Improving The Scope and Quality of Single-Molecule Data Analysis

by

David J. Rowland

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Biophysics)
in the University of Michigan
2017

Doctoral Committee:

Assistant Professor Julie S. Biteen
Professor Jeffrey A. Fessler
Professor Nils G. Walter
Professor Jens-Christian Meiners

DEDICATION

This dissertation is dedicated to my family and friends near and far. You did your best to answer my questions—science or otherwise—and that's all anyone could ever ask for.

ACKNOWLEDGMENT

I would like to acknowledge the help I have recieved from my friends and colleagues in the Biophysics program at the University of Michigan. Matthew Stone and Yi Liao helped a lot with the math involved in correlation spectroscopy and diffusion, respectively. Hannah Tuson helped take and think about the data. Sam Kotler and Carter Swanson joined me on the journey. My adviser, Julie, put up with my antics and showed me how science works.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF APPENDICES

Abstract

Single-molecule fluorescence permits super-resolution imaging of the molecular-scale inside living systems. In this thesis I investigate the analysis techniques that provide useful insights into such single-molecule data sets with the goal of improving both the scope and the quality of the analysis. Traditional algorithms for localizing isolated fluorescent emitters assume stationary point light sources. Chapter 2 proposes two fitting functions that achieve similar nanometer-scale localization precision as the traditional symmetric Gaussian function, while allowing, and explicitly accounting for, directed motion. The precision of these methods is investigated through Fisher information analysis, simulation and experiments, and the new fitting functions are then used to measure, for the first time, the instantaneous velocity and direction of motion of live bacteria cells. These new methods increase the information content of single-molecule images of fast-moving molecules without sacrificing localization precision, thus permitting slower imaging speeds, and my new fitting function promise to improve single-particle tracking (SPT) algorithms by calculating velocity and direction during each image acquisition. Chapter 2 demonstrates a method for increasing the scope of single-molecule data analysis by introducing an additional measurable variable—the distance a point light source traverses while it undergoes straight, directed motion during the image capture.

By following single fluorescent molecules in a microscope, SPT can measure diffusion and binding on the nanometer and millisecond scale. Still, although SPT can at its limits characterize the fastest biomolecules as they interact with subcellular environments, this measurement may require advanced illumination techniques such as stroboscopic illumination. In Chapter 3, I address the challenge of measuring fast subcellular motion by instead analyzing single-molecule data with spatio-temporal image correlation spectroscopy (STICS) with a focus on measurements of confined motion. My SPT and STICS analysis of simulations of the fast diffusion of confined molecules shows that image blur affects both STICS and SPT, and I find biased diffusion rate measurements for STICS analysis in the limits of fast diffusion and tight confinement due to fitting STICS correlation functions to a Gaussian approximation. However, I determine that with STICS, it is possible to correctly interpret the motion that blurs single-molecule images without advanced illumination techniques or fast cameras. In particular, I present a method to overcome the bias due to image blur by properly estimating the width of the correlation function by directly calculating the correlation function variance instead of using the typical Gaussian fitting procedure. My simulation results are validated by applying the STICS method to experimental measurements of fast, confined motion: I measure the diffusion of cytosolic mMaple3 in living *Escherichia coli* cells at 25 frames per second under continuous illumination to illustrate the utility of STICS in an experimental parameter regime for which in-frame motion prevents single-particle tracking and tight confinement of fast diffusion precludes stroboscopic illumination. Overall, my application of STICS to freely diffusing cytosolic protein in small cells extends the utility of single-molecule experiments to the regime of fast confined diffusion without requiring advanced microscopy techniques.

Diffusion in these biological systems is rarely simple—over the course of each individual

particle track, that particle can undergo multiple modes of motion, including for instance motion that is fast and slow, confined and unconfined, or anomalous and normal. To increase the signal-to-noise ratio from single-trajectory analysis, this heterogeneous motion can be effectively quantified by fitting the cumulative probability distribution (CPD) of the total collection of squared step sizes. In the most commonly used CPD data analysis algorithms, however, a two-step fitting process allows degenerate fitting parameters to vary independently. Such over-complicated fitting models are prone to overfitting, or fitting to noise. Here I combine this two-step process into a single least-squares minimization step; this new method vastly reduces the total number of fitting parameters and thus the degrees of freedom to increase the precision with which diffusion coefficients and diffusive population weights may be measured, even when significant heterogeneities exist. I demonstrate this global fit approach on a simulated two-component system with varying amounts of localization precision to show an improvement in fitting robustness as well as a 2- to 3-fold improvement of precision compared to the traditional local fit algorithm. Finally, I measure the diffusion of a mixture of 80 nm and 200 nm gold spheres to yield 3-fold precision improvements in this real system. Chapter 4 of this thesis seeks to improve the quality of an existing single-molecule super-resolution data analysis method by combining the multiple fitting steps required for CPD analysis into a single step. Together with the improvements in scope in Chapter 2, and quality in Chapter 3, this third endeavor of my thesis helps to maximize the information content of single-molecule super-resolution data.

CHAPTER I

Introduction

1.1 Fluorescence imaging of bacteria

Bacteria are very small; at 1 micrometer in length, 100 of them stacked end to end are about as wide as a human hair. They are so small that until Jones et al. in 2001 described bacterial actin homologs that establish an internal cytoskeleton, the commonly held view was that bacteria exhibited no long-range internal structure. [1] Since then, investigations into the intracellular structure of bacterial molecular systems has become an active area of research. [2, 3] Once samples are fixed, frozen, or desiccated, electron microscopy [4, 5] can promise the extraordinary resolution capable of discerning the subcellular structures due to the short wavelength of the electron illumination, but those methods are quite destructive because of sample sectioning, the vacuum required, and chemical fixation and, as a result, cannot monitor the time-resolved changes in structure that often play integral roles in interesting bacterial phenotypes. Life occurs in a time-resolved manner that is abolished by these harsh observation conditions and so there exist many questions that would benefit from imaging technologies that both have molecular resolution and non-destructive sample preparations. For example, common super-resolution fluorescence imaging techniques can be shown to

allow normal cell division even after the conclusion of the experiment.

Live cell fluorescence super resolution imaging is capable of capitalizing on the advances in the past 20 years [6, 7] in the fields of photoswitchable, photoactivatable, and blinking organic dyes and genetically encodable proteins that can be used to observe intracellular position information with high temporal (∼10 ms) and high spatial (∼20 nm) resolution while simultaneously allowing the cells to continue their typical homeostatic operations.

To surpass the diffraction limit, single-molecule super resolution microscopy captures an image of a point source as a distribution of photons with a known shape, i.e. an Airy Disk. The image is subsequently fit to a Gaussian (which is a close approximation to the functionally much more complex Airy Disk) to locate the center of the point source to a resolution of around 20 nm as shown in Figure 1.1.1.

Photoswitching, photoactivation, or experimentally controllable blinking allowes conventional light microscopy and wide-field fluorescence (imaging of bulk labeling of structures with many overlapping fluorophores) to capture images of spatially or temporally separated individual fluorescently labeled molecules. The centers of these individual molecular images can be estimated to ∼20 nm resolution—thereby defeating the Abbe diffraction limit that classically limited the closest distance two structures could be separated and still be differentiable by light microscopy and bulk fluorescence imaging. [10] In other words, the minimum distance between to fluorescent spots, d, is $d = \frac{\lambda}{2NA}$, where λ is the wavelength of the light emitted by the fluorophore, and NA is the numerical aperture of the microscopes objective. Our setup, shown in Figure 1.1.2 and designed for the use of the commonly used fluorophore PA-mCherry which emits at 680 nm, uses an objective with a NA of 1.4, so $d \approx 240$ nm.

Single-molecule super-resolution microscopy has developed into a rich ecosystem of re-

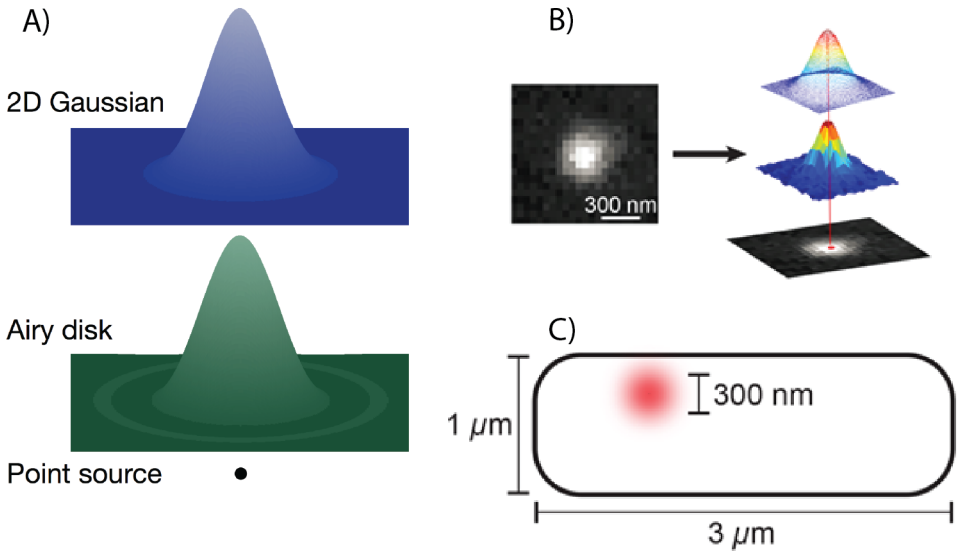

Figure 1.1.1: A) The Airy disk is the expected distribution of photons emitted from a point source emitter as imaged with a typical microscope, but is well approximated by a symmetric 2D Gaussian function. B) The image of a fluorophore (left) is fit to a Gaussian function (right) to locate the center position of the fluorophore. C) Shown in red is the diffraction limited spot of a fluorophore inside a typical bacterium. The interaction distances associated with physiological processes require localization precision that is below this 300 nm spot size, and this is the chief impetus for single molecule super resolution localization microscopy employed in B). Figure adapted from Haas et al. [8] and Tuson et al. [9]

lated techniques that each intend to beat the diffraction limit of light, such as photoactivatable localization microscopy (PALM), [11] and fluorescence photoactivation localization microscopy (FPALM), [12] which use photoactivatable or photoswitchable fluorescent proteins, and stochastic optical reconstruction microscopy [13] (STORM) and direct-STORM13 [14] (dSTORM) which use synthetic fluorophores. All of these related methods seek to separate individual fluorescent molecules in either space or time such that in any given fluorescence image each fluorescent molecule is separated by at least several widths of the PSF of the microscope. By separating the fluorescent molecules (whether they are quantum dots, organic dyes, or genetically encodable fluorescent proteins) in time (photactivatable, photoswitch-

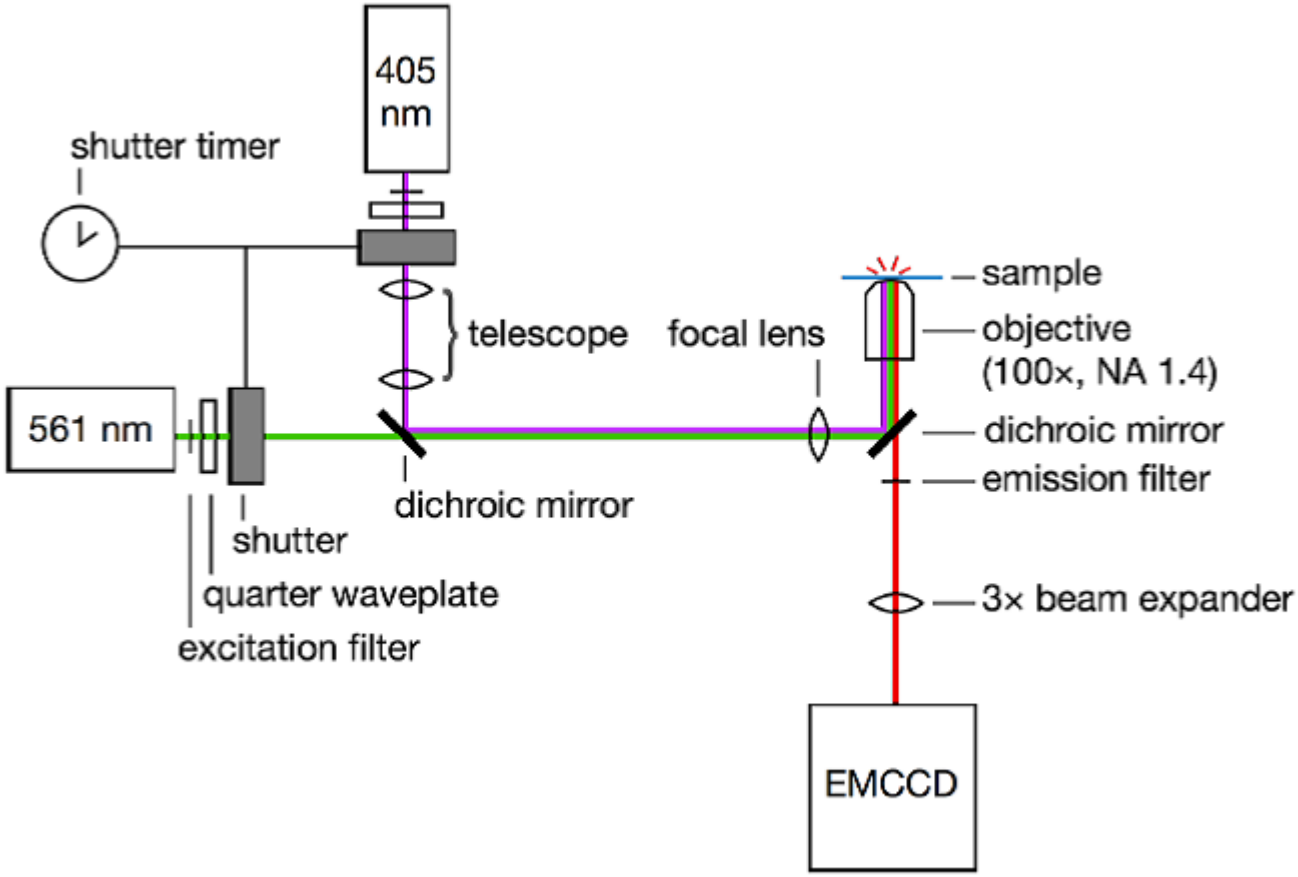

Figure 1.1.2: Schematic of the system used for super-resolution fluorescence imaging using photoactivatable or photoswitchable fluorophores. The sample is first bleached with the 561 nm excitation laser, then a small subset of fluorophores in the sample are activated by a short pulse of 405 nm light and subsequently imaged until bleaching with 561 nm light. Figure taken from Haas et al. [8]

able, or controlled blinking) or space (low concentration), the image of the fluorescence intensity of single fluorophores may be seen as a close approximation to a single symmetric Gaussian spot on a dark background. Knowing this, the images of individual molecules may be localized to ∼20 nm resolution [15] every ∼10 ms by fitting to a Gaussian function with parameters such as the location, width, and amplitude of the fluorescent spot. Other techniques that can estimate similar properties include center of mass [16] and maximum likelihood estimation [17]. Center of mass methods capitalize on the speed of computation of a discrete weighted average of the pixel intensities of a diffraction limited spot, but can suffer from bias due to region of interest ROI selection. Maximum likelihood methods, however, achieve the Cramer-Rao lower bound on the estimation precision, [18] but suffer from slow computational speeds and more stringent theoretical requirements to be met.

