensity of the NADA histogram as a function of window width. The resulting measurement is displayed in a graph which can be used to export the intensity measurement data read from the NADA histogram to a third party programs such as Origin to perform the exponential fit. Because each state of motion generates displacements based on its mechanical properties, the distribution of intensities within the NADA histogram resulting from each state of motion has a different shape dependent on the type of motion that generated it. In order to measure the change in intensity as a function of window width for different types of motion, NADA Histogram Fitter contains separate tools to measure the lifetime of Directed, Tethered, and Diffusing states of motion. These tools are located on individual tabs within the Task Tabs Control (Figure 6D) and contain tools designed to read the population of intensities within the NADA histogram generated by specific types of motion.

Measuring the Lifetime of the Directed State of Motion

The first step in measuring the lifetime of a state of motion is to locate the population of intensities that corresponds to the state of motion's expected

displacements within the NADA histogram. As discussed in chapter 2, populations of intensity generated by directed states of motion within the NADA histograms appear as diagonal regions of intensity with slopes equal to the average velocity of the directed population. If the diagonal populations are not immediately visible within the NADA histogram the user may need to adjust the binning size or filter the image as discussed above to make the directed populations more visible. Once the directed intensity has been located within the NADA histogram, the change in intensity of the directed population can be measured as a function of window width. This measurement is performed using the “Directed Population” tab within the Task Tabs Control (Figure 6D).

The “Directed Population” tab is designed to allow the user to define a region of the NADA histogram between two diagonal lines and measure the intensity of the defined region as a function of window width returning the results as exportable data in the plot control. The tab consists of a centralized plot that displays and exports the measured change in intensity between the two defined lines, four text boxes that define upper and lower bounds of the population as lines using slopes and y intercepts (y_0), a “Read” button that measures the intensity calculated from the raw data which falls between the two lines and two checkboxes that filter the measured data. The functions of the controls within this tab are described in detail in Table 15.

The measurement of the directed population is performed by defining the slope and y intercept of the two lines to contain as much of the diagonal intensity as possible while minimizing any harmonic repeats or tethered regions and then clicking the “Read” button. This will measure the intensities located between the lines in the NADA histogram and return the measurement to the plot control. In addition the Read button

provides feedback to the user by marking the region of the NADA histogram used in the measurement by drawing the two limiting lines on top of the NADA histogram. After the data appears in the plot it can be exported by right clicking the plot and selecting the "Copy Data to Clipboard" option from the right click menu. This will copy the data in the plot to the system clipboard allowing the data to be fit to an exponential decay using third party software to determine the lifetime of the population.

In order to simplify the definition of the directed population, NADA Histogram Fitter includes the "Click to Read" button on the NADA Histogram control (Figure 7E). This tool allows the user to click on the NADA histogram directly to record the mouse's current position in the NADA histogram. The first click will copy the click's position within the NADA histogram to the log Tab. The second click will calculate the slope and y-offset of a line joining the first and second clicks and report the resulting slope and y-offset to the log tab. This function allows the user to draw the edges of the directed and tethered population for determining the lifetimes of these populations.

Measuring the Lifetime of the Tethered State of Motion

The measurement of the lifetime of a tethered state of motion is similar to obtaining the directed state of motion's lifetime. The process consist of locating the population within the NADA histogram, defining the region of the NADA Histogram to measure, measuring the change in intensity as a function of window width and fitting the resulting decay to obtain the lifetime of the measured population. However as described in chapter 2, tethered states of motion do not increase their total displacement as a function of window width, instead the tracked particle fluctuates around a fixed location. The implication of this type of motion is that a tethered state of motion appears as a

horizontal population with a width approximately equal to the average excursion length in the NADA Histogram.

As with the measurement of the directed state's lifetime, the first step in measuring the lifetime of the tethered population is localizing the population within the NADA Histogram. In general Tethered populations are significantly easier to visualize and measure than directed populations because the tethered populations occur at the bottom of the NADA histogram and run parallel to the X-axis of the NADA histogram. However if the population is not easily identified, the same methods discussed for detecting directed populations such as adjusting the histograms bin sizes and image filtering are also applicable to detecting the tethered populations. Once the tethered population has been located within the NADA histogram, the change in intensity of the tethered population can be measured as a function of window width. This measurement is performed using the "Tethered Population" tab within the Task Tabs Control (Figure 6D).

The "Tethered Population" tab is designed to allow the user to define a region of the NADA histogram between two horizontal lines to measure the intensity of the defined region as a function of window width and return the results as exportable data in the plot control. The tab consists of a centralized plot that displays and exports the measured change in intensity between the two defined lines, two text boxes that define upper and lower bounds of the population as horizontal lines, a "Read" button that measures the intensity calculated from the raw data which falls between the two lines and two checkboxes that filter the measured data. The functions of the controls within this tab are described in detail in Table 15.

The measurement of the tethered population is performed by defining the maximum and minimum displacements of the horizontal population and then clicking the “Read” button. The program will then read the intensities located between the maximum and minimum displacements in the NADA histogram and return the measurement to the plot control. In addition the Read button provides feedback to the user by marking the region of the NADA histogram used in the measurement by drawing the two limiting lines on top of the NADA histogram. After the data appears in the plot it can be exported by right clicking the plot and selecting the “Copy Data to Clipboard” option from the right click menu. This will copy the data in the plot to the system clipboard allowing the data to be fit to an exponential decay using third party software to determine the lifetime of the population.

As discussed above the "Click to Read" button on the NADA Histogram control (Figure 7E) can be extremely helpful in quickly determining the position of the tethered population within the NADA Histogram. This function allows the user to click on the NADA histogram to record the position of the click to the log tab. This simplifies the process of determining the maximum and minimum displacements of the tethered population by allowing the user to simply click above and below the tethered population to measure the limits of the tethered population's location.

Measuring the Lifetime of the Diffusing State of Motion

The process of measuring the lifetime of a freely diffusing state of motion is similar to the methods used to measure the lifetime of the other states of motion described above. The process consist of locating the population within the NADA histogram, defining the region of the NADA Histogram to measure, measuring the change in intensity as a function of window

width and fitting the resulting decay to obtain the lifetime of the measured population. As described in chapter 2, a diffusing state of motion increases its displacement approximately equal to $\sqrt{(4Dt)}$, where D is the diffusion coefficient and t is the window width. These results in a complex distribution of intensities within the NADA histogram centered around $\sqrt{(4Dt)}$ often with a wide distribution of displacements making the visualization of the diffusing population difficult. As a result the Diffusing Population Tab contains a tool for calculating the location of the diffusing population from the distribution of displacements at each window width within the NADA histogram.

The “Diffusing Population” tab is designed to assist the user in defining the region of the NADA histogram that contains the diffusing population and measure the intensity of the population as a function of window width. Unlike previous tabs which required the user to define the location of the population within the NADA histogram, the Diffusing Population tab calculates the center of the population within the NADA histogram and returns the intensity within the specified “Read Width” as a function of window width from the calculated center to the plot control. The tab itself consists of a centralized plot that displays and exports the intensity of the population as a function of window width, two text boxes that define the “Read Width” and top of the tethered population and two buttons. The “Read” button calculates the location of the diffusing population from the distribution of displacements at each window width within the NADA histogram and the “Use Existing Reading Path” button that allows the user to define a custom reading path to read the intensity of the NADA histogram. The functions of the controls within this tab are described in detail in Table 15.

The process of measuring the lifetime of the diffusing population is performed by first locating the tethered population within the NADA histogram as described above. The upper limit of the tethered population or “dmax” in μm should be entered into the “Only Consider Values greater than” textbox. The next step is to enter a “read width” value into the matching textbox. This will be the distance in μm above and below the calculated path to include in the measurement. Currently there is no established method of determining the appropriate read width for a diffusing population. However the number should be large enough to minimize noise in the measurement yet small enough to minimize the inclusion of other rates. In chapter 2, we used a read width equal to half of the tethered population’s apparent width. Using this value we were able to measure the lifetime of the simulated diffusing populations with good precision. Once the user has set the values in the two textboxes the user can click the “Read” button. This tool will calculate the center of the distribution of displacements while excluding the specified small displacement values created by the tethered populations within the NADA histogram. The user will be prompted to save this reading path for use in generating figures or for repeating the measurement using the “Use Existing Reading Path” tool. Using this calculated reading path the program will then read the intensities of the NADA histogram within the "Read Width" in μm as a function of the window width reported in the plot control. After the data appears in the plot it can be exported by right clicking the plot and selecting the “Copy Data to Clipboard” option from the right click menu. This copies the data in the plot to the system clipboard allowing the data to be fit to an exponential decay using third party software to determine the lifetime of the population.

It is important to note that when fitting the intensity of the diffusing population as a function of window width, the population's intensity rarely goes to zero unlike the directed and tethered states of motion. This is because as the diffusing population decays it overlaps with the intensity of the blended populations resulting in a non-zero value. This requires that the user find a rigorous method to limit the amount of diffusing population's intensity data at higher window widths. In chapter 2 all of the intensity diffusion measurements decayed until they reached a steady value before becoming chaotic. This region of steady values was used to set the maximum window width used when fitting the decay to determine the lifetime of the diffusing population.

Visualizing the Distribution of Displacements at Each Window Width (Vertical Read Tab)

In the NADA Histogram Fitter program the distribution of displacements or velocities at any given window width can be quickly visualized using the tools in the "Vertical Read Tab" in the Tasked tabs controls (Figure 6C). The functions of these controls are explained in detail in Table 15.

The measurement of the distribution of displacements at any window width can be performed after a data file has been loaded in the software and the NADA histogram image has been generated. The measurement of the distribution of displacements is performed by selecting the "Vertical Read Tab" from the task tabs control and selecting a window width from the Scroll Bar located at the bottom of the tab. If the plot is empty it may be necessary to activate the measurement by moving the slider in the scroll bar. After each change to the Scroll Bar the software reads the intensity of the NADA histogram image at the selected window width and returns the distribution to the graph.

The selected window width is indicated on the NADA histogram as a vertical teal line and at the top of the graph in the “Vertical Read Tab”. The rescale button on the top of the tab adjusts the range of the Scroll Bar to match the range of the NADA histogram image. The data in the plot can be exported by right clicking and selecting “Copy Data to Clipboard” which copies the data as a tabbed table which is compatible with most spread sheets and text editors or by clicking “Send to Origin” which sends the data directly to Origin using their published API.

In addition to reporting the distribution of displacements within the NADA histogram, the “Vertical Read Tab” can report the distribution of velocities at each window width by checking the "Report Velocity" checkbox. This option converts the measured displacements in the plot to effective velocities by dividing the distribution of displacements by the current window width.

Exporting Calibrated NADA Histogram Images for the Generation of Figures

One of the key features of NADA Histogram Fitter is the visualization of displacements within the NADA histogram. NADA Histogram Fitter provides tools not only for dynamically interacting with the NADA histogram image but for exporting calibrated versions of the NADA image to generate figures for publication. These calibrated images can be exported from NADA Histogram Viewer at any point following the generation of the NADA histogram image using the “Copy selection from image” option in the Histogram Control’s (Figure 7) Right Click Menu is accessed by right clicking on the NADA Histogram Control. After selecting this option the user is prompted to select the region of the NADA histogram to export by specifying the maximum window width in seconds and maximum displacement in μm . The software will export

the portion of the NADA image containing the region of the image from the 0 second , 0 μm bin to the specified maximum to the clipboard. If the selected maximums exceed the limits of the NADA histogram the software will fill the empty portion of the image with a grey background denoting the empty region. This image then can be used to generate a figure containing the NADA Histogram by creating a graph with the same ranges used in the generation of the images and scaling the calibrated image to fill the plots area. This method was used in Chapters 2 and 3 to create the figures containing NADA histogram images.

Closing the NADA Histogram Fitter and Saving Current Settings

On closing NADA Histogram Fitter, the program will prompt the user to save the current settings. If the user clicks yes the software will create a file named "Settings.csv" in the executable's folder containing the current values such as the last folder opened and the settings for the NADA histogram images binning. These settings will automatically be applied when next opening the software. This functionality is useful when analyzing data sets containing many conditions because it eliminates the need to enter settings multiple times. If the software is not loading files correctly or crashing deleting this file will restore the built in values. Alternatively the settings file can be modified manually or archived as a means of documenting the settings used when performing the analysis.

Summary

In this chapter we introduced the basics of single particle tracking and provide a user manual for the three programs used to perform the NADA and MSD measurements within this dissertation. The first program Particle Tracker is a robust tool

for single particle tracking of fluorescent labeled molecules within video microscopy images. In this chapter we limited the scope of our discussion of Particle Tracker to its use in performing centroid based single particle tracking, the super resolution refinement of the centroid positions and exporting the resulting trajectories for use in Trajectory Viewer or other software packages. The second program discussed was Trajectory Viewer. This program is focused on the rapid visualization of single molecular trajectories and performing the advance calculations necessary for generating the NADA Histogram and obtaining D and alpha constants from single particle trajectories. The third and final program discussed in this chapter is NADA Histogram Viewer. This program provides a simple graphical interface for the NADA method, generates NADA Histogram Images from the files generated by Trajectory Viewer, and provides tools to measure the change in intensity as a function of window width for individual populations of motion present with in the NADA histogram. These measured changes can then be fit to exponential decays to determine the lifetimes of the states of motion within the single particle trajectories.

Figures

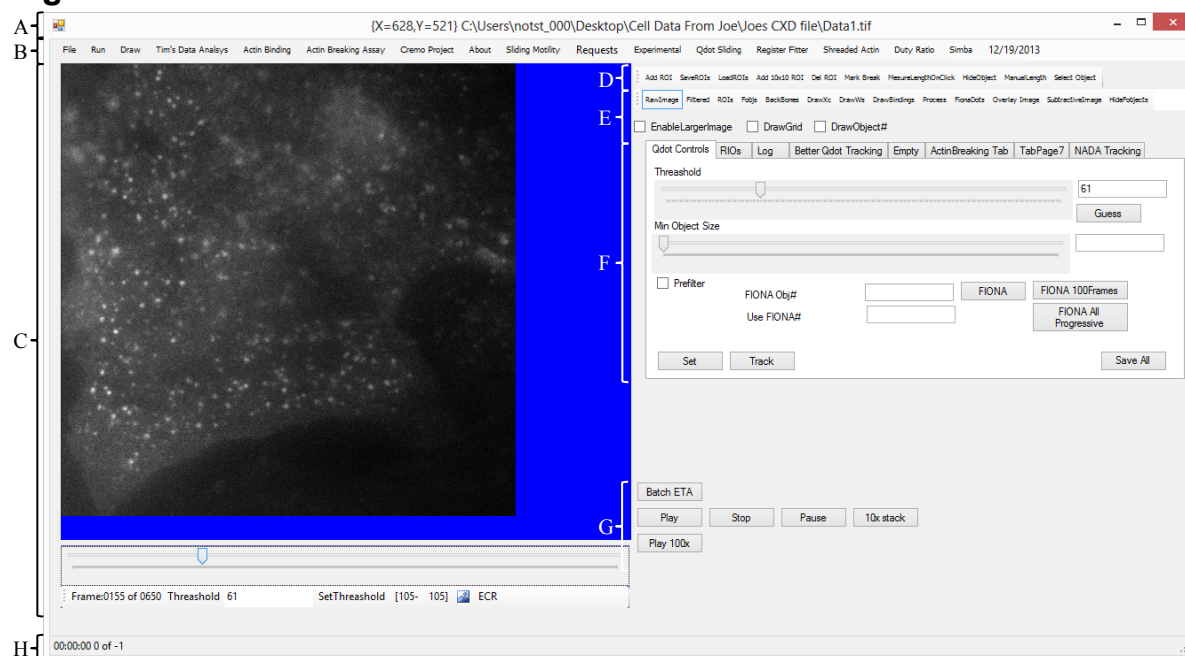


Figure 1 – Particle Tracker’s Main Window

The main window is composed of several controls that provide access to the functions within the software. These functions are summarized in Table 1.

Component	Function
The Title Bar (Figure 1A)	Reports the current loaded file and reports the mouse’s position as the user moves the mouse over the displayed image.
Main Menu Bar (Figure 1B)	The menu provides access to the major functions within the software. These functions are listed in detail in Table 2.
Screen and Playback Controls (Figure 1C)	Displays the images to the user based on the layers selected in the drawing layer tool bar and provides tools for selecting the current frame and frame range to be analyzed. The layers are further described in Figure 2 and Table 5.
Image Interaction Tool Bar (Figure 1D)	Contains a series of tools which allow the user to interact with data displayed on the screen control. The functions of these tools are described in Table 3.
Drawing Layer Tool Bar (Figure 1E)	Enables and disables individual drawing layers allowing the user to visualize different aspects of the analyses process. The layers and their behaviors are summarized in Table 4.
Task Tabs (Figure 1F)	Tabs containing grouped controls designed to address specific workflows within this software. The functions of these controls are described in Table 6.
Other controls (Figure 1G)	Item Function
	Batch ETA Displays the estimated time remaining for batched tasks.

	Play	Displays each frame in the image stack one frame at a time.
	Stop	Stops the current task or playback.
	Pause	Pauses the current task or playback.
	10x Stack	When this button is clicked the user is prompted for the number of frames to add at a time. The function then plays the selected range of frames by calculating the sum of the specified number of frames following from current frame and displays the resulting image on the “Process” drawing layer.
	Play 100x	Plays every 100 th frame in the image stack. This is useful for spotting slow motion in high frame rate movies.
Status Bar (Figure 1H)	Displays instructions to the user and estimated time till completion for individual tasks.	

Table 1 – Description of Particle Tracker’s Controls

Location	Menu Item	Function
File		Holds functions for Opening and Saving files.
	Open > Tiff	Loads a multipage Tiff file (*.tif).
	Open > Data File	Loads a data file (*.dat) generated my Particle Tracker.
	Open > Stanford Camera	Loads video composed of multiple tiff images. Such as the ones generated the XR/turbo G intensified camera manufactured by Stanford Photonics Inc.
	Open > Multipart Save	Loads a multipart data file (*.mps) generated by Particle Tracker.
	Save > Data File	Converts the programs memory to a file allowing the user to save their progress in processing a dataset and reload that progress at a later point.
	Save > Tiff as Gif	Export the range of images select in the playback controls as individual files.
	Save > Displayed Image	Saves the currently displayed image.
	Save > Export Pixels	Generates a movie showing the motion of the selected tracked objects using the layers selected in the Drawing Layer Tool Bar.
	Save > Multipart Data File	Saves the current programs memory as multiple files. Because the save and load times are significantly faster, this function is useful for saving large datasets.
	Clear	Removes any tracking data from memory.
	Exit	Quits the program.
Run*		Contains legacy functions from the development of the software.
Draw*		Contains legacy functions from the development of the software.
Tim's Data Analysis*		Contains tools for analyzing the movement of fluorescent beads embedded in a flexible matrix displaced by individual