Current camera technology that can successfully capture the dim fluorescence of individual molecules is limited by the signal to noise ratio (SNR) of the camera as well as the limited photon budgets of the fluorescent molecules. Temporal resolutions below 10 ms require the number of photons and the fidelity of their capture to reach the lower limit of SNR required by the super-resolution image analysis algorithms. But because scientific cameras are approaching 100% quantum efficiency, faster integration times might simply require brighter or more stable fluorescent molecules.

The single-molecule super-resolution microscopy paradigm allows researchers to probe the previously ignored subcellular structure and live-cell dynamics of bacteria in a less destructive way than electron microscopic methods. The molecule or structure in question is labeled genetically with a fluorescent protein and can be shown to maintain functionality. The minimally perturbed cell can then grow relatively unperturbed albeit with a fluorescent marker tracing the live-cell dynamics of the labeled structure. There is a simple and easily

used paradigm for super resolution imaging that uses wide-field excitation with low power diode lasers, circular polarizers, dichroic filters, a typical inverted microscope, and an electron multiplying charge coupled device (EMCCD) camera (see Figure 1.1.2 and reviews). [6, 9]

Alternative to this simple design, there are several imaging modalities that offer improvements in the ratio of signal to noise in the images. Confocal imaging [19] (a pinhole and scanning mirrors work in concert to capture fluorescence from a grid of locations on the sample on a single-pixel photodetector), light sheet microscopy [20] (fluorescence excitation of only a thin plane of the sample thereby eliminating background fluorescence), stroboscopic illumination (a method to further decrease the fluorophore excitation time beneath the minimum camera exposure time), and 4pi imaging [21] (fluorescence imaging that uses two objectives to capture twice as many of the photons emitted from the excited fluorophore) improve upon this basic paradigm but my work concerns improvements to the commonly used methods and does not seek to reach the absolute best setup for a given experiment. Instead, I ask what is the best analysis that can be done given the type of data that we can already acquire? Also, typical academic laboratories are rarely able to purchase the newest and best system for each new experiment and so versatility is important. And this versatility is made possible by theoretical extensions such as those discussed. It is an added bonus that the improvements discussed in this dissertation can also be readily applied to many of the techniques mentioned above.

1.2 Point spread function (PSF) modifications

1.2.1 Engineered PSFs

The pattern of fluorescence emission from emitters (fluorescent proteins or dyes) as captured by a camera contains information about the location, local environment, and motion of the emitters. The image of the fluorescent molecule can be modified by optical components such as in astigmatic optics, [22, 23] or spatial light modulators [24] to increase the amount of useful information that can be gathered by altering the classical symmetric PSF of a fluorescent molecule to be for instance an asymmetric Gaussian or double helix point spread function DH-PSF. Figures 1.2.1 and 1.2.2 are schematics of these two methods, which both enable localization along the imaging axis. Also notable is the improved sensitivity near the focal plane of the DH-PSF due to its shape and subsequent increased Fisher Information. In other words, the shape of the DH-PSF contains more information about the position of the molecule than the original unperturbed PSF.

1.2.2 Experimentally altered PSFs

Additionally, the image of a fluorescent molecule can be modified by the local environment or by qualities of the molecule itself. Isotropic diffusion on short time scales acts to broaden the PSF monotonically with increasing diffusion coefficient, allowing for single-image readouts of the diffusion coefficient, D, of a diffusing particle [25, 26] according to the following equation

$$D = \frac{s_{x,y}^2 - s_0^2}{(2A_{x,y} + 0.0643)t} \tag{1.2.1}$$

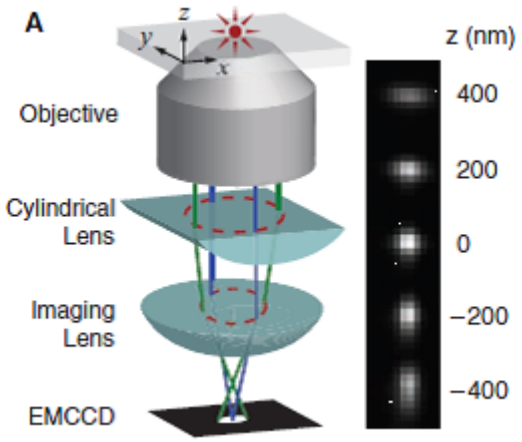

Figure 1.2.1: For astigmatic three-dimensional imaging, a cylindrical lens is placed in the optical path between the objective and the camera that causes the focal point of the x and y (in-plane) dimensions to be at different sample depths. The ratio of PSF widths determines the z (axial) location of the molecule. Figure taken from Huang et al. [22]

where $s_{x,y}$ is the standard deviation of the Gaussian fit to the single image of a diffusing fluorophore, s_0 is the standard deviation of a Gaussian fit to an immobile fluorophore, and t is the exposure time of the camera. Both $A_{x,y}$ and the 0.0643 number are empirically determined coefficients found by simulation or separate calibration experiments. Directional motion also modifies the image of a fluorescent molecule. Systems consisting of largely one-dimensional motion such as motion along DNA curtains [27] or diffusion along one-dimensional microchannels in nanocrystals [28] restricts the in-frame displacements to a straight line, causing the images to be skewed in a single direction. If the nature of these deviations from the PSF of an immobile point light source is understood, the fitting function used for localization can elucidate much more than just the molecular location and trajectories through time. In Chapter 2, I expand the toolbox of modified PSF fitting functions to include one that explicitly accounts for directed in-frame motion that allows the estimation of instantaneous particle velocity. [29]

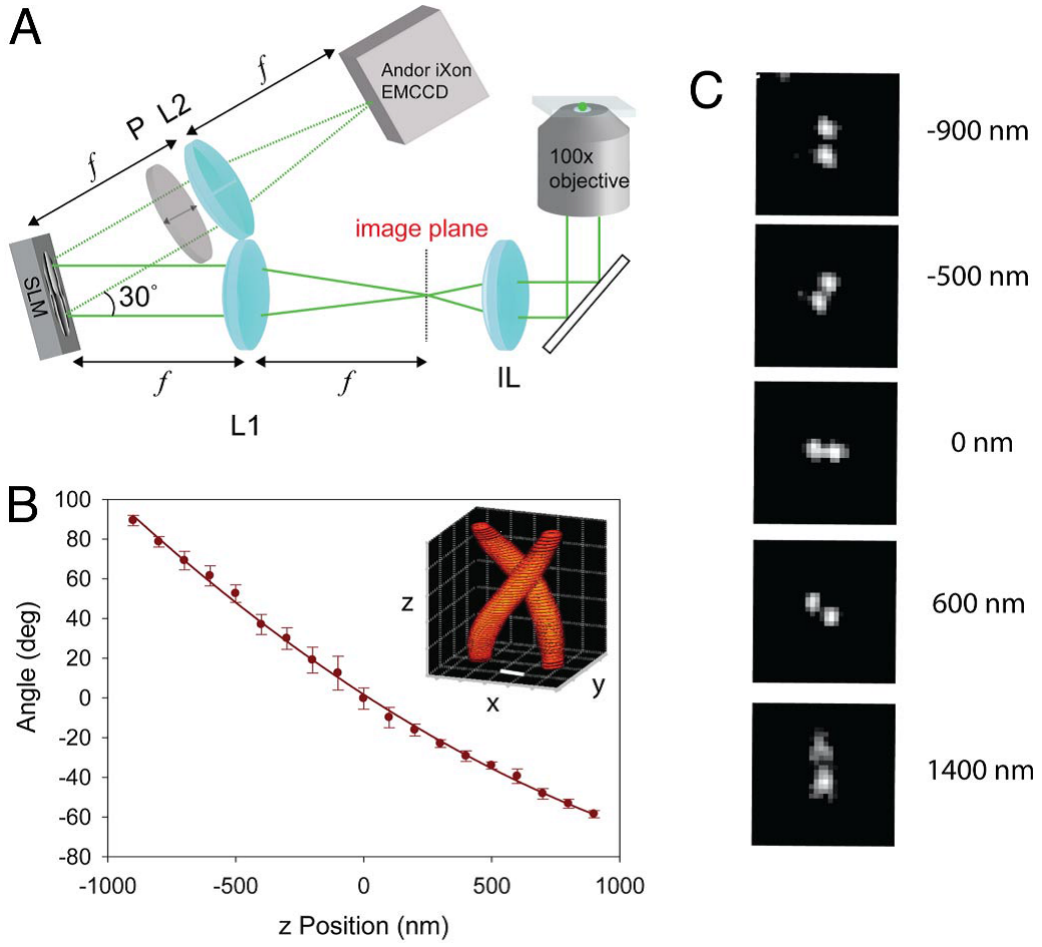

Figure 1.2.2: To create a double helix point spread function, a spatial light modulator (SLM) is placed in the Fourier plane of the fluorescence signal between the objective and the camera. The SLM causes the axial position of a fluorophore with respect to the focal plane of the objective to manifest as an angle between the lobes and the horizontal. Figure taken from Pavani et al. [24]

1.3 Image correlation spectroscopy

Alternatively, the motion of fluorescent molecules can be analyzed by the two-dimensional correlation of image pairs from a time series of images of the diffusing molecules. Image correlation, simplified and illustrated in Figure 1.3.1 as the correlation of two one-dimensional functions, compares the differences between two functions. The cross-correlation of the top-hat and triangle shapes results in a function with a center of mass displaced to the right, which can be understood as a readout of the fact that the peak of the triangle is to the right of its center of mass. Similarly, if the two functions are Gaussian PSFs of a diffusing molecule, their correlation will be a third Gaussian displaced from the origin by the distance that the molecule moved from one image to the other. Isotropic diffusion will result in cross-correlations that are displaced with equal probability around the origin. To utilize spatiotemporal image correlation spectroscopy (STICS), the cross-correlations corresponding to identical time-lags are averaged, resulting in a time-lag series of correlation functions that broaden as time-lag increases. This broadening is caused by the fact that longer time-lags allow more time for the diffusing particle to move farther from its previous location. [30]

Apart from STICS, fluorescence fluctuation spectroscopy methods such as two-color image cross-correlation spectroscopy [31] (which allows separate imaging systems to be accurately cross-correlated), k-space ICS [32] (which accounts for the complex photophysics of real fluorophores) and raster ICS [33] (which accurately formulates the cross-correlation procedure for a scanning microscope) further generalize the methods to a wide range of biophysical systems. Chief among the benefits of these spectroscopic methods is the relaxing of the requirement for particle immobility that Gaussian localization relies upon. Without the requirement that the imaged fluorophores be stationary enough to be approximately

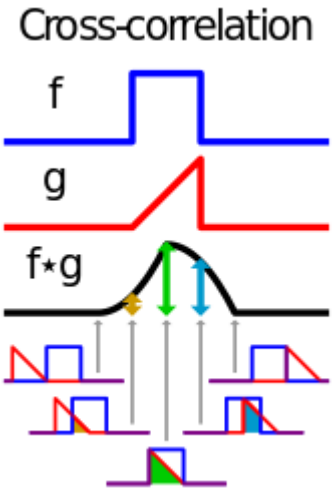

Figure 1.3.1: The cross-correlation of two functions, f and g. Adapted from Wikipedia Commons.

symmetric, the maximum frame rate of the microscopes camera plays less of a role in determining maximum measurable velocities. With these various ICS techniques, blurring due to slow camera frame-rate or fast diffusion does not distort the quality or quantity of data recovery. By including all pixels of the image of a quickly diffusion molecule, ICS methods are less prone to the false negatives that are common in localization-based analyses because of their inherent selection bias of the brightest, most Gaussian-shaped images. In Chapter 3, I apply spatiotemporal ICS to single molecule fluorescence inside bacteria for the first time and explore its uses and drawbacks that arise from fast diffusion and tight confinement—two difficulties that make it particularly hard to image molecules in bacteria cells. [34]

1.4 Diffusion analysis via cumulative probability distributions and mean squared step sizes

Localization of the centers of the PSFs of diffusing molecules, however, effectively enhances the signal to noise ratio for experimental readouts of the system such as diffusion coefficients and radius of gyration, or the width of the region inside of which a molecule diffuses. This localization step potentially improves on image correlation techniques because the localization algorithm ignores large sections of the total region of interest in favor of the pixels immediately surrounding the signal. By confining the possible locations of a molecule to a small area, only pixels with high information content are included in the analysis. Also, there are many opportunities to reduce the probability of falsely identifying noise as fluorescent molecules by using thresholds to prune the bulk output of the localization algorithm to only include particle localizations that conform to physically relevant parameter regimes. If localization of diffusing molecules is possible, meaning that within a single camera exposure, the molecule diffuses less than approximately the width of its PSF (commonly less than 10 m^2s^{-1}), the set of locations may be associated into trajectories with single particle tracking (SPT). [35] From the trajectories of mobile particles the mean squared displacements (MSD) as a function of time-lag can be used to estimate the diffusive properties of the particle. [36–39] These studies are limited to homogeneously diffusing systems, as well as—albeit to a lesser degree—heterogeneous systems. The limitation to homogeneous systems is largely due to the fact that to address the diffusion of multiple species from a set of trajectories, an empirical threshold must be used to separate the individual MSDs (one for each trajectory) into subsets representing the different populations.

Another, more promising method of addressing data that contain trajectories of molecules

that diffuse at different speeds is to make use of the trajectories cumulative probability distributions (CPD) of squared step sizes. [40,41] The CPDs (one for each time-lag) are first used to estimate the MSD vs. time-lag for each population that are then used to estimate the diffusive properties of each population, such as diffusion coefficients, diffusive population amplitudes, or velocity of drift. For example, given a collection of particle trajectories, the mean squared displacement for a given time-lag is estimated from the CPD of squared step sizes for that time lag. One mean squared displacement is estimated from each CPD, and the MSD vs time lag curve is then fit to a diffusion model (for unconfined 2D isotropic diffusion: $MSD(\tau) = 4D\tau + 4\sigma^2$, where D is the diffusion, τ is the time-lag, and σ is the localization precision). In Chapter 4, I simplify this algorithm that estimates the diffusion coefficients and population amplitudes of heterogeneous diffusion from SPT data and report its improved precision and robustness through simulation and experiment.

1.5 Thesis outline

In this Thesis, I aimed to improve the quantity and quality of the analysis of single-molecule super-resolution data by rethinking the fitting functions used for localization of mobile fluorescent molecules. I applied and analyzed the measurement biases of image correlation spectroscopy to fast confined diffusion inside bacteria. By combining a multi-step analysis algorithm into a single-step procedure, I designed and analyzed a more robust and more precise SPT analysis method. These investigations into different analyses techniques all focus on the analysis of the same single molecule super resolution data sets and can potentially be used in tandem to greatly strengthen the types of conclusions that may be made from a single experiment.

In Chapter 2, I considered a method for measuring instantaneous directed motion of single molecules by altering the fitting function used to estimate the shape of the intensity profile of moving fluorophores./citeRowland2014 Simulation, experiment, and Fisher information analysis were used to investigate two methods for measuring in-frame directed motion. We found that these methods directly increased the information content of single-molecule images of fast-moving molecules without sacrificing localization precision.

In Chapter 3, I investigated STICS as a method for measuring the blurry motion of a bacterial cytosolic protein that moves too quickly to be localized with Gaussian fitting./citerowland2016 With simulation we found that, while STICS subverts the biases inherent in SPT of fast molecules, STICS imposes other biases due to confinement and fast diffusion. In addition, we present a method to overcome these previously undescribed biases with a look-up table and present for the first time the measurement of fast confined diffusion inside bacteria.

In Chapter 4, I reexamined the commonly used CPD/MSD method for measuring parameters related to the diffusion of molecules and find that using a single two-domain fitting procedure instead of many separate one-domain fits reduces the number of free parameters to such a degree that the measurement precision of the diffusion coefficient is greatly improved. Simulation of a two-population system as well as the diffusion of large and small gold nanospheres in glycerol solution were used to validate the method as a direct improvement to the classical CPD algorithm in both precision and robustness to noise.

Chapter 5 contains the conclusions regarding the scope of this work and future directions. I describe how from a single type of data, single-molecule super-resolution image sequences, there are a variety of analysis methods. In-frame directed motion can be directly observed and future work may include the analytical of in-frame diffusion. I showed in this work that

STICS may be performed on live bacteria albeit with the caveat of previously undescribed biases and then I present possible modifications to the method to avoid these biases. Finally, I discuss the direct improvements that can be gained by combining multi-step fitting processes into single-step processes in order to reduce the number of degrees of freedom and how such a concept may also be extended to STICS in the future.

CHAPTER II

Fitting functions for directed motion

The work presented in this chapter has been published in the following paper:

Rowland, D. J. & Biteen, J. S. Top-Hat and Asymmetric Gaussian-Based Fitting Functions for Quantifying Directional Single-Molecule Motion. *ChemPhysChem* 15, 712-720 (2014).

2.1 Introduction

Single-molecule fluorescence permits super-resolution imaging, but traditional algorithms for localizing these isolated fluorescent emitters assume stationary point light sources. Proposed here are two fitting functions that achieve similar nanometer-scale localization precision as the traditional symmetric Gaussian function, while allowing, and explicitly accounting for, directed motion. The precision of these methods is investigated through Fisher information analysis, simulation and experiments, and the new fitting functions are then used to measure, for the first time, the instantaneous velocity and direction of motion of live bacteria cells. These new methods increase the information content of single-molecule images of fast-moving molecules without sacrificing localization precision, thus permitting slower imaging speeds,

and our new fitting functions promise to improve tracking algorithms by calculating velocity and direction during each image acquisition.

Fluorescence microscopy is traditionally limited to several hundred nanometers by the diffraction of light. However, centroid fitting of isolated molecule fluorescent emission enables the locations of single molecules to be determined with very high precision. [42, 43] The ability to accurately locate individual molecules afforded by single-molecule fluorescence (SMF) microscopy can address fundamental questions in biology and chemistry that cannot be approached with bulk measurements or kinetics. Since SMF microscopy can be applied even to mobile molecules, it has been used to directly observe motions as diverse as molecular motors, proteins in live bacterial cells, DNA dynamics, and nanocargo translocation. [44–47] Here, SMF experiments record images of isolated fluorescent emitters, and the position of each emitter is considered to be the center of that emitters image (point spread function; PSF). This position determination can be achieved by calculating the center-of-mass, [48] by least-squares fitting to a Gaussian function, [39] or by maximum likelihood estimation (MLE). [49] If the emitter is small enough to be considered a point source, if the fluorophore emits isotropically, [50] and if the objects motion is negligible during the imaging frame, then these PSF-fitting methods provide reasonable estimates for the emitter position.

For point light sources that are stationary and in-focus during fluorescent imaging, the resulting PSF is an Airy disk that can be reasonably approximated by a symmetric Gaussian function with standard deviation, σ, which depends on the numerical aperture of the microscope objective and the wavelength of light. Thompson et al. reported closed-form expressions for the localization precision of immobile fluorescent molecules fit with this symmetric Gaussian fitting function. [15] Later, the theoretical limits of PSF fitting were determined from a more rigorous and general treatment of these fitting methods and their

statistics based on Fisher information theory, [51] and using the true image of a diffracted immobile point light source corrupted by noise. [18] Furthermore, this MLE-based method was implemented as a fast, iterative algorithm. [52] The simple symmetric Gaussian fitting function has been adapted for many applications, including three-dimensional imaging, [53] and Wang and co-workers added precision estimates for localization along the vertical axis by making use of the relation of standard deviation of the molecule image to the molecules distance from the objective focal plane. [54]

Furthermore, in many situations, even mobile molecules can be described by fits to the symmetric Gaussian function, as a sufficiently fast imaging speed can compensate for molecular motion. However, fast-moving molecules can no longer be considered essentially immobile since fast motion will blur the PSF extensively. Since the accuracy of the position determination depends on the quality of the fitting function, a fitting function that accounts explicitly for molecular motion is desirable. For the case of diffusive motion, no net direction of motion is expected: the molecule PSF will be blurred in all directions, and the convolution of the normal distribution of diffusive motion with the Gaussian-approximated photon distribution function of an immobile molecule results simply in another symmetric Gaussian with a larger standard deviation. This broadened σ has been related to the diffusion coefficient by empirical calibration [25] and by assuming the blurring to be represented by a path distribution function characterized by a normal distribution with a standard deviation that can be related directly to the molecular diffusion coefficient. [26] Error propagation and the results from these earlier works allow the error in the diffusion coefficient estimate to be predicted.

Though diffusion is ubiquitous, many biological processes are characterized by directed motion. For instance, bacterial chemotaxis and particle flow in a current. Rather than producing isotropic PSF widening, such directional motion causes the image PSF to be

blurred in the direction of motion, x. In this case, a fitting function that is derived considering molecular mobility will have the added benefit of providing information about the speed and directionality of the motion. To quantify directional motion, we have developed two methods that explicitly account for motion at a fixed velocity. The first method more exactly models a moving molecule PSF as the convolution of a stationary molecule PSF (approximated by a symmetric Gaussian function) and motion at a fixed speed and direction (represented by a top-hat function). The second method applies the asymmetric Gaussian, which has been previously used for three-dimensional imaging through an astigmatic lens [22], to a moving molecule; here the long axis of the asymmetric Gaussian corresponds to the direction of motion. Though Yüce et al. recently treated moving single-molecule fluorophores with a maximum likelihood approach, [55] the two fitting functions presented here have the benefit of being easily incorporated into established data processing routines.

In this chapter, we use Fisher information theory to analytically evaluate the precision (Section 2.3.1), and we fit experimental measurements of scattering from gold nanoparticles moving at a constant velocity to experimentally measure the precision (Section 2.3.3) for each of the fitting functions, q_{SErf} and q_{AG}, introduced above. We then apply the mobile single-molecule asymmetric Gaussian and SErf functions to fluorescently labeled *Vibrio cholerae* cells swimming in the imaging plane of a microscope in Section 2.3.4.

2.2 Theory and Experimental

2.2.1 Sum of Error functions (SErf) Fitting Function

The distribution of photons emitted from a fluorophore moving at a constant in-plane velocity during an imaging frame is approximated by the analytical 2D convolution (Eq. 2.2.1) of a top-hat function, h, with a symmetric Gaussian function, g:

$$h(x,y) = \delta(y) \times \begin{cases} 0: & x < \frac{D}{2} \\ 1: & -\frac{D}{2} < x < \frac{D}{2} \\ 0: & x > \frac{D}{2} \end{cases}$$

$$g(x,y) = \exp\left(-\frac{(x-x_0)^2 + (y-y_0)^2}{2\left(\frac{\sigma'}{a}\right)^2}\right)$$

$$q_{SErf} = h \otimes g \tag{2.2.1}$$

Here, $\delta(y)$ is a Dirac delta function, D is the displacement (along the x axis), σ' is the point spread function (PSF) standard deviation in nm, and the magnification factor, a, which depends on the dimensions of the image sensor and magnification of the microscope, is in the range of 50-200 nm pixel^{-1} for single-molecule fluorescence experiments. The center position of the molecule during the imaging frame is (x_0, y_0). We consider 2D motion restricted to the xy plane, where the axes are free to rotate through an angle, ϕ, such that the Cartesian coordinates x' and y' become

$$x = x'\cos(\phi)\hat{\mathbf{i}} - y'\sin(\phi)\hat{\mathbf{j}}; y = x'\sin(\phi)\hat{\mathbf{i}} + y'\cos(\phi)\hat{\mathbf{j}} \tag{2.2.2}$$

where $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ are the Cartesian unit vectors. All quantities x, y, x_0, y_0, D, and $(\sigma'/a$ are in units of pixels. Throughout the remainder of this discussion, the variable $(\sigma'/a$ is simplified to σ, the standard deviation in units of pixels, i.e. the number of pixels in one standard deviation of the PSF of the microscope. A higher σ means the PSF is spread across more pixels.

The distributive property of the derivative of convolutions is used to convolve a top-hat function, h, with a symmetric Gaussian function, g, as in Equation 2.2.1.

$$\frac{\partial}{\partial_x} q_{SErf}(x,y) = \left[\delta(y)\left(\delta\left(x+\frac{D}{2}\right)-\delta\left(x-\frac{D}{2}\right)\right)\right] \otimes \exp\left(-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}\right) \tag{2.2.3}$$

Equation 2.2.3 is convolved, integrated over all space and normalized to produce the SErf function:

$$q_{SErf}(x,y) = \frac{N}{D\sigma\sqrt{8\pi}} \exp\left(-\frac{(y-y_0)^2}{2\sigma^2}\right)\left(\mathrm{erf}\frac{x-x_0+D/2}{\sigma\sqrt{2}} - \mathrm{erf}\frac{x-x_0-D/2}{\sigma\sqrt{2}}\right) + b. \tag{2.2.4}$$

Equation 2.2.4 has been normalized to include the photon count, N, and the function is offset by a background level, b. Figure 2.2.1(a) shows g and h for typical single-molecule imaging conditions. Figure 2.2.1(b) and (c) show the results of this convolution for $D = 2$ pixels, and $D = 8$ pixels, respectively, with $N = 10^5$ photons, $b = 200$ photons pixels^{-1}, and $\sigma = 2$ pixels. In this chapter, noise is modeled as a Poisson distribution with mean and variance equal to b^2. The traces below and to the left of Figure 2.2.1(b) and (c) show the integrals of the respective PSF images over dy and dx, respectively. The PSF widening is nearly imperceptible in Figure 2.2.1(b), and becomes more obvious in Figure 2.2.1(c).

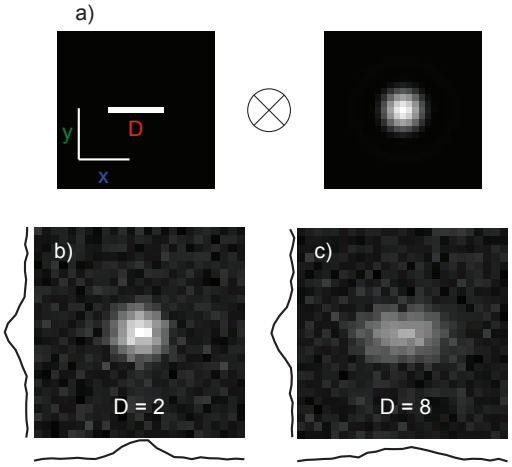

Figure 2.2.1: a) Graphical representation of the mathematical convolution of a top-hat function of width D (in-frame displacement length) in the x direction (direction of motion) and a symmetric Gaussian function that gives rise to the SErf function. b,c) Simulated SErf functions for $D = 2$ pixels (b) and $D = 8$ pixels (c). The integral of the images over dy and dx are shown below and to the left, respectively, of each image. Here, $N = 10^5$ photons, $b = 200$ photons pixel^{-1} , and $s = 2$ pixels.

2.2.2 Asymmetric Gaussian Fitting Function

As an alternative to the SErf function introduced in Section 2.1.1, we consider the asymmetric Gaussian function, q_{AG}:

$$q_{AG}(x,y) = \frac{N}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{(y-y_0)^2}{2\sigma_y^2} - \frac{(x-x_0)^2}{2\sigma_x^2}\right) + b \tag{2.2.5}$$

Here, σ_x and σ_y are the the standard deviations (in units of pixels) along the direction of motion and perpendicular to it, respectively, and x and y are again free to rotate in the xy plane about an angle ϕ according to Equation 2.2.2. The asymmetric Gaussian function, q_{AG} becomes the conventional symmetric Gaussian function, q_{SG} when $\sigma_x = \sigma_y = \sigma$.