	cardiac myocytes cultured on a patterned surface with nanometer resolution.
Actin Binding*	Contains tools for detecting periods of myosin interaction with actin filaments using the localized damping of the lateral dynamics of the actin filament. The dynamics are reported by obtaining the position of the actin filament perpendicular to the backbone of the filament in nanometer resolution.
Actin Breaking Assay*	Tools for the automated detection of actin breaking as a reporter of intra-actin strain.
Crema Project*	Tool for detecting the multiplicity of Q-dot labels attached to MLCK molecules by quantizing the intensity of blinking of individual fluorescence sources within the image.
About	Displays a message box with my email address.
Sliding Motility	Tools used for fully automated analysis of sliding motility.
	“w\Gorga et al. 2003 Filtering”
	A tool for the complete automation of the analysis of <i>in vitro</i> motility assays with filtering to qualify trajectories as moving based on the data filters described in Gorga et al 2003 ¹¹ .
	Go
	A tool for the complete automation of the analysis of <i>in vitro</i> motility modeled after the description of automated analysis by J. Klinth ¹² .
Requests*	Collection of requested tools by co-workers to accomplish specific tasks.
Experimental*	Collection of tools still under development or that lack sufficient testing to confirm proper function.
Qdot Sliding*	Tools for measuring the motility of small fluorescent probes on actin.
Register Fitter	Performs the operating system command to install the external non-linear curve fitter required by several routines in Particle Tracker.
Shreaded Actin*	Tool for detecting periods of myosin's interaction with the actin filament using the localized damping of the dynamics of the actin filament by determining the position of localized labels on the actin filament in two dimensions as a reporter for myosin binding.
Duty Ratio	A collection of tools for determining the duty ratio of myosin by measuring and comparing the length of actin filaments with the velocity of the observed directed motility.
Simba*	Tools for quantifying Simba (Simple individual myosin binding assay) movies. These tools quantify the movement of diffusing shredded actin filaments interacting with a low density of myosin attached to the surface to detect periods of binding (I.E. periods of low dynamics) as a reporter of myosin binding.
Last Data Updated	Displays the date of the last major update.
*Experimental functions or functions which exceed the scope of this dissertation.	

Table 2 – Description of Particle Tracker's Main Menu Items

Item	Function
------	----------

Add ROI	Allows the user to create a rectangular ROI (Region of Interest) by drawing a box around the region on the screen control. The first click starts drawing the rectangle. The second click finishes the rectangle. The time range for the ROI is set by the start and stop frame selected in the playback controls.
SaveROIs	Save the ROI to a file.
LoadROIs	Loads the ROI from a file
Add 10x10 ROI	Creates a 10x10 pixel ROI
Del ROI	Deletes a ROI by clicking on it.
Mark Break	Marks an event in the movie. Event times can be exported using the Export Breaks button on the Actin Breaking tab.
MeasureLengthOnClick	After filaments have been tracked, this tool measures the length of fluorescent objects by clicking on them. This is performed by fitting the locations of the pixels describing the fluorescent object to a 4th order polynomial line. Using the resulting fit equation the distance along the curve limited by the bounding box of the object is measured and returned to the user.
HideObject	After filaments have been tracked, this tool hides objects clicked on by the user by painting the object pixels blue on the HideFobjects layer.
ManualLength	<p>Allows the quick measurement of lengths within the image field by drawing lines.</p> <p>This tool functions by clicking on images displayed in the screen control. If the user clicks on the image without holding shift a new measurement will be created. The measurement is continued by holding shift down and clicking on the screen control. Each click adds a node to the current measurement line. The resulting distance is reported as the sum of the lengths between the nodes in the measurement ROI's path and is displayed on screen next to the measurement line and in the RIO tab in the Task Tabs control.</p>
Select Object	Selects a current object and displays the trajectory of the selected object on the ROI drawing layer.

Table 3 – Description of Particle Tracker's Image Interaction Tool Bar

Item	Function
RawImage	Draws the input images at the frame selected by the track bar in the playback controls. By default the contrast of the image is expanded to enable viewing of dim images.
Filtered*	Used to display the intermediate steps in the filtering of images for later processing.
ROIs	Displays the ROI's on top of the Raw image.
Fobjs	Draws green lines around the detected fluorescent objects after centroid tracking. This tool is best used in connection with the "RawImage" layer. When an object is larger than 5000 pixels the object area is indicated by green area to reduce drawing time. If an object shares pixels with the sides of the image the objects is drawn as a tan filled area to indicate

	that it has been excluded from the analysis.
BackBones	Draws the resulting line from the fit to determine the length and orientation of fluorescent objects. Requires that a length measurement or orientation measurement of fluorescent objects has been performed. This tool is best used in connection with the RawImage layer.
DrawsXc's*	Draws the perpendicular center of the actin filament obtained by intensity profile refinement of the perpendicular intensity profile of the filament measured along its length.
DrawWc*	Draws the width of the actin filament obtained by intensity profile refinement of the perpendicular intensity profile measured along the length of the actin filament.
DrawBindings*	Displays the location of possible binding events from the ActinBinding and Simba tools.
Process	Displays images generated by the image processing algorithm on top of all other images.
FionaDots	Displays the resulting center position from the intensity profile enhancement of the center position of fluorescent objects.
Overlay Image	Displays a static image in the red channel of the screen. This tool is useful for creating overlaid images of two different fluorescent channels.
SubtractivelImage*	An incomplete function to highlight motion in an image by subtracting the previous frame's image from the current frame.
HideFobjects	Draws black on top of objects based on the object numbers in the HideObjectList text box. This tool is intended to assist the manual analysis of images by marking objects that have already been measured.
HideObjectList	A list tracked objects that are hidden by the HideFobjects command.
EnableLargerImage	Shows and hides the Enlarged View Window (Figure 3).
DrawGrid	Draws a grid over the screen.
DrawObject#	Draws the internal reference number for the tracked objects next to the object. Requires the Fobj layer to be enabled.
*Experimental functions or functions which exceed the scope of this dissertation.	

Table 4 – Description of Particle Tracker Drawing Layer Tool Bar

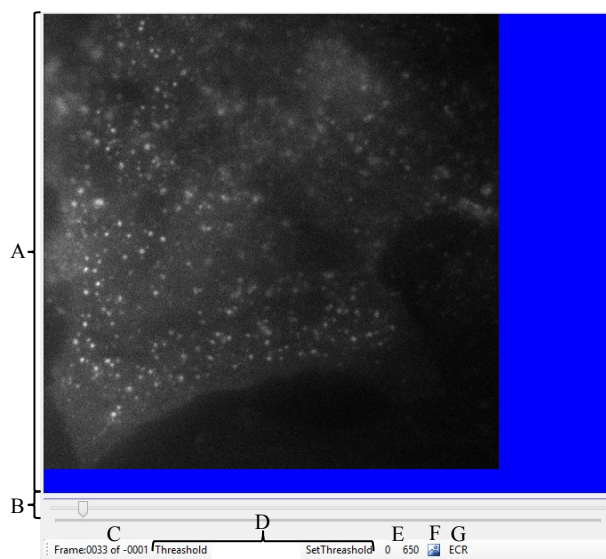
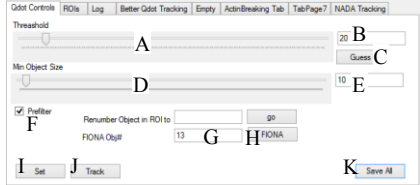


Figure 2 – Screen and Playback Controls

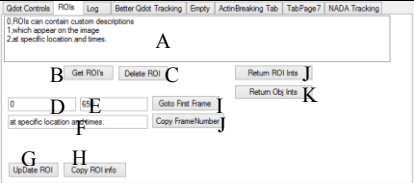
A collection of custom controls that functions both as a display for the composite image composed of the layers chosen in the “Screen and Playback Controls (Figure 1C)” and as an interface for the location dependent tools in the “Image Interaction Tool Bar” (Figure 1D).

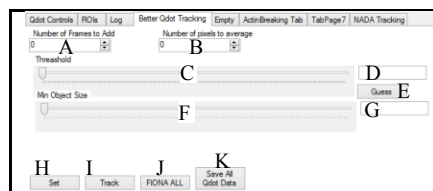
Component	Function
Screen (Figure 2A)	Renders images based on the layers chosen in the Drawing Layer Tool Bar (Figure 1E).
Frame Selection Bar (Figure 2B)	Selects the current frame to be displayed.
Current frame display/Last Frame Display (Figure 2C)	Displays the current frame and the total number of frames that have data associated with them. (“-0001” denotes that no data is currently associated with the loaded image.)
Custom threshold tools (Figure 2D)*	A set of tools for adjusting the threshold for object detection in each frame.
Frame Range Selection Buttons (Figure 2E)	Two buttons for setting the range of the data to be analyzed. The left hand button sets the lower limit and the right and button sets the upper limit. Values for these controls are selected by using the Frame Selection Bar to select a frame and then clicking on either of the limit controls to set the limit.
Frame Selection Bar Enable/Disable Button (Figure 2F)	Enables and disables the selection bar.
ECR (Exceed Contrast Ratio, Figure 2G)*	Allows the displayed image contrast to exceed the normal maximum contrast range allowing more contrast to be used on the lower intensity portion of the image.
*Experimental functions or functions which exceed the scope of this dissertation.	

Table 5 – Description of Particle Tracker’s Screen and Playback Control’s Controls.