The adaptation to the asymmetric Gaussian function that allows for the measurement of instantaneous directed motion requires the introduction of a calibration curve that relates the ratio of the standard deviations parallel and perpendicular to the direction of motion, that is, σ_x/σ_y, to D. The curve is populated by simulated data (see Section 2.2.4) with varying in-frame displacement, D, and the ratio σ_x/σ_y is obtained from an asymmetric Gaussian fit. The simulated data had $N = 10^5$ photons, $b = 200$ photons pixel^{-1}, $\sigma_y = 2$ pixels, and 1000 simulated data sets were fit for each D. This calibration curve is fit by a parabola,

$$\frac{\sigma_x}{\sigma_y} = cD^2 + 1 \tag{2.2.6}$$

The parabolic fit serves both as a look-up table for estimating displacement, D, from fitting parameters $\sigma_x(D)$ and $\sigma_y(D)$, as well as the method of estimating, by error propagation, the precision of D.

2.2.3 Fisher Information

Exact solutions for the asymmetric Gaussian function Fisher information integrals, Equation 2.3.1, were found by splitting these integrals into high- and low-noise regimes ($b > q$ and $b < q$, respectively). For the less analytically tractable SErf function, further simplification was required and Taylor expansions at $D = 0$ were used. Because numerical integration showed Δy to be independent of D at high photon count, the integral for y was further simplified by taking

$$\lim_{D\to 0}\left\{\frac{1}{b^2+q}\left(\frac{\partial q}{\partial y}\right)^2\right\}.$$

Because the variance of each parameter is reciprocal to its Fisher information, the square roots of the results in Table 2.1 give the RMS error. The Fisher information for the two functions was numerically integrated in Mathematica for comparison to the closed-form solutions. Error propagation,

$$(\Delta D)^2 = \left(\frac{\partial D\left(\sigma_x, \sigma_y\right)}{\partial \sigma_x}\right)^2 (\Delta \sigma_x)^2 + \left(\frac{\partial D\left(\sigma_x, \sigma_y\right)}{\partial \sigma_y}\right)^2 (\Delta \sigma_y)^2 \tag{2.2.7}$$

and the calibration curve in Equation 2.2.6, was used to find the precision of D as estimated by the asymmetric Gaussian function. This treatment was not necessary when evaluating ΔD for the SErf function because D is explicitly included in the function. Values of parameters for the results in this section were as follows: $D = 5.1$, $\sigma = 2$, $b = 200$ for the SErf integrals, and: $\sigma_x = 2.54$, $\sigma_y = 2$, $b = 200$ for the asymmetric Gaussian integrals. This value of σ_x was calculated from Equation 2.2.6 using the calibration constant from simulation, $c = 0.0104$, $D = 5.1$, and $\sigma_y = 2$.

Pixelation, which adds noise due to the finite size of a, is ignored in our solutions to the Fisher information (i.e. Figure 2.3.1 a), and b) and Table 2.1) because our results for RMS error were found to depend only weakly on pixel size within parameter regimes relevant to single-molecule experiments. For the SErf function and the asymmetric Gaussian function, pixelation can be accounted for by adding the variance of a top-hat distribution with width a, $(\Delta a)^2 = a^2/12$, to the variance of the point spread function, that is, σ^2 or σ_x^2 and σ_y^2 .

2.2.4 Simulated Data

Simulated images of point-source emitters were generated by weighting random numbers (generated by the Matlab routine *rand*) by a distribution function given by the numerical

convolution of a top-hat function and an Airy disk. The convolution was done at high resolution, 1000x1000 pixels, and the distributions were then down-sampled by linear interpolation to 25x25 pixels. The number of photons for each simulated image was chosen from a Poisson distribution with mean equal to the desired photon count. Poissonian noise with standard deviation of 200 counts per pixel was added to the simulated image. The computed precision in Figure 2.3.1 c), and d) was determined by fitting simulated data with the asymmetric Gaussian and SErf functions with the built-in bounded non-linear least squares minimization Matlab routine *lsqcurvefit*. The variance of the parameter estimations was calculated based on fitting 1000 simulated data sets for each value of N and D.

2.2.5 Controlled Nanoparticle Motion Imaging

Glass microscope coverslips were cleaned using an oxygen plasma etch (10 min at 200 mTorr; Plasma Etch, Inc. PE-50). Gold nanoparticle substrates were prepared on the cleaned coverslips using a spin-assisted layer-by-layer technique [56] based on polyelectrolyte films of positively charged poly(diallyldimethyl ammonium chloride) solution (PDADMAC, Sigma Aldrich) and negatively charged poly(sodium 4-styrene) solution (PSS, Sigma Aldrich). 26 nm diameter spherical gold nanoparticles (Nanopartz, Inc.) were used as received. Polyelectrolyte solutions (20% by weight in water) were diluted in distilled deionized (DDI) water. 20 mm PDADMAC (calculated using monomeric weights) was spun onto coverslips (300 mL, 15 s, 4000 rpm), then washed three times with DDI water (300 mL, 15 s, 4000 rpm). A mixture of nanoparticles and 20 mm PSS was then spun onto the PDADMAC-coated coverslips (100 mL NPs, 200 mL PSS, 15 s, 4000 rpm), followed by three washes with DI water.

The resulting sparse gold nanoparticle samples were imaged using with a 60x 1.49-NA oil-

immersion objective (APON60XOTIRFM) in an Olympus IX81 inverted microscope. Wide-field epifluorescence single-molecule microscopy was performed using 532 nm light (Crysta-laser CL532-150mW-L) as an excitation source via fiber input. Scattered light was imaged on a 512x512 pixel EMCCD (Andor iXon 897) at a frame rate of 2 Hz for 3 min. This setup had a magnification factor $a = 50$ nm pixel^{-1}. The samples were translated during imaging at a constant velocity of 0.5 mm s^{-1} with a capacitive piezoelectric xyz stage (Physik Instrumente).

2.2.6 Controlled Motion Analysis

Movies of translating gold nanoparticles were fit by the Matlab routine lsqcurvefit to the asymmetric Gaussian and SErf functions, yielding instantaneous values for all fit parameters for each localized nanoparticle in each imaging frame. To determine the Δx and Δy as a function of photons detected in Figure 2.3.6, particle tracks were separated into five-frame segments. The average center-to-center displacement/frame for each segment was calculated then subtracted from the position measurements, producing five stationary images. Then the measured x, y, and D values were binned by N and the variance of the parameter estimations were calculated for each bin.

2.2.7 Cellular Motion Imaging

Cells of the *Vibrio cholerae* classical strain O395 were grown in LB rich medium at 37 °C, then grown to turbidity (OD 0.3) at 30 °C in M9 minimal medium. *V. cholerae* cells were incubated first with intact whole-cell anti-*V. cholerae* polyclonal rabbit primary antibodies (1/2000, Abcam) for 30 min at RT, rinsed 3 times in M9, then incubated with Alexa-488 goat

anti-rabbit secondary antibodies (1/1000, Life Technologies) for 30 min at RT, and again rinsed three times in M9. This produced cells with 0-3 fluorescent labels on their surface. 2.0 mL of cells in M9 media were sandwiched between a 1.5% agarose in M9 pad and a glass coverslip. The agarose pad was wet enough to allow for two-dimensional motion within the focal plane.

The labeled *V. cholerae* cells were imaged using with a 100x 1.40 NA oil-immersion objective in an Olympus IX71 inverted microscope. Wide-field epifluorescence single-molecule microscopy was performed, using 488 nm light (Sapphire 488-50) as an excitation source with excitation powers of 100-500 mW. Scattered light was eliminated with a dichroic and a long-pass filter (Semrock Di01-R488 and Semrock BLP01-488), emitted signal was magnified by a 3.3x beam expander to a final magnification of $a = 49$ nm pixel^{-1}, and the emission was recorded on a 512x512 pixel EMCCD (Photometrics Evolve EMCCD) at a frame rate of 10 Hz for 5 min.

2.2.8 Cellular Motion Analysis

Swimming cells with a single visible fluorophore were identified for analysis, and the fluorescent label was fit in each imaging frame with the SErf and asymmetric Gaussian functions. Each fit yielded a measured instantaneous D and ϕ for each imaging frame, i, and these values were compared to an estimate for D and ϕ determined from the trajectory between the center positions of that fluorophore in frames $i-1$ and $i+1$.

2.3 Results and Discussion

2.3.1 Analytical Precision

The smallest possible variance, or Cramér-Rao lower bound, of a fitting parameter, θ_i, is equal to the inverse of the Fisher information matrix of the fitting function, $\mathbf{I}(\theta)$. Here, for both q_{SErf} and q_{AG}, it is the 7×7 square matrix:

$$[\mathbf{I}(\theta)]_{i,j} = \int \frac{1}{b^2 + q} \frac{\partial q}{\partial \theta_i} \frac{\partial q}{\partial \theta_j} dx dy \tag{2.3.1}$$

In Equation 2.3.1, q is the image function q_{SErf} or q_{AG} (Eq. 2.2.4 or Eq. 2.2.5), here used as a likelihood function. b^2 is the mean of the Poissonian noise currpting the image, in photons per pixel. Cross-terms of $[\mathbf{I}]_{i,j}$ are neglected because symmetry causes them to be near-zero or identically zero. The derivatives of q_{SErf} and q_{AG} are taken with respect to each fit parameter: $x, y, D, \sigma, \phi, N, b$ for q_{SErf} and $x, y, \sigma_x, \sigma_y, \phi, N, b$ for q_{AG}. For the diagonal elements of $[\mathbf{I}]_{i,j}$, each derivative is squared, divided by the original function plus a noise term, $(b^2 + q)$, and then integrated over all space per Equation 2.3.1. The inverse square root of this quantity is the standard deviation (RMS error), $\Delta\theta_i$, of the corresponding parameter, θ_i.

Equation 2.3.1 is applied to determine the variances $(\Delta x)^2$, $(\Delta y)^2$, and $(\Delta D)^2$ of x, y, and D, respectively, for the asymmetric Gaussian and SErf functions, and the results are given in Table 2.1 below. As described in above, solutions for the asymmetric Gaussian function are exact except for an initial separation of the integrals into high- and low-noise regeimes (a simplification used for all solutions). To find ΔD for the asymmetric Gaussian function, the calibration fit (Eq. 2.2.6) is used in the error propagation according to Equation

	Gaussian	SErf
$(\Delta x)^2$	$\frac{8b^2\pi\sigma_x^3\sigma_y}{N^2} + \frac{\sigma_x^2}{N}$	$\frac{2b^2D^2\pi\sigma^2}{N^2\left(1-\exp-\frac{D^2}{4\sigma^2}\right)} + \frac{24\sigma^2}{N(24\sigma^2-D^2)}$
$(\Delta y)^2$	$\frac{8b^2\pi\sigma_y^3\sigma_x}{N^2} + \frac{\sigma_y^2}{N}$	$\frac{192b^2\pi\sigma^6}{N^2(24\sigma^2-D^2)} + \frac{\sigma^2}{N}$
$(\Delta D)^2$	$\frac{4b^2\pi\sigma_x\left(1+\sigma_x^2\right)}{3cN^2(\sigma_x-\sigma_y)} + \frac{1+\sigma_x^2}{8cN\sigma_y(\sigma_x-\sigma_y)}$	$\frac{768b^2\pi\sigma^6}{N^2D^2} + \frac{72\sigma^4}{ND^2}$

Table 2.1: Closed-form analytical solutions to the Fisher information for three parameters of the asymmetric Gaussian and SErf fitting functions. Each solution is given as the sum of the solution in the high-noise regime and the solution in the low-noise regime, and Taylor series expansions are used for some of the SErf results, as described in the text.

2.2.6. Furthermore, for the SErf function, Taylor expansions around $D = 0$ were required to determine $(\Delta D)^2$ (high- and low-noise limits), $(\Delta y)^2$ (high-noise limit only), and $(\Delta x)^2$ (low-noise limit only); those solutions in Table 2.1 are therefore best when displacements are small. The variance in the estimation of the displacement does not noticeably differ between numberical and closed-form evaluations of the integrals. Plotted in Figure 2.3.1a), and b) are the RMS errors (Δx, Δy, ΔD) found by numerical integration of the Fisher information of x, y, and D (solid lines), and the RMS errors from the closed form solutions from Table 2.1 below (dotted lines), for the asymmetric Gaussian function (Figure 2.3.1a) and the SErf function (Figure 2.3.1b), as a function of the number of detected photons, N. These Cramér-Rau lower bounds depend on D, b, and σ in addition to N. Here, $D = 5.1$ pixels, $b = 200$ photons pixel^{-1}, $\phi = 0$, and $\sigma = 2$ pixels.

As expected, all the errors decrease monotonically with increasing N, and we find consistently that $\Delta D \gg \Delta x \geq \Delta y$. Importantly, Δy is identical (in the high-photon-count regime) to errors in the position estimators, Δx and Δy, for fitting immobile molecules with a symmetric Gaussian. Figure 2.3.1a), and b) show only slight discrepancies in the errors of the two fitting functions, so we expect the two functions to perform similarly. Good agreement is

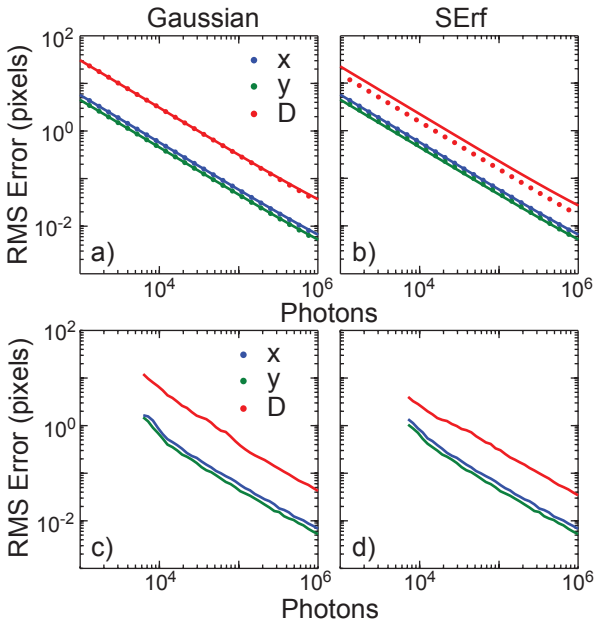

Figure 2.3.1: Precision for x, y, and D. a,b) Precision computed using Fisher information theory. Solid lines: numerical solutions to the integrals in Equation 2.3.1; dotted lines: approximate closed-form analytical solutions from Table 2.1. c,d) Precision calculated from variance of fits to 1000 simulated images. Here, $D = 5.1$ pixels, $b = 200$ photons pixel^{-1}, $\phi = 0$ and $\sigma = 2$ pixels.

observed between closed-form and numerical solutions, so one may use the analytical expressions for the error in Table 2.1 to design and troubleshoot experiments. In Figure 2.3.1b), there is a significant discrepancy between the closed-form and numerical solution for ΔD in the SErf function; this discrepancy arises from the limit of small D imposed by the Taylor expansions, and the deviation grows with D (Figure 2.3.2). For both fitting functions, as the displacement is increased, Δx increases while Δy stays fairly constant, and ΔD decreases. At the $D = 5.1$ pixels case considered in Figure 2.3.1a) and b), Δx and Δy have already diverged.

Though D cannot be determined from a fit to a conventional symmetric Gaussian function, the theoretical localization error for the symmetric Gaussian function can be found

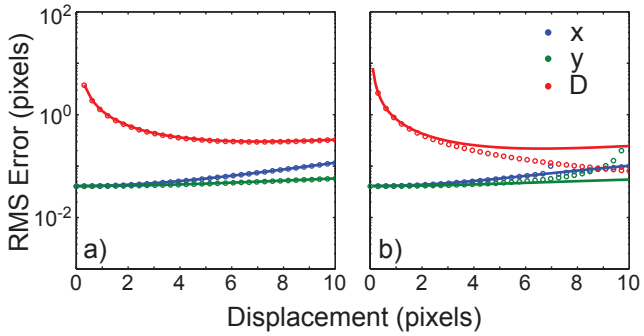

Figure 2.3.2: Precision for x, y, and D computed using Fisher information theory for the Gaussian (a) and SErf (b) functions. Solid lines: numerical solutions to the integrals in Equation 2.3.1; circles: approximate closed-form analytical solutions from Table 2.1. Here, $N = 10^5$ pixels, $b = 200$ photons pixel^{-1}, $\phi = 0$ and $\sigma = 2$ pixels.

by considering the closed-form solutions for Δx and Δy for the asymmetric Gaussian function in Table 2.1 under the condition $\sigma_x = \sigma_y = \sigma$. The error for the symmetric Gaussian function is thus $(\Delta x)^2 = (\Delta y)^2 = 8b^2\pi\sigma^4/N^2 + \sigma^2/N$, as was found by Thompson et al. in their consideration of the symmetric Gaussian fitting function. The theoretical localization precisions of the symmetric and asymmetric Gaussian functions are therefore identical when measuring stationary sources (Figure 2.3.3).

2.3.2 Numerical Precision

The analytical treatment in Section 2.3.1 gives the lower bounds for RMS error in x, y, and D based only on the choise of fitting function. To provide an estimate of the RMS errors in the localization and displacement estimation for realistic experimental conditions, simulated data was created to span a reasonable range of photon count, N, and displacement, D. For each condition, 1000 simulated images were fit by least-squares minimization to each of the fit functions, q_{SErf}, and q_{AG}, and the variance in the fit parameters was measured. Figure

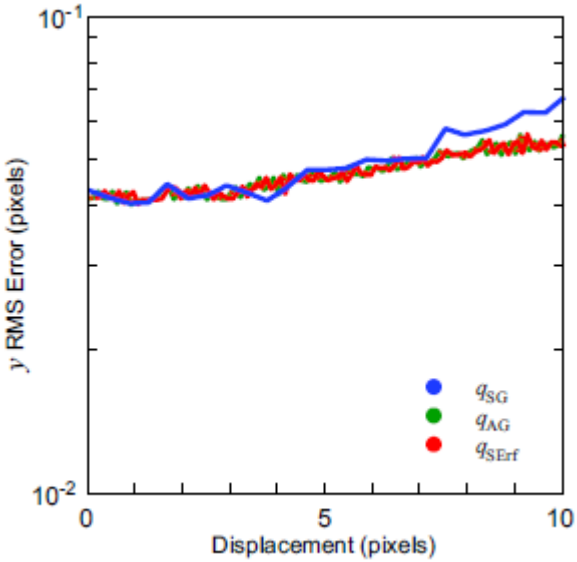

Figure 2.3.3: Comparison of symmetric, asymmetric and SErf fitting functions (blue, green and red lines, respectively) for fitting a mobile particle. As in-frame displacement, D, increases, the asymmetric Gaussian and SErf functions (q_{AG} and q_{SErf}, respectively), begin to outperform the symmetric Gaussian (q_{SG}).

2.3.1c), and d) show Δx, Δy, and ΔD for a typical displacement, $D = 5.1$ pixels. The precision of all parameters improves with higher photon count, N, and when $D > 0$, Δx and Δy diverge and we consistently find $\Delta D \gg \Delta x \geq \Delta y$.

At the limit of $D = 0$, the localization precision of the new fitting functions, q_{SErf} and q_{AG}, are the same as that of the traditional symmetric Gaussian function, q_{SG}. Furthermore, even as D becomes finite, the localization errors are still nearly identical (Figure 2.3.4). Finally, as in-frame displacement, D, increases, q_{SErf} and q_{AG} begin to outperform q_{SG} in terms of localization accuracy (Figure 2.3.3). Since the traditinoal symmetric Gaussian function, q_{SG} cannot measure D, the new functions have the added benefit of providing an instantaneous measure of velocity.

The errors Δx and Δy agree between the two fitting functions, but we find in the low

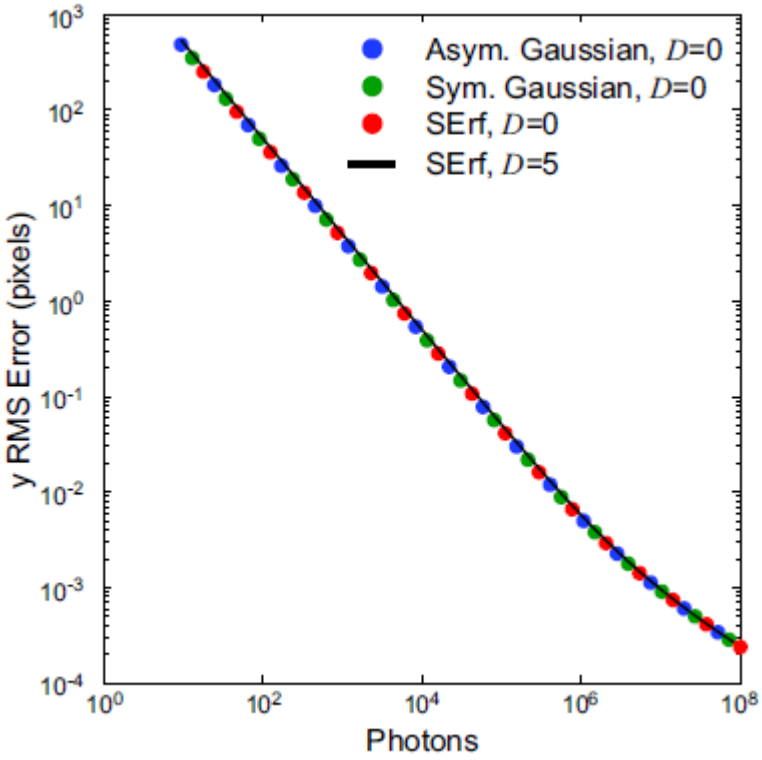

Figure 2.3.4: Comparison of precision in y (perpendicular to the direction of motion) calculated from Fisher information, based on using the symmetric Gaussian, asymmetric Gaussian and SErf fitting functions (dotted green, blue and red lines, respectively) for estimating the position of an immobile particle. The three functions all have the same performance. Solid black line: localization precision in y for fitting a mobile molecule ($D = 5$ pixels) with the SErf function; Δy for mobile particles converges at high photon count with Δy for the immobile particles.

N regime that ΔD is smaller for the SErf function, i.e. the SErf function performs better than the asymmetric Gaussian function for estimating D. For the $b = 200$ photons pixel^{-1} level considered here, which is effectively low photon count, neither fitting function is able to distinguish the particle from background noise, giving rise to random guesses within the imaged area. The data in Figure 2.3.1c), and d) is therefore limited to the $N > 6000$ regime in which the parameter variances depend on N.

Because the fitting functions proposed in this chapter are not perfect estimators of the PSF, there exists a bias in the estimation of D, in part because q_{SErf} and q_{AG} cannot distinguish between forward and reverse motion and is therefore always taken to be positive. We computed this bias from fits to simulated data by averaging over 1000 simulated data sets for each condition. Figure 2.3.5 a), and b) compare the estimated displacement, D_{fit} from fits to the asymmetric Gaussian and SErf functions, respectively, to the actual displacement, D, Here the solid lines are the average D_{fit} as a function of D, and the shaded region shows $\pm\Delta D_{fit}$. For both fitting functions, the correspondence is worst (large bias in D) at low D, and this bias decreases as D increases, finally disappearing altogether for $D > 3$. The bias in the asymmetric Gaussian fit, Figure 2.3.5 a), is attributed to errors in the $c = 0.0104$ value from the fit tot he calibration curve, Equation 2.2.6. This parabolic function is mathematically simple, but gives rise to a large bias at small D. This bias could be removed with a more elaborate calibration curve. On the other hand, Figure 2.3.5 b) shows that for the SErf function at small D the bias depends on D with a slope of -1. This bias in the estimation of D arises because least-squares fitting to the SErf function never returns a D_{fit} value smaller than some minimum value (-2 pixels), while the true displacement may be as small as 0. Displacement estimations below that value are no longer dependent on the real displacement, and instaed always return the same value on average.

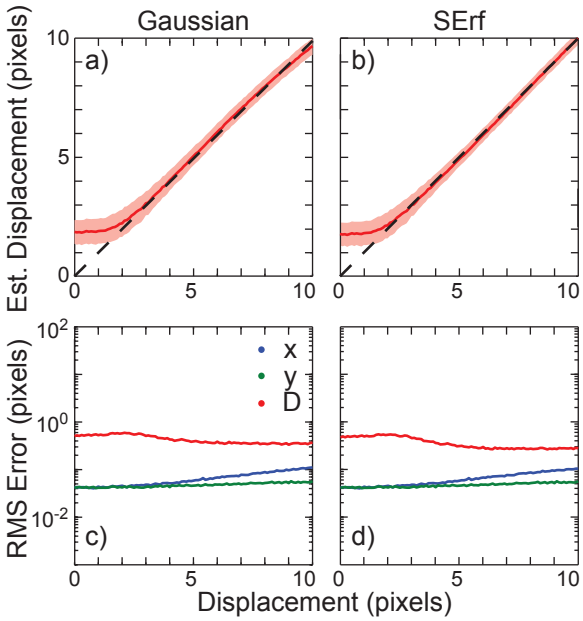

Figure 2.3.5: Effect of changes in the actual simulated displacement, D, on fit parameters computed based on fits to simulated data. a,b) Fit displacement, D_{fit}, vs. actual displacement, D. c,d) Precision for x, y, and D computed from fits to simulated data. Here, $N = 1105$ photons, $b = 200$ photons pixel^{-1}, $\phi = 0$ and $s = 2$ pixels.

Overall, Figure 2.3.5 a), and b) show that the two fitting functions have very similar biases in D.

Figure 2.3.5 c), and d) show the errors, Δx, Δy, ΔD for the asymmetric Gaussian and SErf functions, respectively, as a function of D. For both functions, Δy is mostly independent of D, and indeed this relationship, $\Delta y \neq f(D)$, is used in Section 2.3.1 to derive the closed-form precision for y in the high photon count regime. Δx and Δy are nearly identical for the two fitting functions, so the localization preicision is unaffected by choice of function. Furthermore, though D was found to be biased at small dispalcements, no bias in x, y, or ϕ was identified (data not shown). The asymmetric Gaussian and SErf functions can therefore both make accurate estimatinos of position, (x, y), and angle, ϕ, at all values of D.

2.3.3 Controlled Motion

Scattering from 26 nm gold nanoparticles immobilized on a coverslip was imaged under 532 nm laser excitation and translated at a constant velocity across the microscope viewing area with a piezo scanner. Figure 2.3.6 a), and b) show the precision in x, y, and D from fitting the recorded movies to the asymmetric Gaussian and SErf functions, respectively. Though the actual experimental velocity was in principle known, the piezo stage motion was very non-uniform over the course of the experiment. These significant variations in speed introduced some additional systematic error to the experiment. To minimize this error in D, the estimated D from the fits were compared to the average velocity of the stage over segments of only five consecutive frames. Those average values were then subtracted from the position measurements to measure the variance in the localzation estimates.

As was found analyticall and numerically, at all N, the errors followed the trend $\Delta D \gg \Delta x \geq \Delta y$, and the differences in the localization precision for both Δx and Δy between the two methods were subtle. The asymmetric Gaussian function, however, gave greater errors in D. This additional ΔD may be due to the fact that D is not a fit parameter in q_{AG} but rather related to the ratio of two fit parameters and dependent on another parameter in the calibration curve, Equation 2.2.6. Furthermore, for both fitting functions, ΔD depends only weakly on N. This may be related to the fact that D can only be well estimated if ϕ is accurately determined beforehand.

The results from this experiment are similar to the results from simulation (Figure 2.3.1 c), and d)), and only slightly worse than the lower bounds suggested by Fisher information analysis (Figure 2.3.1 a), b)). The shallower slopes of error vs. N in Figure 2.3.6 can be attributed to the background, which is no longer constant as it was in Figure 2.3.1, rather,

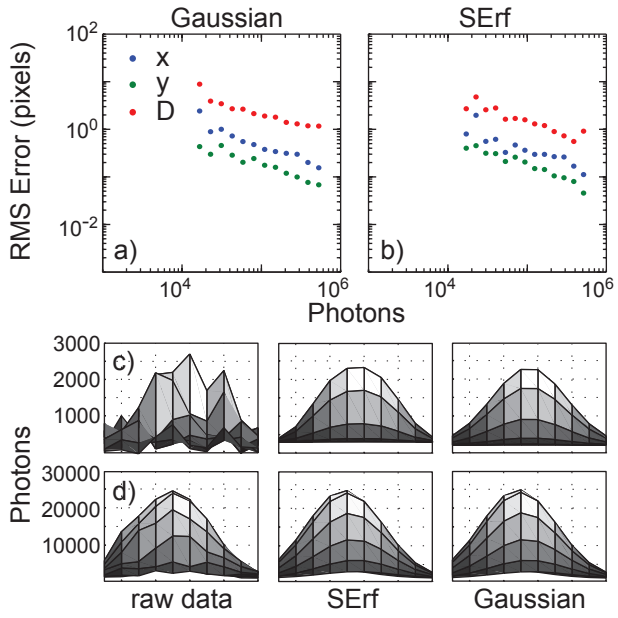

Figure 2.3.6: Experimental precision for x, y, and D measured by translating immobilized gold nanoparticles on a motorized stage. a,b) Precision using asymmetric Gaussian and SErf fits, respectively. $b > 100$ photons pixel^{-1}, $c = 0.0104$. (c). d) Example data at low photon counts ($N = 4.56 \times 10^4$) and at high photon counts ($N = 5.13 \times 10^5$), respectively. Left panels: experimental data, center panels: SErf fits, right panels: asymmetric Gaussian fits.

in these experiments, there is a non-negligible increase in b as N increases.