Tab Name and Image	Controls and function								
<p>Qdot Controls</p> 	<p>Controls for tracking and refining the position of single fluorescent sources such as individual quantum dots attached to proteins in nanometer resolution.</p> <table border="1"> <thead> <tr> <th data-bbox="654 331 1040 363">Control</th> <th data-bbox="1040 331 1435 363">Function</th> </tr> </thead> <tbody> <tr> <td data-bbox="654 363 1040 972"> <p>Threshold Track Bar (A) and Textbox (B).</p> </td> <td data-bbox="1040 363 1435 972"> <p>Sets the threshold used to segment objects from the background.</p> <p>Moving the track bar (A) sets the current threshold level and generates a preview of the segmentation process using the current frame. The white pixels represent the fluorescent signal and black pixels as the background. The preview is visible on the process layer.</p> <p>Clicking set saves these values for further steps.</p> </td> </tr> <tr> <td data-bbox="654 972 1040 1108"> <p>Guess Button(C)</p> </td> <td data-bbox="1040 972 1435 1108"> <p>Sets the threshold to the average pixel intensity + the standard deviation of the pixel intensity.</p> </td> </tr> <tr> <td data-bbox="654 1108 1040 1879"> <p>Minimum Object Size Track Bar (D) and Textbox (E).</p> </td> <td data-bbox="1040 1108 1435 1879"> <p>Sets the minimum sized object for centroid tracking.</p> <p>Moving the track bar (D) shows the effect of the minimum size cut off on the objects used for tracking on the progress layer.</p> <p>Green objects are included in subsequent processing while red objects are excluded.</p> <p>The Orange numbers adjacent to each object indicates the area in pixels of the object.</p> <p>The threshold value must be set with the set button before this tool can be used.</p> </td> </tr> </tbody> </table>	Control	Function	<p>Threshold Track Bar (A) and Textbox (B).</p>	<p>Sets the threshold used to segment objects from the background.</p> <p>Moving the track bar (A) sets the current threshold level and generates a preview of the segmentation process using the current frame. The white pixels represent the fluorescent signal and black pixels as the background. The preview is visible on the process layer.</p> <p>Clicking set saves these values for further steps.</p>	<p>Guess Button(C)</p>	<p>Sets the threshold to the average pixel intensity + the standard deviation of the pixel intensity.</p>	<p>Minimum Object Size Track Bar (D) and Textbox (E).</p>	<p>Sets the minimum sized object for centroid tracking.</p> <p>Moving the track bar (D) shows the effect of the minimum size cut off on the objects used for tracking on the progress layer.</p> <p>Green objects are included in subsequent processing while red objects are excluded.</p> <p>The Orange numbers adjacent to each object indicates the area in pixels of the object.</p> <p>The threshold value must be set with the set button before this tool can be used.</p>
Control	Function								
<p>Threshold Track Bar (A) and Textbox (B).</p>	<p>Sets the threshold used to segment objects from the background.</p> <p>Moving the track bar (A) sets the current threshold level and generates a preview of the segmentation process using the current frame. The white pixels represent the fluorescent signal and black pixels as the background. The preview is visible on the process layer.</p> <p>Clicking set saves these values for further steps.</p>								
<p>Guess Button(C)</p>	<p>Sets the threshold to the average pixel intensity + the standard deviation of the pixel intensity.</p>								
<p>Minimum Object Size Track Bar (D) and Textbox (E).</p>	<p>Sets the minimum sized object for centroid tracking.</p> <p>Moving the track bar (D) shows the effect of the minimum size cut off on the objects used for tracking on the progress layer.</p> <p>Green objects are included in subsequent processing while red objects are excluded.</p> <p>The Orange numbers adjacent to each object indicates the area in pixels of the object.</p> <p>The threshold value must be set with the set button before this tool can be used.</p>								

	Prefilter (F)	<p>Toggles the use of additional filters to remove shot noise from the image.</p>
	FIONA Tools (G-H)	<p>Refines the center position of a tracked object chosen by the object number in the textbox (G) by fitting the intensity to a Gaussian distribution in two dimensions.</p> <p>Outputs a CSV file containing nanometer precise trajectory information created from the selected object in a format compatible with Trajectory Viewer.</p> <p>The refined centers are visible on the FionaDots Layer.</p> <p>These tools are best used in connection with the “select object” tool located in the Image Interaction Tool Bar. The “select object” tool automatically enters the object number in the textbox by clicking on the tracked object.</p>
	Set Button (I)	<p>Sets the threshold and minimum save tracking steps.</p>
	Track Button (J)	<p>Executes the functions to threshold, qualify, and link the fluorescent objects in the field using the settings established above.</p> <p>The tracking process can be viewed on the Progress layer.</p>
	Save All (K)	<p>Saves all of the positional data to a single trajectory file.</p>
ROIs	Contains controls for interacting with ROI within the program.	

Control	Function
 <p>The screenshot shows a software window titled "Qdot Controls ROIs Log Better Qdot Tracking Empty ActinBreaking Tab TabPage7 NADA Tracking". It contains a list of ROIs (A) and several buttons: "Get ROI's" (B), "Delete ROI" (C), "Return ROI Info" (J), "Return Obj Info" (K), "Update ROI" (G), "Copy ROI info" (H), "Go to First Frame" (I), and "Copy Frame Number" (H). There are also textboxes for "First Frame" (D), "Last Frame" (E), and "Description" (F).</p>	<p>List of ROIs (A) A selectable list ROI. In the format "ROI number" ", " "Description."</p>
	<p>Get ROIs (B) Button populates/refreshes the list of ROI. Updates textboxes D-F with the ROI's information.</p>
	<p>Delete ROI (C) Deletes the selected ROI.</p>
	<p>First Frame Textbox (D) Displays the first frame of the selected ROI's range.</p>
	<p>Last Frame Textbox (E) Displays the last frame of the selected ROI's range.</p>
	<p>Description Textbox (F) Displays the description from the selected ROI.</p>
	<p>Update ROI (G) Updates the ROI with the values from the textboxes D-F</p>
	<p>Copy ROI info (H) Copies the ROI information to the clipboard in a format compatible with spread sheets.</p>
	<p>Go to first frame (I) Sets the current frame to the first frame of the ROI by moving the "Frame Selection Bar" (Figure 2B)</p>
	<p>Copy Frame Number (H) Appends the current frame number to the ROIs description textbox.</p>
	<p>Return ROI Int (J) Copies the total and average intensity for the pixels within the ROIs as a function of time to the clipboard.</p>
	<p>Return Obj Int (K) Copies the intensity of all of the tracked objects as a function of time and copies the total and average intensity for each object and the total and average intensity for the whole field.</p>
<p>Log*</p>	<p>Contains a single textbox that reports debugging information.</p>
<p>Better Qdot Tracking</p>	<p>An implementation of the intensity based positional refinement tools with additional filters designed to be used on high frame rate images. Specifically the filters on this tab allow particles to be tracked initially at low temporal resolution. This low resolution information acts as a seed for positional refinement, allowing the background to be ignored and the intensity profile fitting refinement to be limited to only</p>



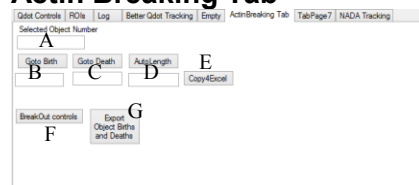
regions which are known to contain signal. This greatly improves the ability of the software to perform nanometer precision refinement in high temporal resolution.

Control	Function
Number of frames to Add number selector (A)	Sets the number of frames to sum in the tracking algorithm.
Number of frames to average number selector (B)	Sets the number of adjacent pixels to average in the tracking algorithm.
Fluorescent Object Tracking Tools (C- I)	This tool functions the same as tools in the “Qdot Controls” tab above with the exception that additional filters have been added to average the adjacent pixels and the number of frames can be summed to increase the effective exposure.
FIONA ALL (J)	Performs the same intensity profile refinement as the FIONA button on the Qdot tab but performs the refinement on all of the tracked objects.
Save All Qdot Data (K)	Saves All current data from the intensity based positional refinement of fluorescent objects as trajectories compatible with Trajectory Viewer.

Empty*

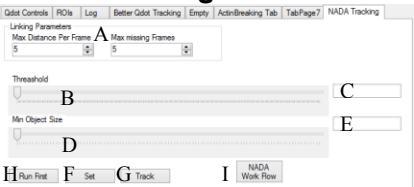
Contains controls used in development hidden from the user.

Actin Breaking Tab



Provides tools for the rapid semi-automatic analyses of actin breaking.

Control	Function
Selected Object Textbox (A)	Displays the object number of the currently selected object selected by the “Select Object” tool Image Interaction Tool Bar
Goto Birth (B)	Jumps to the first frame that the object was detected and copies the frame number to the text box below the control.
Goto Death (C)	Jumps to the last frame that the object was

	detected and copies the frame number to the textbox below the control.								
	Auto Length (D) Performs the same measurement as the “MeasureLengthOnClick” tool in the Image Interaction Tool Bar and copies the result to the textbox below.								
	Copy4Excel (E) Copies objects first frame, last frame and measured length from the textboxes to clipboard so these parameters can be noted in excel.								
	Breakout controls (E) Produces a window with the controls present on this tab. When this tool is used in connection with “EnableLargerImage” it greatly simplifies the interface for the user increasing usability.								
	Export Object Births and Deaths (F) Generates a file containing the Object reference number, the first frame, the last frame and the observed lifetime for each tracked filament.								
Tab Page 7*	Contains development tools hidden from the user.								
<p>NADA Tracking</p>  <p>The screenshot shows the NADA Tracking software interface. At the top, there are menu items: Qdot Controls, FIOs, Log, Better Qdot Tracking, Empty, ActinBreaking Tab, TabPage7, and NADA Tracking. Below the menu is a 'Linking Parameters' section with a dropdown menu for 'Max Distance Per Frame' (set to A) and a 'Max missing Frames' input field (set to 5). There are two sliders: 'Threshold' (set to B) and 'Min Object Size' (set to D). At the bottom, there are buttons: 'H Run First', 'F Set', 'G Track', and 'I NADA Work Flow'.</p>	<p>Tools for the export of SPT trajectories for processing in other software packages.</p> <table border="1"> <thead> <tr> <th data-bbox="651 1356 1032 1392">Control</th> <th data-bbox="1032 1356 1401 1392">Function</th> </tr> </thead> <tbody> <tr> <td data-bbox="651 1392 1032 1528">Controls for experimental linking method (A)*</td> <td data-bbox="1032 1392 1401 1528">Setting for an unfinished method of linking objects into trajectories. They are currently unused.</td> </tr> <tr> <td data-bbox="651 1528 1032 1633">Fluorescent Object Tracking Tools (C- G)</td> <td data-bbox="1032 1528 1401 1633">These tools function the same as tools in the “Qdot Controls” tab.</td> </tr> <tr> <td data-bbox="651 1633 1032 1896">“Run First” (H)</td> <td data-bbox="1032 1633 1401 1896">Automatically measures the intensity of each frame in the selected frame range to adjust for fluctuations in average intensity resulting from processes such as photo-bleaching and provides an</td> </tr> </tbody> </table>	Control	Function	Controls for experimental linking method (A)*	Setting for an unfinished method of linking objects into trajectories. They are currently unused.	Fluorescent Object Tracking Tools (C- G)	These tools function the same as tools in the “Qdot Controls” tab.	“Run First” (H)	Automatically measures the intensity of each frame in the selected frame range to adjust for fluctuations in average intensity resulting from processes such as photo-bleaching and provides an
Control	Function								
Controls for experimental linking method (A)*	Setting for an unfinished method of linking objects into trajectories. They are currently unused.								
Fluorescent Object Tracking Tools (C- G)	These tools function the same as tools in the “Qdot Controls” tab.								
“Run First” (H)	Automatically measures the intensity of each frame in the selected frame range to adjust for fluctuations in average intensity resulting from processes such as photo-bleaching and provides an								