Figure 2.3.6 c), and d) show typical experimental data from the low photon-count ($N = 4.56 \times 10^4$) and high-photon-count ($N = 5.13 \times 10^5$) regimes of this experiment, respectively, as well as the fits of each data set to the asymmetric Gaussian and SErf functions. The asymmetric Gaussian fits are too sharply peaked at their center, whereas the SErf function fits with smaller residuals.

2.3.4 Cellular Motion

The fit functions q_{Serf} and q_{AG} proposed in this chapter can be used to characterize directional motion of small objects such as bacterial cells. Here, we excited fluorescent markers on the surface of *V. cholerae* cells with a 488 nm laser and record images as the cells move in two dimensions within a thin layer of media between a coverslip and an agarose pad. Figure 2.3.7 a) presents the results of one typical trajectory. Here, a cell marked with a single Alexa-488/antibody label is tracked for 65 imaging frames, during which time it diffuses randomly (frames 1-7), then moves consistently at an angle of $\phi \approx 135°$ from the horizontal (frames 8-41), and then finally resumes diffusive motion (frames 42-65). Though the average displacement, D, and angle, ϕ, can be estimated from the trajectories, fits to q_{SErf} and q_{AG} allowed for instantaneous measurements of the velocity and directionality.

During the intermediate frames, the cell moved fast enough that fits to the SErf function and the asymmetric Gaussian function yielded meaningful data. Here, we expect that the instantaneous D estimated from the fit to be reasonable; unfortunately, the average displacement calculated from the average of changes in position was a bad approximation and so the comparison between instanteous D and average D was poor (Figure 2.3.8). Fortunately, the average angle of motion, ϕ, was more accurately esimated from the center-to-center average position over three-frame segments, and a comparison of the instantaneous angle from fits to q_{AG} and q_{SErf} to the average angle from frame-to-frame movements is presented in Figure 2.3.7 a), and b), respectively. Here, the dashed lines show perfect correspondence, the color scale indicates the instantaneous displacement from the fit, and the size of the points is inversely related to the square of the 95% confidence interval for the fitted value of the angle, ϕ. For both functions, there is good agreement between instantaneous and frame-to-frame

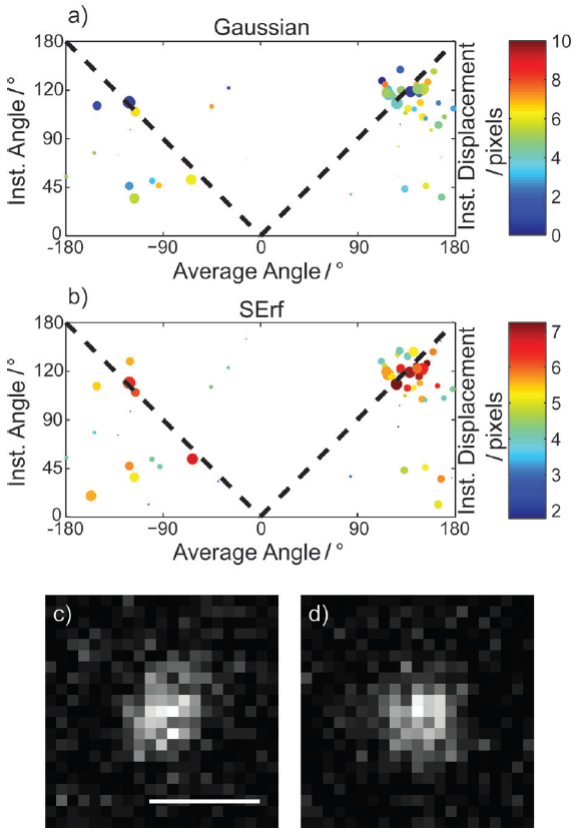

Figure 2.3.7: Analysis of experimental images of a *V. cholerae* cell in motion in the 2D plane. a,b) Comparison of instantaneous direction of motion, ϕ, from a fit to the asymmetric Gaussian function (a) and the SErf function (b), to average direction of motion from the center-to-center average angle over three-frame segments. The dashed lines (a) show perfect correspondence, the color scale indicates the instantaneous displacement for each measurement from the fit, and the size of the points is inversely related to the square of the 95% confidence interval for the fit to the angle, ϕ. c) Representative image of a fluorophore undergoing directed motion (frame 40). Scale bar: 500 nm. d) Representative image of a fluorophore undergoing diffusive motion (frame 56)

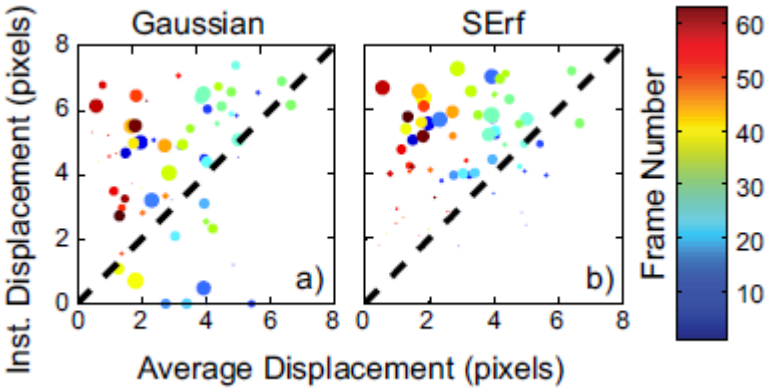

Figure 2.3.8: Analysis of experimental images of a *Vibrio cholerae* cell in motion. Comparison of instantaneous displacement, D, from a fit to (a) the asymmetric Gaussian function ($c = 0.0104$) and (b) the SErf function, to average direction of motion, from the center-to-center average displacement over three-frame segments. The dashed lines show perfect correspondence, the color scale indicates the frame number, and the size of the points is inversely related to the square of the 95% confidence interval for the fit to the angle, ϕ.

angle measurements for the directed motion at $\phi \approx \pm 135°$, where D is large. As expected, in the small-D case (random cellular diffusion), isotropic blurring dominates and the methods described here produce incorrect instantaneous angle measurements. Accordingly, Figure 2.3.7 a), b) show no correspondence for the fits where D is small.

Figure 2.3.7 c) shows the fluorescent image of the cell during the directional motion portion of its trajectory. The fluorescent label on the cell appears to be asymmetric and elongated in the direction of motion (135° from the horizontal), and the displacement and angle can both be estimated from fits to q_{AG} and q_{SErf}. Figure 2.3.7 d) shows the fluorescent image of the cell during the random portion of its trajectory. Here, the asymmetry vanishes and the resulting PSF is a symmetric Gaussian with increased standard deviation. The directional motion fitting functions, q_{AG} and q_{SErf} cannot accurately determine D or ϕ here.

2.4 Conclusions

In this chapter, we have introduced two new fit functions, the asymmetric Gaussian function and the Sum of Error Functions (SErf) function, that can each measure the instantaneous position, directionality and velocity (in-frame displacement) from a fit to the fluorescent image of a single molecule undergoing directional motion. Both functions have the benefit of being easily integrated into standard single-molecule fluorescence microscopy fitting routines. The results from Fisher information analysis provide a lower bound to the precision of each parameter in the fitting function. Though deriving the closed-form solutions in Table 2.1 required several simplifications, the true precision of each parameter was easily attained in Section 2.3.2 by numerically solving the Fisher information integrals. Overall, we find that the two fitting functions perform very similarly, and that the errors consistently follow the trend $\Delta D \gg \Delta x \geq \Delta y$, where x is the direction of motion. Here, the errors in the position estimator for the y direction (perpendicular to the motion) are identical to those derived for fitting stationary molecules to a symmetric Gaussian function, indicating that no localization precision is lost with the new functions relative to standard single-molecule fitting algorithms.

The simulations in Section 2.3.2 show that the two fitting functions perform similarly in nonlinear least squares minimization, though in the low photon count regime, the SErf function provides more precise estimates of position and displacement. Unfortunately, though estimations of position and angle are unbiased, we have found a systematic bias when estimating in-frame displacement by fitting to either function for small D. This regime could be avoided in applications by choosing experimental conditions that result in D>3 pixels/imaging frame and by carefully calibrating the asymmetric Gaussian function. The results from

simulations are similar to the Cramér-Rao lower bounds, and the slight increase in error in the simulations can be attributed to our use of nonlinear least squares minimization, which is superior to maximum likelihood estimation (MLE) in speed and simplicity, but inferior to MLE in precision. Though these simulations were done with very bright objects in a very noisy background, applications to lower photon count and noise cases does not alter the results qualitatively. Results for this regime, which was recently considered by Yüce et al., [55] are presented in Figures 2.4.1 and 2.4.2.

The application of the fitting functions to the motion of gold nanoparticles in Section 2.3.3 verified our method in the context of a controlled experiment, and indeed, we find that the new fitting method provides the velocity and directionality of each nanoparticle in each image without degrading the localization precision relative to the symmetric Gaussian fitting function. Though these nanoparticle tracking experiments are not perfect controls due to variations in the stage velocity, the experiments validate our analytical framework, yielding experimental errors similar to the errors in simulations, and only slightly worse than the Cramér-Rao lower bounds.

In Section 2.3.4, we applied our new fitting functions to the motion of a live *V. cholerae* cell moving in the imaging plane of a microscope. To the best of our knowledge, this is the first example of instantaneous direction and velocity measurement of a freely moving bacteria cell. We observed both random, diffusive motion and steadily directional motion, and found that the instantaneous displacement and angle were only well measured by the fitting functions in the case of directional motion, yet the localization precision was not compromised in any case. The mobile single-molecule fitting functions proposed in this chapter are therefore appropriate for any situation where there is directional motion for some part of a trajectory.

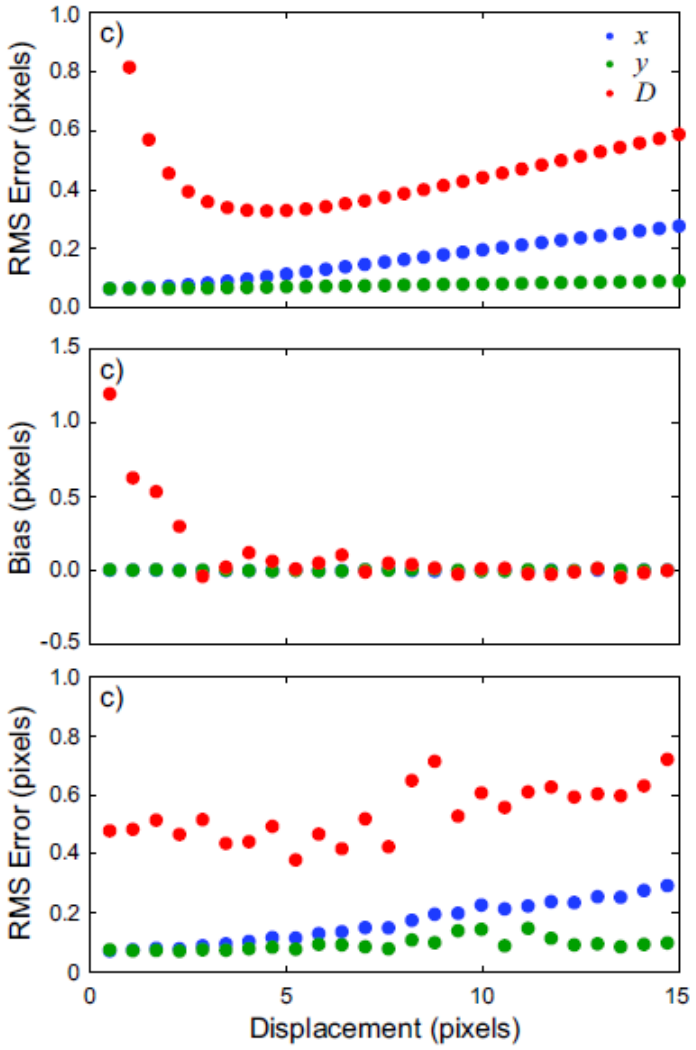

Figure 2.4.1: Extension of the SErf fitting function analysis to the low photon count, low noise regime. (a) Precision for x, y, and D computed using numerical solutions to the Fisher information integrals. (b) Bias in estimation of x, y, and D from fitting simulated data. (c) Precision for x, y, and D from variance of fits to 1000 simulated images. Here, $b = 4$ photons pixel^{-1}, $N = 750$ photons, and $\sigma = 1.2$ pixels.

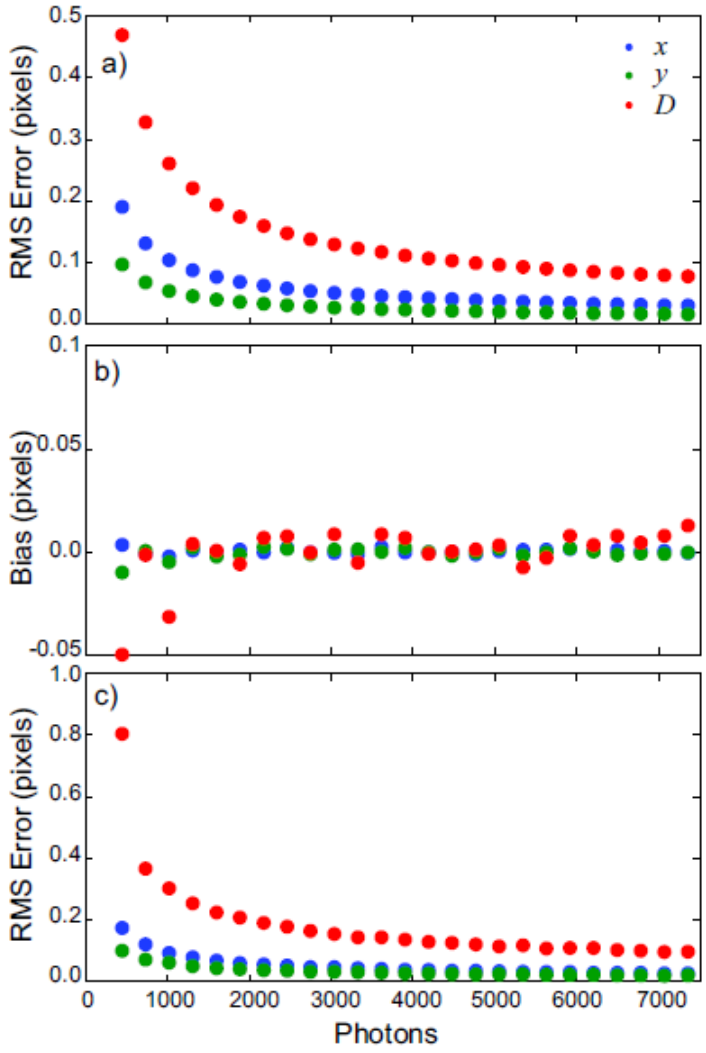

Figure 2.4.2: Extension of the SErf fitting function analysis to the low photon count, low noise regime. (a) Precision for x, y, and D computed using numerical solutions to the Fisher information integrals. (b) Bias in estimation of x, y, and D from fitting simulated data. (c) Precision for x, y, and D from variance of fits to 1000 simulated images. Here, b increases linearly with N from 3 to 12 photons pixel^{-1}, $\sigma = 1.2$ pixels, and $D = 7$ pixels.

In conclusion, presented here are two methods for measuring positions and in-frame velocities of single point light sources. By explicitly accounting for the motion of the fluorophore, these fitting functions permit even mobile single molecules to be fit in standard post-processing routines. These methods promise to improve single-molecule tracking as fewer data sets must be rejected, and tracking algorithms can also be refined to use the instantaneous displacement and directionality as a predictor for the next data point. In a crowded environment, these two factors will make tracking algorithms less ambiguous. These methods do not require unique experimental design, and can even be applied to pre-existing data to recover information previously unnoticed. The precision of the two new fitting functions is nearly identical under all of the conditions we have considered, though least-squares fitting with the SErf function is 4.5% slower than least-squares fitting with the asymmetric Gaussian function.

The directional-motion fitting functions presented in this chapter have the potential to inform on the dynamics of living, functioning cells. In addition, we envision that these methods can enable single-molecule imaging of bio-molecules within live, moving cells, since cellular imaging with the new fitting functions, like in Section 2.3.4, can provide a moving frame of reference for concurrent intracellular single-molecule imaging. This is particularly promising for studies of cellular chemotaxis and signaling, though it can be applied in any situation where directional motion is expected.

CHAPTER III

Resolving Fast, Confined Diffusion in Bacteria with Image Correlation Spectroscopy

The work presented in this chapter has been published in the following paper:

Rowland, D. J., Tuson, H.H., and Biteen, J. S. Resolving Fast, Confined Diffusion in Bacteria with Image Correlation Spectroscopy. *Biophys. J.* 110, 2241-2251 (2016).

3.1 Introduction

Microscopy has long been a staple technique in biological research. In particular, based on the development of techniques for selectively labeling specific cellular components, fluorescence microscopy has enormous value for elucidating intracellular biology. [57] More recently, the ability to visualize a single molecule at a time has improved the localization precision below the standard diffraction limit of light. [11–13] In live cells, the function of a protein in a biological process can be inferred from its rate of diffusion under different chemical or genetic conditions. [8, 58–61] Traditionally, optical measurements of subcellular diffusion have been

done using fluorescence recovery after photobleaching (FRAP), [62–64] but single-molecule imaging techniques like single-particle tracking (SPT) are being increasingly used to precisely evaluate the motion of a diffusing biomolecule. [65–67] SPT is achieved by connecting a series of single-molecule positions over time. In an optimal SPT experiment, the camera integration time is fast enough that within one frame, the target diffusing molecule does not produce blur by moving far compared to the diffraction limit of light. Acquiring long trajectories improves the statistical significance, but unfortunately, there is a tradeoff due to the finite fluorescence yield of a single-molecule probe: increasing the illumination intensity to enable single-molecule detection in a shorter imaging frame time will shorten the trajectory lengths. This is a particular problem for the most common fluorescent labels in live-cell intracellular imaging, fluorescent proteins. [57, 68] These two conflicting requirements of fast imaging and long tracks limit the total range of measurable diffusion rates; this range limit is an important issue when heterogeneities yield a range of diffusion coefficients that are measured simultaneously. If a pulsed illumination source is available, stroboscopic illumination can be applied to decrease the amount of blur without increasing the frame rate or decreasing the trajectory lengths. [69] However, confining diffusion to a small volume-for instance, within a cell or organelle-introduces a further constraint: the data acquisition rate must be faster than the time it takes for the diffusing molecule to explore the entire confinement volume.

Stroboscopic illumination does not increase the data acquisition rate, and so the maximum measureable diffusion coefficient for a molecule in a confined volume is still limited by the maximum camera frame rate. Additionally, in all cases, because SPT relies on a tracking algorithm to construct trajectories for single diffusing molecules, these trajectories should not overlap. Overall, SPT is best suited for characterizing a collection of sparse and homogeneously diffusing molecules in unconfined environments.

Alternatives that can overcome some limitations of the localization-based SPT analysis include spatiotemporal image correlation spectroscopy (STICS), [30,70–72] which has previously been used both in vitro [72] and in live cells, [73] and the related methods of k-space image correlation spectroscopy [32,74] and superresolution optical fluctuation imaging. [75] All of these methods compute the correlation function of an entire fluorescence imaging movie instead of relying on localization and tracking, and like fluorescence correlation spectroscopy, these powerful image correlation approaches can be extended to measure 2D maps of heterogeneous diffusion [70,76] and very fast diffusion. [77] Particle image correlation spectroscopy measures diffusion dynamics by computing a similar correlation function from data that have already been processed with the Gaussian localization used in SPT and other methods. [78] In STICS, the spatial cross-correlation at some time lag, τ, is the average of the spatial cross-correlations of all pairs of images separated by this τ and has a width (termed the image-meansquared displacement (iMSD)) that increases with t. The iMSD-vs.-τ function can be fit to calculate the diffusive characteristics of a collection of fluorescent molecules.

In this chapter, we use STICS to resolve fast, confined motion in a widefield imaging microscope. If a confocal microscope is available, extremely fast diffusion ($>$100 mm2/s) can be measured with raster image correlation spectroscopy, which uses a scanning mirror to eliminate image blur. [77] Here, we instead focus on developing STICS to extend the capabilities of a single-molecule imaging microscope beyond the limits of SPT (maximum measurable diffusion coefficient, ~ 10 mm^{2}s^{-1} [79]) to an order-of-magnitude-faster motion. Unlike SPT, STICS does not require accurate position determinations for delocalized, fast-moving molecules. Rather, we show here that STICS analysis can explicitly account for in-frame motion blur based on a modification to the mathematical formulation. Conventionally, STICS assumes that molecular diffusion is described by a Gaussian distribution of step sizes,

but both in-frame motion and tight confinement will alter the shape of this distribution. In-frame motion corrupts the distribution because the time lag between two frames becomes poorly defined when the integration time or the diffusion rate is large. Tight confinement further distorts the distribution, because this boundary condition excludes the largest step sizes. The artifacts introduced by confinement and in-frame motion were not encountered in previous applications of image correlation spectroscopy, because those experiments focused on slow motion [76, 80–83], or else minimized pixel dwell time, as in raster image correlation spectroscopy [77]. On the other hand, here in the regime of fast, confined motion, the two independent effects of blur and confinement manifest themselves as two independent diffusion-coefficient measurement biases.

Here, we use STICS to measure the rapid subcellular dynamics of a freely diffusing protein in the highly confined interior of a bacterial cell for the first time, to our knowledge. We describe the measurement biases that arise from confinement and fast motion by simulating unconfined diffusion and diffusion confined to bacteria-sized cylinders, both with and without in-frame blur. We develop a method to correct these diffusion-coefficient measurement biases when molecules diffusing up to 15 $\mu m^2 s^{-1}$ are confined inside bacterial cells as small as 1 μm in length. We find that in-frame motion adds a positive bias to the diffusion-coefficient estimation and that confinement adds an independent, nonmonotonically varying bias. We show how both biases can be removed by directly computing the variances of the STICS correlation function. Finally, we apply STICS to experimental fluorescence microscopy movies of freely diffusing mMaple3 fluorescent protein in the *Escherichia coli* cytoplasm, and we find a diffusion coefficient, $D = 9.6 \pm 1.0$ $\mu m^2 s^{-1}$, for these molecules. This value agrees well with previous fluorescence recovery after photobleaching measurements of fluorescent proteinmobility in the *E. coli* cytoplasm ($D = 6.1 - 14.1$ $\mu m^2 s^{-1}$ [66]), as well

as with SPT measurements with very short camera integration times (0.75-4 ms, $D = 7.3$ $\mu m^2 s^{-1}$ [79]). In this last case, a shortened camera integration time minimized the data loss due to in-frame motion [79]. In this work, we instead analyze data taken with typical camera imaging rates (25-100 frames s^{-1}) by modifying the STICS protocol to explicitly account for confinement and in-frame motion. Overall, this relaxation of the experimental constraints enables direct, single-molecule-based diffusionmeasurements in a conventional wide-field single-molecule imaging setup [84] of samples that necessarily contain densely packed fluorophores, systems that contain multiple interacting diffusive populations, and systems that exhibit fast confined diffusion in cellular and subcellular compartments and nanomaterials [28].

3.2 Theory and Experimental

3.2.1 Simulations

Three-dimensional diffusion was simulated by generating step sizes from a zero-mean normal distribution with variance of 200 nm^2 in each of the three dimensions independently. This variance produces a root mean-square (RMS) displacement that is $<10\%$ of the waist of the smallest microscope point-spread function (σ_{PSF}) considered, enabling us to simulate smooth motion blurring. For simulations of diffusion confined to a cylinder, candidate steps that fell outside of the cylinder boundary were elastically reflected. The cell boundary is expected to act as an impermeable wall that does not interact with cytoplasmic proteins like the mMaple3 fluorescent protein used here. These 3D trajectories were projected onto the xy plane to simulate two-dimensional (2D) imaging.

The pixel brightness value in each subframe was calculated using a symmetric Gaussian function centered at the position of the diffusing molecule. For one-dimensional (1D) motion, the step-size distribution variance is $2Dt_{\text{frame}}$, where D is the diffusion coefficient and the camera integration time is t_{frame}. Motion with any desired D was therefore simulated by combining the appropriate number of simulation subframes; i.e., any D and t_{frame} can be simulated by taking the average of $n = 2Dt_{\text{frame}}/200 \text{ nm}^2$ subframes. The pixel width was set to 49 nm for all simulations for consistency with experiments. The 2D probability distribution of step sizes from one frame to the next is a normal distribution with zero mean and a variance of $4Dt_{\text{frame}}$:

$$p = N\left(0, 4Dt_{\text{frame}}\right) \tag{3.2.1}$$

The squared step sizes and correlation functions were calculated for the SPT and STICS methods, respectively, as described below. For the bias estimation simulations (see Figs. 3.3.2 and 3.3.4), incremental iterations were used to more precisely compute the STICS correlation function (see STICS analysis, below, for details). Once $\sim$ 50,000 total image frames were simulated, D was estimated with STICS and we computed the bias on this value, $(\theta/\theta_0) - 1$, from the ratio of the estimated value, θ, to the true value, θ_0. For the error estimation simulations (see Fig. 3.3.6), both STICS and SPT were performed on identical simulated raw data to which zero-mean white noise had been added with the Matlab built-in function *randn* to achieve a signal/noise ratio (SNR) of 6 for immobile molecules. The effective SNR, however, is reduced by motion blur and will be $<$6. The simulated data set used in this study consisted of 1200 frames of contiguous motion.

3.2.2 SPT analysis

Candidate molecular locations were found in simulated single-molecule movies by following a band-pass filter in the Fourier domain with a watershed algorithm. These positions were refined to the local center of mass within a 15 pixel x 15 pixel window. The refined locations were then used as the initial guess for a least-squares fit of the raw image of the molecule by a seven-parameter generalized bivariate Gaussian function using Matlabs built-in function *lsqcurvefit*. The center of this fit was taken to be the position of the diffusing molecule. Because only a single molecule was present in our simulations at any time, the time series of localizations was used as the single-particle trajectory, with no need for a more advanced tracking algorithm. Infrequently, the analysis of a single-molecule image incorrectly indicated more than one molecule, in which case all putative molecules from that image were ignored. The MSDs for the first 15 possible time lags, τ, were then computed and these results were fit to a model for 1D confined diffusion inside an infinite square well [65]:

$$MSD(\tau) = \frac{L^2}{6}\left(1 - \frac{96}{\pi^4}\sum_{\text{n odd}}\frac{1}{n^2}\exp\left[-\left(\frac{n\pi}{L}\right)^2 D\tau\right]\right) + B. \tag{3.2.2}$$

Here, L is the confinement length and B is a constant offset equal to the measurement variance of the position of an immobile molecule. Only the MSD curve computed in the direction of the cylinder long axis was considered because with the diffusion coefficients considered here, the diffusing molecule may explore the entire short axis of the cell in a single frame. 1D motion was recovered from 2D position data by projection onto the long axis of the cell.

3.2.3 STICS analysis

For both simulated and experimental data, the full time-space correlation functions were computed using the Fourier transform method for convolutions, as described previously [69, 81]. The width of the correlation function increases with time lag, and Gaussian fitting can estimate these widths. The time lag series of widths is called the $iMSD$, which for our model of square confined diffusion [65] has the functional form

$$iMSD(\tau) = \frac{L^2}{6}\left(1 - \frac{96}{\pi^4}\sum_{\text{n odd}}\frac{1}{n^2}\exp\left[-\left(\frac{n\pi}{L}\right)^2 D\tau\right]\right) + C. \tag{3.2.3}$$

Here, C is a constant offset proportional to the diameter of the image of the diffusing molecule. The sum was truncated after the change in the function value dropped below 10^{10}, a condition which was always satisfied with fewer than 10 terms. Only the $iMSD$ curve computed from the cell long axis was considered, because for the fast diffusion coefficients considered here, the diffusing molecule explores the entire short axis of the cell in a single image frame. For the experimental bacterial cell data analysis, one $iMSD$ curve was calculated for each of 87 movies, which had ~ 4000 imaging frames each. The function $iMSD(t)$ was fit to Eq. 3.3.9 for the first 15 values of τ (with the Matlab function *lsqcurvefit*) to estimate the diffusion coefficient, D.