	initial guess for the threshold value.	
	<p>NADA Work Flow (I)</p> <p>Single button work flow for exporting SPTs from this software.</p> <p>After setting the threshold and “min object” size the user clicks this button and follows the prompts to produce trajectories compatible with Trajectory Viewer.</p>	
*Experimental functions or functions which exceed the scope of this dissertation.		

Table 6 – Particle Tracker’s Main Task Tabs

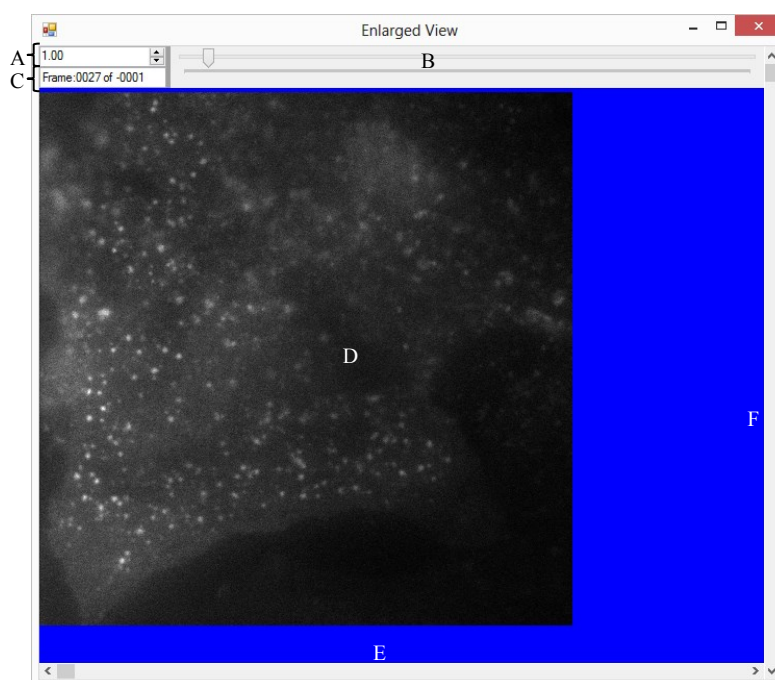


Figure 3 – Particle Tracker’s Enlarged View Window

The Enlarged View Window mirrors many of the controls and functions of the “Screen and Playback Controls” located on Particle Tracker’s main window. This increases flexibility and simplifying the user interface of the particle tracer main window allowing user to work across multiple monitors. Further this control includes tools to enlarge the view of the image being analyzed which is often a great assistance to manual annotation of videos.

Component	Function
Zoom Control (Figure 2A)	Sets the level of magnification for the image displayed in Screen the control.
Frame Selection Bar (Figure 2B)	Selects the current frame. Mirrors the Frame Selection Bar on the main window.

Current frame display/Last Frame Display (Figure 2C)	Displays the current frame and the total number of frames that have data associated with them.
Screen (Figure 2D)	Renders images based on the layers chosen in the Drawing Layer Tool Bar (Figure 1E).
Scrollbars (Figure 2 E and F)	Allows the user to view different parts of the enlarged image.

Table 7 – Description of Particle Tracker’s Enlarged View Window’s Controls

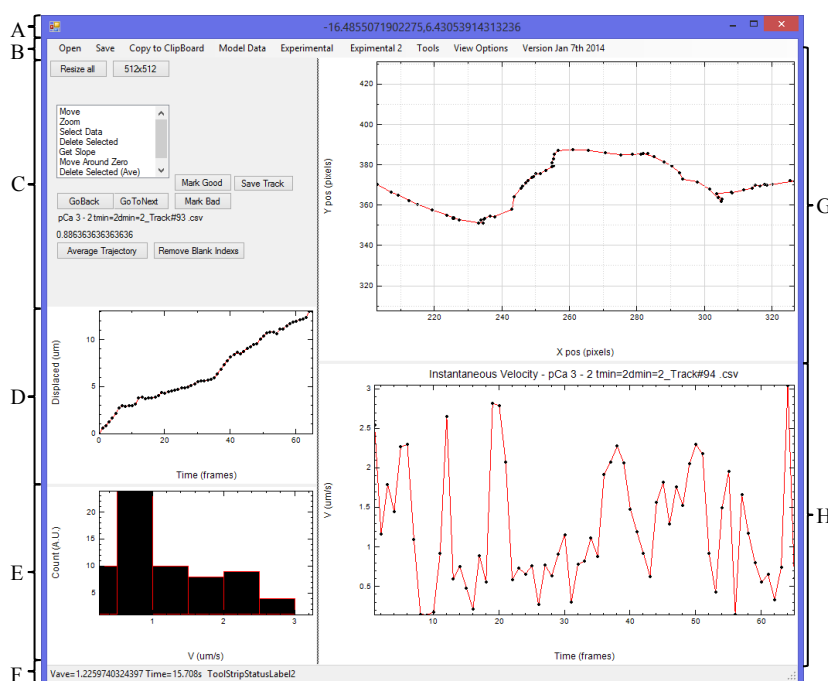


Figure 4 – Trajectory Viewer’s Main Window

This program provides the tools to rapidly qualify and manipulate trajectories and perform both simple and advance calculations on trajectories in a user-friendly manner.

Component	Function
Title Bar (Figure 4A)	Displays the position of the mouse as the user moves the mouse over the plots in the main window in the units of the plot.
Menu Bar (Figure 4B)	Contains the tools and functions described in Table 9.
Control Cluster (Figure 4C)	A collection of controls to manipulate the loaded trajectories, interact with the plots, index through the list of loaded trajectories The individual functions of these controls are described in Table 10.
Distance from First Point Graph (Figure 4D)	This tool displays the distance in μm between the first point in the graph and the point at the specified time index in frames. Right clicking the plot loads a menu that allows the user to copy the displayed data.
Instantaneous	Displays a histogram of instantaneous displacements calculated with a

Velocity Histogram (Figure 4E)	<p>window width of one frame.</p> <p>Right clicking the plot loads a menu that allows the user to copy the displayed data.</p>
Status Bar (Figure 4F)	<p>Displays information about the loaded trajectories such as the average velocity and duration and provides feedback regarding the progress and ETA for other processes.</p> <p>Clicking on the text within the status bar copies the text to the clipboard.</p>
Trajectory Graph (Figure 4G)	<p>Displays single or multiple trajectories in pixels as a scatter plot.</p> <p>Each of the individual points in the plot represents a point in the trajectory. A line is used to aid the eye linking the individual points together.</p> <p>Missing points in the trajectories are indicated as breaks in the red lines between points.</p> <p>Right clicking the plot loads a menu that allows the user to copy the displayed data.</p>
Instantaneous Velocity Graph (Figure 4H)	<p>Displays the instantaneous velocity of single or multiple trajectories calculated with a window width of one frame.</p> <p>If multiple trajectories are loaded the graph will contain one line for each trajectory.</p> <p>Right clicking the plot loads a menu that allows the user to copy the displayed data.</p>

Table 8 – Description of Trajectory Viewer’s Main Window’s Controls

Location	Menu Item	Function
Open		
	Open Single Trajectories	Opens a file dialog which allows the user to select one or multiple trajectories to load in the Trajectory Viewer. When multiple files are chosen the user is prompted to view all the files at once or one at a time.
	Open Directory of Trajectories	Behaves the same as “Open Single Trajectories” except that it opens all of the trajectories in the selected directory and any sub directories to overcome the OS limit imposed on the maximum number of files which can be selected in a dialog at once.
Save		
	Copy To Clipboard*	The menu contains several legacy features. The user should use the right click menus for each plot.
	Model Data*	Experimental functions for modeling motion of single particles.
	Experimental*	Various experimental and obsolete functions.
	Experimental	Various experimental and obsolete functions.

2*		
Tools		
Average duration calculator	Measures the duration for each file in the list of trajectories and calculates the average duration. This tool also allows the user the option to copy raw values to the clipboard.	
Append Trajectories	Appends multiple trajectories loaded at once end to end with no net rotation to build a continuous trajectory.	
Append Trajectories w/ Random Rotation	Appends multiple trajectories loaded at once end to end with random rotation to build a continuous trajectory.	
Build NADA Displacement File	Generates a NADA histogram file to be viewed in NADA Histogram Fitter.	
MSD Calculator Step 1	<p>First step in calculating the mean squared displacement for a collection of trajectories.</p> <p>This menu option loads each file in the directory and calculates a table of displacements between points at every possible window width. This is the same calculation performed to generate the NADA displacement file only applied to single trajectories.</p>	
MSD Calculator Step 2	<p>Second step in calculating the mean squared displacement for a collection of trajectories.</p> <p>This option uses files generated in step one to calculate the mean squared displacement at each window width for the files in the selected directory. The function generates a multi column .csv file containing the results of the measurement.</p>	
View Options		
Draw Dots	Suppresses rendering of dots in the plots which vastly speeds up the drawing of large datasets but causes some undesired effects.	
Size to 512x512 after redraw	Every time the trajectory is redrawn the screen is restored to 512x512 pixels which is equivalent to the viewable field of the Ropper 512b Camera used in our lab.	
Change Velocity Histogram Options	Changes the parameters used to produce the Instantaneous Velocity Histogram (Figure 4E) plot.	
Date last Modified	Shows the last time the program was last modified.	
*Experimental functions or functions which exceed the scope of this dissertation.		

Table 9 – Summary of Menu Functions in Trajectory Viewer

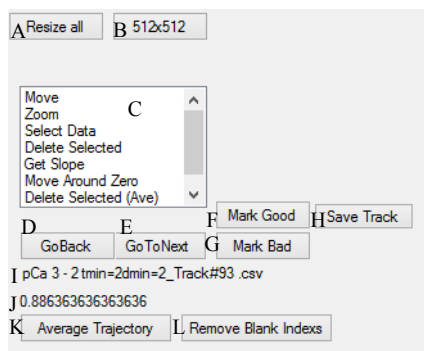


Figure 5 – Control Cluster from Trajectory Viewer

An annotated view of the control cluster in Trajectory Viewer. The functions of the control clusters controls are listed in Table 10.

Component	Function																		
Resize all (Figure 5A)	Redraws all plots to their full range.																		
512x512 (Figure 5B)	Sets the visible area to the equivalent of 512x512 pixels.																		
Plot Tools (Figure 5C)	A set of tools which function on all of the plots within the program and allow the user to manipulate the data. <table border="1"> <thead> <tr> <th>Item</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>Move</td> <td>Centers the view to match the location of the mouse click.</td> </tr> <tr> <td>Zoom</td> <td>User draws a box using two mouse clicks and the zoom is adjusted to match the extents of the box.</td> </tr> <tr> <td>Select Data</td> <td>User draws a box using two mouse clicks to select a portion of the trajectory and the selected data is copied to the clipboard in a spread sheet compatible format.</td> </tr> <tr> <td>Delete Selected</td> <td>User draws a box using two mouse clicks to select a portion of the trajectory and the selected data is deleted from the plot.</td> </tr> <tr> <td>Get Slope*</td> <td>A function for getting the average slope of the selected points. Disabled because it doesn't work when multiple trajectories are loaded.</td> </tr> <tr> <td>Move Around Zero*</td> <td>Obsolete function for moving selected points to be centered around 0,0. Use the "Return Centered Around Zero" tools under the save menu.</td> </tr> <tr> <td>Delete Selected (Ave)*</td> <td>Functions the same as the "Delete Selected" except this function replaces the deleted point with average of the adjacent points. Disabled because of an unhandled error on deleting the first or last point in trajectory.</td> </tr> <tr> <td>Count</td> <td>Allows the user to count points in a region of the plot by drawing a box around the points.</td> </tr> </tbody> </table>	Item	Function	Move	Centers the view to match the location of the mouse click.	Zoom	User draws a box using two mouse clicks and the zoom is adjusted to match the extents of the box.	Select Data	User draws a box using two mouse clicks to select a portion of the trajectory and the selected data is copied to the clipboard in a spread sheet compatible format.	Delete Selected	User draws a box using two mouse clicks to select a portion of the trajectory and the selected data is deleted from the plot.	Get Slope*	A function for getting the average slope of the selected points. Disabled because it doesn't work when multiple trajectories are loaded.	Move Around Zero*	Obsolete function for moving selected points to be centered around 0,0. Use the "Return Centered Around Zero" tools under the save menu.	Delete Selected (Ave)*	Functions the same as the "Delete Selected" except this function replaces the deleted point with average of the adjacent points. Disabled because of an unhandled error on deleting the first or last point in trajectory.	Count	Allows the user to count points in a region of the plot by drawing a box around the points.
Item	Function																		
Move	Centers the view to match the location of the mouse click.																		
Zoom	User draws a box using two mouse clicks and the zoom is adjusted to match the extents of the box.																		
Select Data	User draws a box using two mouse clicks to select a portion of the trajectory and the selected data is copied to the clipboard in a spread sheet compatible format.																		
Delete Selected	User draws a box using two mouse clicks to select a portion of the trajectory and the selected data is deleted from the plot.																		
Get Slope*	A function for getting the average slope of the selected points. Disabled because it doesn't work when multiple trajectories are loaded.																		
Move Around Zero*	Obsolete function for moving selected points to be centered around 0,0. Use the "Return Centered Around Zero" tools under the save menu.																		
Delete Selected (Ave)*	Functions the same as the "Delete Selected" except this function replaces the deleted point with average of the adjacent points. Disabled because of an unhandled error on deleting the first or last point in trajectory.																		
Count	Allows the user to count points in a region of the plot by drawing a box around the points.																		
GoNext (Figure 5D)	Used when reviewing trajectories. Loads the next trajectory in the list.																		
GoBack (Figure 5E)	Used when reviewing trajectories. Loads the previous directory in the list																		

Mark Good (Figure 5F)	Used when reviewing trajectories. When the user clicks on this button the trajectory file is moved into a subdirectory named “Good”.
Mark Bad (Figure 5G)	Used when reviewing trajectories. When the user clicks on this button the trajectory file is moved into a subdirectory named “Bad”.
Save Track (Figure 5H)	Saves any modifications made to the displayed trajectory.
Last Loaded File (Figure 5I)	Displays the filename of the last loaded trajectory.
% progress (Figure 5J)	Used when reviewing trajectories to display the percentage of the file list that remains to be reviewed.
Average Trajectory (Figure 5K)	Prompts the user for the number of adjacent points to average and then performs adjacent averaging for each point in the trajectory. Should not be performed when multiple trajectories are viewed at once.
Remove Blank Indexes (Figure 5L)	Removes empty indexes from the loaded trajectory.
*Experimental functions or functions which exceed the scope of this dissertation.	

Table 10 – Description of the Control Clusters Controls

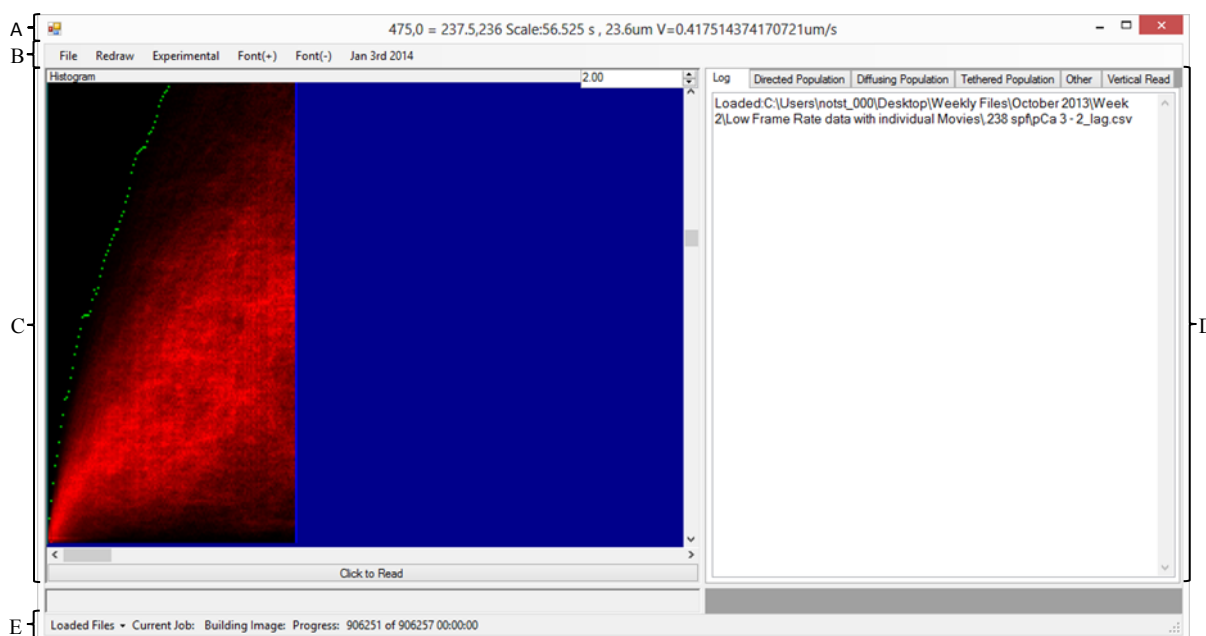


Figure 6 – NADA Histogram Fitter's Main Window

The NADA Histogram Fitter's main window provides access to the utilities and functions to visualize and quantify the NADA Histogram data generated in the trajectory viewer software. These functions are summarized in Table 11.

Component	Function
The Title Bar (Figure 6A)	<p>When loading large data files the title bar indicates the progress by updating the text in the title bar caption every 10^4 loaded lines.</p> <p>Reports the position of the mouse in pixels and scaled units as the user moves the mouse over the NADA histogram image.</p>
Menu Bar (Figure 6B)	<p>The menu provides access to the major functions within the software.</p> <p>These functions are listed in detail in Table 12.</p>
NADA Histogram Control (Figure 6C)	<p>The NADA histogram control is a composite of several different controls which together provide a method of visualizing the NADA histogram and interacting with the visualization. This control is further discussed in Figure 7 and Tables 13 and 14.</p>
Task Tabs (Figure 6D)	<p>The task tabs organize the functions of the software into tabs based on the task being performed. These tabs are summarized in Table 15.</p>
Status Bar (Figure 6E)	<p>Provides feedback to the user. This component is composed of two controls, a pull down list (left side of the control) that indicates the loaded data files, and a textbox that informs the user of the current task being performed and provides an estimate of the time remaining till completion in the format DD.HH:MM:SS.</p>

Table 11 – Description of NADA Histogram Fitters Controls

Location	Menu Item	Function
File Menu		Contains commands related to loading files and saving program settings.
	Open Lag File	Loads NADA histogram files (Files ending with “_lag.csv”)
	Batch Load Lag Files	Automates loading of several NADA histogram files by loading each file in a new instance of NADA fitter.
	Save Setting	Saves the current settings for the program such as the bin sizes for the NADA histogram. These settings are used by default when opening the software and using the “Batch Load Lag Files” option.
Redraw Menu		Contains Commands related to the Drawing of the NADA histogram control.

Create Histogram Image	Repeats the drawing of the NADA histogram. Allows for rescaling the NADA histogram or for rebuilding the image.
Change Cell Size	Allows the user to change the binning of the NADA histogram without recalculating the bins in the histogram. This is useful when working with large datasets.
Set LagMax and Dmax	Limits the range of the histogram of data used to build NADA histograms by setting a maximum window width and displacement for the NADA histogram. This accelerates work with large data sets.
Velocity Transform*	Converts the histogram into a plot of Velocity as a function of window width, Ideally converting velocity populations to horizontal intensities.
Square Displacements*	Squares the displacements within the NADA histogram data linearizing diffusing populations.
Do not normalize	Draws the histogram without any normalization or intensity maximization. The resulting image will appear dark and external contrast enhancement is required to visualize the resulting histogram.
Enable Reading Lines	Enables/Disables the reading lines on the NADA histogram.
Enable Edge of Data Dots	Enables/Disables the reading lines on the NADA histogram.
Experimental	Contains many unfinished/incompletely tested routines.
Font(+)	Increases the Font Size in the program
Font(-)	Decrease the Font Size in the program
Last Modification Date	Displays the last date the software was modified.
* Experimental function or functions beyond the scope of this dissertation.	

Table 12 – Summary of Menu Function in NADA Histogram Fitter

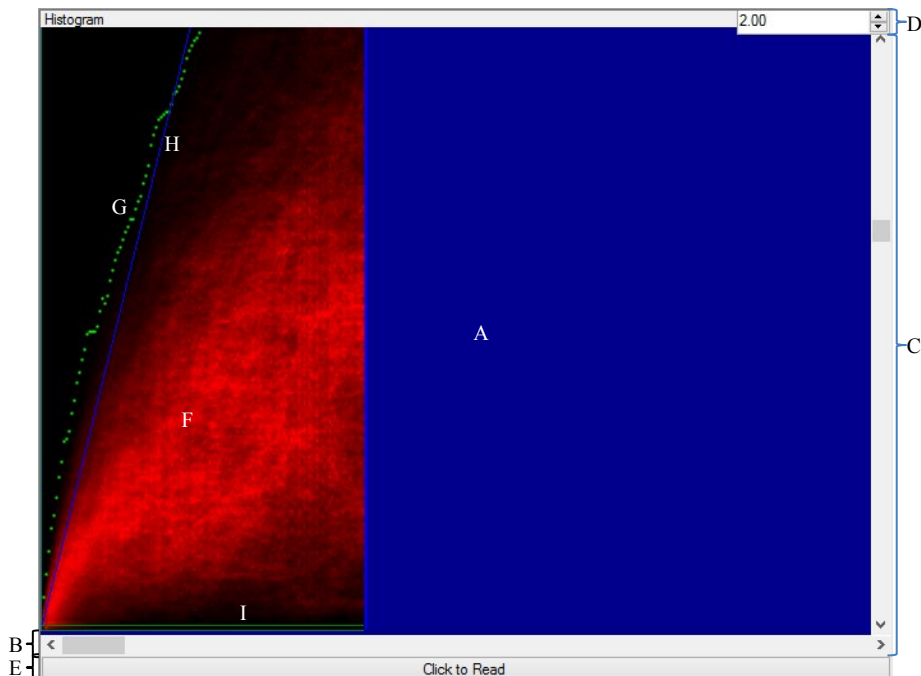


Figure 7 – The NADA Histogram Control

An annotated view of the NADA histogram control with the major components labeled. The function of these components is summarized in Table 13.

Component	Function
Drawing Surface (A)	Displays the NADA histogram image and provides visual feed back to the user about the locations of the NADA histogram that are used in other functions. The right click menu's functions for this control are summarized in Table 14.
Scroll Bars(B and C)	Move the NADA histogram image so the whole image can be viewed.
Zoom Control (D)	Enlarges the view of the NADA image within the drawing surface.
“Click to Read Button” (E)	A tool that allows the user to click on the NADA histogram directly to record the mouse's current position within the NADA histogram. The first click will copy the click's position within the NADA histogram to the textbox within the log Tab. The second click will calculate the slope and y-offset of a line joining the first and second clicks and report the resulting slope and y-offset to the log tab.
NADA Histogram Image (F)	A graphical representation of the Data in the NADA histogram.

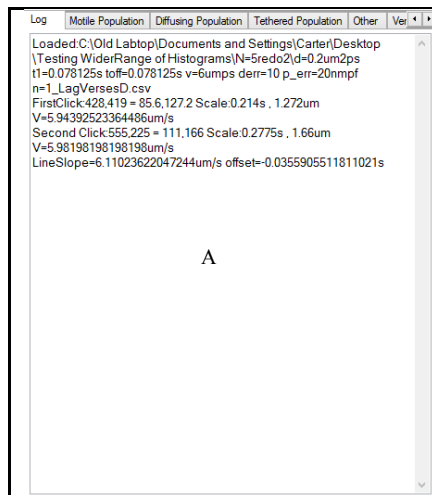
Edge of Data Indicator (G)	Indicates the upper limit of the displacements at each window width with a green pixel.
Directed Population's Reading Lines(H)	Indicates the upper and lower limits of the Directed population's intensity measurement.
Tethered Population's Reading Lines (I)	Indicates the upper and lower limits of the Tethered population's intensity measurement.

Table 13 – Summary of the NADA Histograms Control's Components.

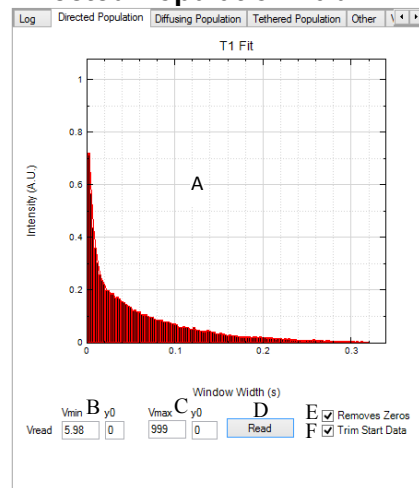
Right click Menu Item	Function
Copy Whole Image	Copies the whole NADA Histogram to the system clipboard to be used in other software packages.
Copy Image File	Generates a .PNG file in the system clipboard containing the NADA Histogram image.
Copy selection from image	Copies a selection of the NADA Histogram image specified from time and displacement of zero to the specified window width and displacement. This is useful for generating figures for publication.
Paste Image	Replaces the current image in the NADA histogram control with an image in the system clipboard. This function allows the user to use and external software package such as ImageJ to filter the image to enhance the signature of individual populations within the NADA histogram and then transfer the filtered image back into the software.
Revert Image	Replaces the image currently in the NADA histogram control with the image stored internally during the histograms generation or during the last paste operation.

Table 14 – Description of NADA Histogram Control's Right Click Menu Options

Tab Name and Image	Controls and function				
Log Tab	Used to provide feedback to the user such as measurements from clicking the NADA image. <table border="1" data-bbox="646 1549 1430 1661"> <thead> <tr> <th>Control</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>Log textbox (A)</td> <td>Displays the log for the software.</td> </tr> </tbody> </table>	Control	Function	Log textbox (A)	Displays the log for the software.
Control	Function				
Log textbox (A)	Displays the log for the software.				



Directed Population Tab

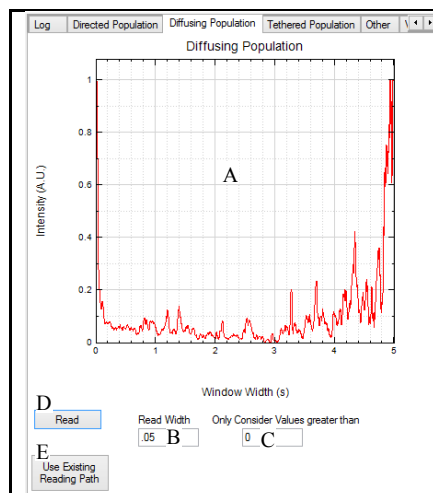


Measures the intensity of populations running horizontally in the NADA histogram to determine the lifetime of the directed population. After the first read has been performed 2 blue lines (Figure 7H) will indicate the region within the NADA histogram corresponding to the read location.

Control	Function
Plot (A)	Plots the result from the directed population read. The Right click menu options for this plot are described in Table 16.
Lower Range controls (B)	Sets the slope (V) and y offset (y0) for the line describing the lower range of the directed population.
Upper Range controls (C)	Sets the slope (V) and y offset (y0) for the line describing the Upper range of the directed population.
Read (D)	Reads the intensity between the upper and lower limits and returns the result to the plot.
Removes Zeros	Removes trailing Zeros from the plot.
Trims Start Data	Removes noisy data located at small window widths.

Diffusing Population Tab

Reads the center of the distribution to obtain the diffusing populations lifetime above the tethered population. This function can also be used to read any other complex reading paths through the histogram.



Control

Function

Plot (A)

Plots the result from the directed population read.

The Right click menu options for this plot are described in Table 16.

Read Width (B)

Sets the distance above and below the calculated path to include in the measurement.

Only Consider Values greater than (C)

Sets the lower limit of displacements to include in path calculation. Allows for the user to exclude tethered regions of the NADA histogram.

Read (D)

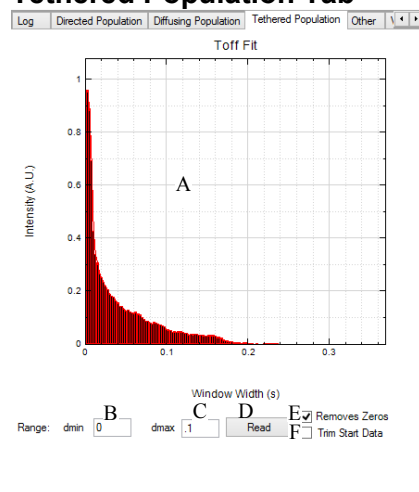
Calculates the average displacement at each window width and uses that path as the center of the intensity read of the NADA histogram. The result is plotted in the plot control.

Use Existing Reading Path (E)

Performs the same read as the "Read" Button but takes a user provided path.

Accepts two column CVS files with window width in units of seconds as the first column and the displacement in μm as the second column.

Tethered Population Tab



Measures the intensity of populations running horizontally in the NADA histogram to determine the lifetime of the tethered population. After the first read has been performed 2 green lines (Figure 4I) will indicate the region within the NADA histogram which corresponds to the read location.

Control

Function

Plot (A)

Plots the result from the tethered population's read.

The Right click menu options for this plot are described in Table 16.

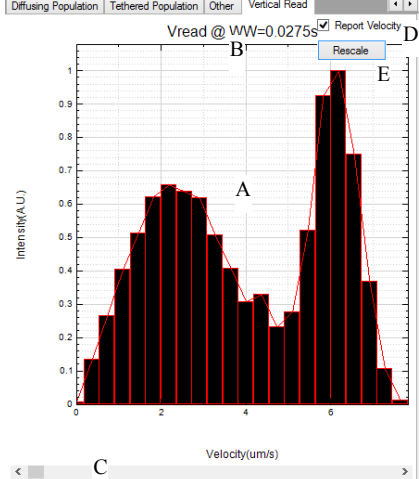
	<table border="1"> <tr> <td data-bbox="654 199 1023 331">Lower Range controls (B)</td> <td data-bbox="1023 199 1435 331">Sets the offset (y0) for the line describing the lower range of the tethered population.</td> </tr> <tr> <td data-bbox="654 331 1023 464">Upper Range controls (C)</td> <td data-bbox="1023 331 1435 464">Sets the offset (y0) for the line describing the upper range of the tethered population.</td> </tr> <tr> <td data-bbox="654 464 1023 596">Read (D)</td> <td data-bbox="1023 464 1435 596">Reads the intensity between the upper and lower limits and returns the result to the plot.</td> </tr> <tr> <td data-bbox="654 596 1023 674">Removes Zeros</td> <td data-bbox="1023 596 1435 674">Removes trailing Zeros from plot.</td> </tr> <tr> <td data-bbox="654 674 1023 779">Trims Start Data</td> <td data-bbox="1023 674 1435 779">Removes noisy data located at small window widths.</td> </tr> </table>	Lower Range controls (B)	Sets the offset (y0) for the line describing the lower range of the tethered population.	Upper Range controls (C)	Sets the offset (y0) for the line describing the upper range of the tethered population.	Read (D)	Reads the intensity between the upper and lower limits and returns the result to the plot.	Removes Zeros	Removes trailing Zeros from plot.	Trims Start Data	Removes noisy data located at small window widths.		
Lower Range controls (B)	Sets the offset (y0) for the line describing the lower range of the tethered population.												
Upper Range controls (C)	Sets the offset (y0) for the line describing the upper range of the tethered population.												
Read (D)	Reads the intensity between the upper and lower limits and returns the result to the plot.												
Removes Zeros	Removes trailing Zeros from plot.												
Trims Start Data	Removes noisy data located at small window widths.												
<p data-bbox="186 787 435 814">Vertical Read Tab</p> 	<p data-bbox="646 787 1435 919">Allows the user to sample the distribution of displacements or velocities at any given window width. After the first use a cyan line indicates the location of the read on the NADA histogram.</p> <table border="1"> <thead> <tr> <th data-bbox="654 919 1023 961">Control</th> <th data-bbox="1023 919 1435 961">Function</th> </tr> </thead> <tbody> <tr> <td data-bbox="654 961 1023 1297">Plot (A)</td> <td data-bbox="1023 961 1435 1297"> <p data-bbox="1023 961 1435 1171">Plots the distribution of displacements at the window with indicated by the label on the top of the plot. (a vertical read of the NADA plot)</p> <p data-bbox="1023 1192 1435 1297">The Right click menu options for this plot are described in Table 16.</p> </td> </tr> <tr> <td data-bbox="654 1297 1023 1360">Read Location Label (B)</td> <td data-bbox="1023 1297 1435 1360">Indicates the location of the read.</td> </tr> <tr> <td data-bbox="654 1360 1023 1430">Scroll Bar (C)</td> <td data-bbox="1023 1360 1435 1430">Sets the location of the current read position.</td> </tr> <tr> <td data-bbox="654 1430 1023 1570">Report Velocity (D)</td> <td data-bbox="1023 1430 1435 1570">Changes the scale of the plot and reports the effective V of the displacements.</td> </tr> <tr> <td data-bbox="654 1570 1023 1671">Rescale (E)</td> <td data-bbox="1023 1570 1435 1671">Rescales the scroll bar to include the whole range of the NADA histogram.</td> </tr> </tbody> </table>	Control	Function	Plot (A)	<p data-bbox="1023 961 1435 1171">Plots the distribution of displacements at the window with indicated by the label on the top of the plot. (a vertical read of the NADA plot)</p> <p data-bbox="1023 1192 1435 1297">The Right click menu options for this plot are described in Table 16.</p>	Read Location Label (B)	Indicates the location of the read.	Scroll Bar (C)	Sets the location of the current read position.	Report Velocity (D)	Changes the scale of the plot and reports the effective V of the displacements.	Rescale (E)	Rescales the scroll bar to include the whole range of the NADA histogram.
Control	Function												
Plot (A)	<p data-bbox="1023 961 1435 1171">Plots the distribution of displacements at the window with indicated by the label on the top of the plot. (a vertical read of the NADA plot)</p> <p data-bbox="1023 1192 1435 1297">The Right click menu options for this plot are described in Table 16.</p>												
Read Location Label (B)	Indicates the location of the read.												
Scroll Bar (C)	Sets the location of the current read position.												
Report Velocity (D)	Changes the scale of the plot and reports the effective V of the displacements.												
Rescale (E)	Rescales the scroll bar to include the whole range of the NADA histogram.												

Table 15 – Summary of the Task Tabs in NADA Histogram Fitter

Right click Menu Options for the Plots in NADA Histogram Fitter	
Copy Data to Clipboard	Copies the data used to build the plot to the clipboard in a format compatible with spread sheet software.
Copy Image 2 Clipboard	Copies the Image of the plot to the clipboard.
Send to Origin	Uses OriginLab's API to import the data directly.

Table 16 – Description of NADA Histogram Fitter's Plots Control's Right Click Menu Options

References

1. Thompson, R. E., Larson, D. R. & Webb, W. W. Precise nanometer localization analysis for individual fluorescent probes. *Biophys. J.* **82**, 2775–83 (2002).
2. Jaqaman, K. *et al.* Robust single-particle tracking in live-cell time-lapse sequences. **5**, (2008).
3. Skiljan, I. IrfanView. at <<http://www.irfanview.com/>>
4. Schneider, C. a, Rasband, W. S. & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. *Nat. Methods* **9**, 671–675 (2012).
5. Ruhnnow, F., Zwicker, D. & Diez, S. Tracking single particles and elongated filaments with nanometer precision. *Biophys. J.* **100**, 2820–8 (2011).
6. Cheezum, M. Quantitative Comparison of Algorithms for Tracking Single Fluorescent Particles. *Biophys. J.* **81**, 2378–2388 (2001).
7. Yildiz, A. & Selvin, P. R. Fluorescence imaging with one nanometer accuracy: application to molecular motors. *Acc. Chem. Res.* **38**, 574–82 (2005).
8. Qian, H. Single particle tracking. Analysis of diffusion and flow in two-dimensional systems. *Biophys. J.* **60**, 910–921 (1991).
9. Saxton, M. Single-particle tracking: models of directed transport. *Biophys. J.* **67**, 2110–2119 (1994).
10. Martin, D., Forstner, M. & Kas, J. Apparent Subdiffusion Inherent to Single Particle Tracking. *Biophys. J.* **83**, 2109–2117 (2002).
11. Gorga, J. a, Fishbaugher, D. E. & VanBuren, P. Activation of the calcium-regulated thin filament by myosin strong binding. *Biophys. J.* **85**, 2484–91 (2003).
12. Klinth, J., Arner, A. & Månsson, A. Cardiotonic bipyridine amrinone slows myosin-induced actin filament sliding at saturating [MgATP]. *J. Muscle Res. Cell Motil.* **24**, 15–32–32 (2003).