Each frame of the simulated movies was symmetrically padded to double the original width with a padding value of zero because the data were generated with zero intensity offset. For the experimental data, the phase contrast image of the bacterial cell was used to automatically compute, with a valley filter, a selection mask that describes the position of each bacterium. Pixels that fell outside of this mask were replaced with the mean intensity value inside the selection mask. Because the absolute offset of the real bacterial cell image

is ill-defined or weakly measureable, the experimental bacterial cell data were padded with the average intensity value inside the cell.

3.2.4 Bacterial strains and growth conditions

DH5α *E. coli* cells containing a pBAD plasmid encoding the fluorescent protein mMaple3 under the arabinose promoter [28] were struck out from a freezer stock on a 1.5% Luria Bertani (LB) agar plate containing 50 mg mL^{-1} ampicillin and incubated overnight at 37 °C. A single colony from the plate was used to inoculate 2 mL LB containing 50 mg mL^{-1} ampicillin. This culture was incubated overnight ($\sim$ 16 h) at 37 °C with shaking. Following incubation, the culture was diluted 1:50 into M9 minimal medium containing 0.4% glycerol as the carbon source. The culture was incubated at 37 °C with shaking to an optical density at 595 nm of $\sim$ 0.35, at which time mMaple3 expression was induced by adding arabinose to a final concentration of 0.01%. After induction, the culture was incubated at 37 °C with shaking for an additional 2 h. Cells were pipetted onto a 2% agarose/M9/arabinose pad and inverted onto a plasma-etched coverslip for imaging, as previously described [85].

Alternatively, longer cells were grown by using cephalexin, a β-lactam antibiotic that blocks cell division but allows cell growth [85]. Here, the overnight culture was diluted 1:100 into LB and incubated at 37 °C with shaking to an optical density at 595 nm of $\sim$ 0.5. The culture was then centrifuged for 5 min at 5000 g, the supernatant was decanted, and the cell pellet was resuspended in an equal volume of M9 medium containing 0.4% glycerol and 0.01% arabinose. Cephalexin was added to a final concentration of 60 mg mL^{-1}. The culture was then incubated at 37 °C with shaking for an additional 30 min. Cells were pipetted onto a 2% agarose/M9/arabinose pad (which also contained 60 mg mL^{-1} cephalexin) and inverted

onto a plasma-etched coverslip for imaging.

3.2.5 Imaging conditions

Bacterial samples were imaged at room temperature using wide-field epifluorescence microscopy in an Olympus IX71 inverted microscope with a 100x, 1.40 NA oil immersion objective (in Immersol 518F immersion oil, Carl Zeiss, Oberkochen, Germany) and appropriate excitation, emission, and dichroic filters (LL01-561, BLP01-561, and Di01-R561, respectively, Semrock, Rochester, NY). After a 3x beam expander, a Photometrics (Tucson, AZ) Evolve EMCCD camera with >90% quantum efficiency captured the images at 25-100 frames s^{-1}. Each camera pixel corresponds to a 49 nm x 49 nm area of the sample. Fluorescence of mMaple3 in the cells was photoswitched from green to red photon emission using a 406 nm laser (Cube 406-100, Coherent, Santa Clara, CA), coaligned with the 561 nm fluorescence excitation laser (Sapphire 560-50, Coherent). The samples were illuminated with low laser power densities (0.07 mW mm^{-2} and 0.01 mW mm^{-2} for the 406 nm illumination and the 561 nm illumination, respectively). Both laser beams were circularly polarized with a quarter waveplate (AO15Z 1/4 556 and AO15Z 1/4 408, respectively, Tower Optical, Boynton Beach, FL). During imaging, the cells were given a 1- to 10-ms dose of 406 nm light every $\sim$ 30 s. Movie acquisitions lasted 2-5 min each.

Due to the small ($\sim$ 1 mm) cell thickness, a low background noise is maintained even when single-molecule imaging in bacteria cells is done with a wide-field microscopy configuration [86]. Thus, this work describes the analysis of wide-field microscopy, though these methods could be readily applied to total internal fluorescence microscopy.

3.2.6 Characteristic motion, M_{C}: a unit-free parameter describing the amount of in-frame motion and the extent of confinement

Because of the high dimensionality of the simulation parameter space, which includes, e.g., camera integration time, pixel size, and magnification, it is desirable to have an invariant description of the degree of in-frame motion. To this end, we renormalize the diffusion rate, D, to produce a unit-free characteristic motion parameter, M_{C}. This M_{C} simplifies the presentation of the simulation results and can highlight consistencies among parameter sets. M_{C} is proportional to the squared distance, Dt_{frame}; traveled in each captured image and inversely proportional to the variance of the point-spread function of the diffusing molecule (σ^2_{PSF}). For consistency among dimensionalities, the factor $2/d$ is applied, where $d = 2$ for 2D image data:

$$M_{\mathrm{C,unconfined}} = \frac{2Dt_{\mathrm{frame}}}{d\sigma^2_{\mathrm{PSF}}} \tag{3.2.4}$$

For confined diffusion, doubling the confinement length, L, is equivalent to halving the RMS in-frame displacement value. In the case of confined diffusion, the unconfined characteristic motion, $M_{C,\mathrm{unconfined}}$, is therefore modified by a scaling factor that describes the degree of confinement, the unit-free factor L^2/L^2:

$$M_{\mathrm{C,confined}} = \frac{2Dt_{\mathrm{frame}}}{d\sigma^2_{\mathrm{PSF}}}\frac{L_0^2}{L^2} \tag{3.2.5}$$

where L_0 is a reference confinement length chosen here to be the average cell length, 3 mm. In Figs. 3.3.2, 3.3.4, and 3.3.10, we rescale D based on the simulation parameters (Figure 3.3.3) according to these definitions of M_{C}. The overlap seen in these figures demonstrates the validity of Eqs. 3.2.4 and 3.2.5.

3.2.7 The effect of in-frame motion on STICS

In STICS, the correlation function, G, is the convolution of the step-size probability distribution, p, with W_{PSF}, the approximately Gaussian microscope point-spread function [30]:

$$G(\xi, \chi, \tau) = \frac{\gamma}{N} p(\xi, \chi, \tau) \otimes W_{\mathrm{PSF}}(\xi, chi) \tag{3.2.6}$$

Here, ξ and χ are displacements in x and y, respectively, for each time lag τ. N is the average number of molecules present in each frame and γ is a geometric factor that accounts for an underrepresentation of intensity fluctuations due to the illumination beam shape. The 2D step-size distribution, p, of unconfined Brownian diffusion is traditionally assumed to be normally distributed:

$$\mathrm{Var}(p) = 2dD\tau, \tag{3.2.7}$$

where d is the dimensionality and D is the diffusion coefficient. When Eq. 3.2.7 applies, G in Eq. 3.2.6 will be a Gaussian distribution, because it is the convolution of two Gaussians. However, when there is significant in-frame motion, the diffusing molecule will have moved a nonnegligible distance during the imaging integration time, t_{frame}. In this case of significant inframe motion, the step-size distribution, p, is not only a function of the average time lag, τ_0; p also depends on a range of effective time lags, τ_{eff}, in the range of $\tau_0 - t_{\mathrm{frame}} < \tau_{\mathrm{eff}} < \tau_0 + t_{\mathrm{frame}}$. This mixture of step lengths from nonidentical time lags causes p to become the sum of nonidentical normal distributions, and importantly, this p is not a normal distribution. When such in-frame motion is present, the correlation function, G_{blur}, is the convolution of

the point-spread function with a weighted average of Gaussians:

$$G_{\text{blur}} = \frac{\gamma}{N}\left[\left(\int_{\tau_0-t_{\text{frame}}}^{\tau_0+t_{\text{frame}}} g(\tau_{\text{eff}})d\tau_{\text{eff}}\right)^{-1} \times \int_{\tau_0-t_{\text{frame}}}^{\tau_0+t_{\text{frame}}} g(\tau_{\text{eff}})N(0,4D\tau_{\text{eff}})d\tau_{\text{eff}}\right] \otimes W_{\text{PSF}}(\xi,\chi). \tag{3.2.8}$$

Here, $N(0, 4D\tau_{\text{eff}})$ is a zero-mean normal distribution with variance, $4D\tau_{\text{eff}}$, and weighting factor, $g(\tau_{\text{eff}})$. The point spread function, W_{PSF}, is an Airy Disk, which we approximate here by a Gaussian function [15].

3.2.8 The effect of confinement on STICS

To simulate confinement, we imposed reflective boundary conditions on the step size distribution, p. In the limit of fast diffusion and small confinement, p approaches a uniform distribution. In general, if zero-flux boundary conditions are imposed on ξ and χ in the distribution, $p(\xi, \chi, \tau)$, which is a symmetric function centered on zero, then the correlation function, G, is modified:

$$G_{\text{conf}} = \frac{\gamma}{N}\left[2\sum_{n,\text{int.}} p(\xi+nL, \chi+nL, \tau)\right] \otimes W(\xi,\chi), -\frac{L}{2} \leq \xi,\chi \leq \frac{L}{2} \tag{3.2.9}$$

Overall, the effect of Eqs. 3.2.8 and 3.2.9 is a deviation of the Gaussian function in Eq. 3.2.6 to a degree that depends directly on the amount of in-frame motion or confinement.

3.2.9 Estimation of the diffusion coefficient by direct calculation of the variance

According to Eq. 3.2.7, D is proportional to the variance of the step size distribution, p. If p describes ideal free diffusion, then p is Gaussian-shaped. The correlation function, G, computed by STICS for this type of data would then also be Gaussian. However, G is not generally Gaussian due to in-frame motion and confinement, as described above. In its traditional implementation, STICS estimates the correlation function variance, $iMSD$, as being the variance parameter of a least-squares fit of G to a Gaussian function. If G is not Gaussian, then fitting G to a Gaussian will not accurately measure the variance. In this case, the inaccuracy can be removed by directly computing the variance according to its definition:

$$iMSD_x = Var(G_\xi) = \sum_i \xi_i^2 G_\xi \tag{3.2.10}$$

$$iMSD_y = Var(G_\chi) = \sum_i \chi_i^2 G_\chi \tag{3.2.11}$$

Here we consider in-plane diffusion in each of the two dimensions, x and y, separately. The marginal distributions G_ξ and G_χ for displacements in x and y, respectively, are computed from the 2D correlation function, G, by discrete summation.

$$G_\chi = \sum_\xi G(\xi, \chi, \tau) \tag{3.2.12}$$

$$G_\xi = \sum_\chi G(\xi, \chi, \tau) \tag{3.2.13}$$

This separation allows diffusion along the long and short axes of bacterial cells to be

treated independently. We fixed the orientation of the cylindrical boundary in the simulations to lie parallel to the x axis and directly calculated the variance for the simulated data. This direct variance calculation method was not used to estimate the $iMSD$ of the experimental data because the measurement precision was too low to be useful for this analysis, especially after the interpolation involved in image rotation.

It should be noted that this method for finding the variance requires the distribution, G, to be normalized at every value of τ. In general, however, the integral of G with respect to ξ and χ is not unity for each time lag. In principle this integral can be normalized by dividing it by the discrete integral of G or by dividing it by the known prefactor $\frac{\gamma}{N}$ in Eq. 3.2.6. In our simulations, we ensured a priori that G was normalized at every τ by fixing $\frac{\gamma}{N} = 1$.

3.3 Results and Discussion

3.3.1 Effect of in-frame motion on single-particle tracking

We simulated diffusion with 10 different values of D in 3 μm long cylinders with reflective boundary conditions. These movies were processed by single-particle tracking (SPT), and the diffusion coefficient, D, for each trajectory was estimated from the MSDs. Each measured D was scaled to the corresponding unit-free characteristic motion, M_{C}, according to Eq. 3.2.5. Because not all single-molecule images could be fit in our regime of finite signal to noise, trajectories were truncated. We computed for each M_{C} the percent recovered displacements, $n_1/(n_T - 1)$, where n_1 is the number of measured displacements which have a time lag of 1 frame and is the total number of frames in the simulated movie. The percent recovered displacements decreases as M_{C} increases, and the number of recovered

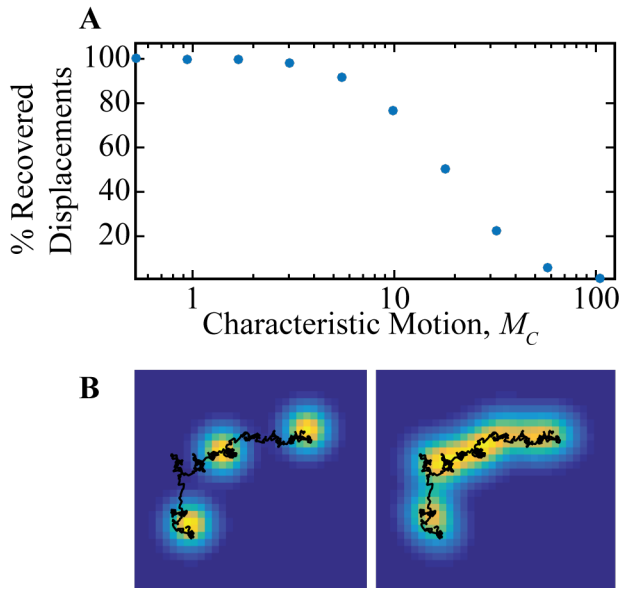

Figure 3.3.1: Motion blur degrades the single-particle tracking (SPT) signal as characteristic motion, M_{C}, increases. Diffusion confined in a cylinder was simulated and analyzed with SPT. M_{C} is defined according to Eq. 3.2.5. (A) Percent recovered data decreases with increasing M_{C}. Pixel width: 49 nm, confinement length, $L = 3$ μm, integration time, $t_{\mathrm{frame}} = 50$ ms, and PSF standard deviation, $\sigma_{\mathrm{PSF}} = 98$ nm. (B) Representative simulated images of a diffusing molecule with three of the subframes (left panel) or all of the subframes (right panel) included in the image. The actual trajectory (black curve) is the same in both panels.

displacements approaches zero for large M_{C} (Figure 3.3.1).

In general, M_{C} is proportional to D and tframe, and increasing either of these parameters increases the degree of in-frame motion of the diffusing molecule. Accordingly, Figure 3.3.1A indicates that increasing in-frame motion leads to decreased data recovery. Figure 3.3.1B compares simulations of the same trajectory (black line) with different amounts of in-frame motion. On the left side, the trajectory is sampled instantaneously at three distinct time points, producing three ideal point spread functions (punctate yellow spots); the molecular positions at those time points can be determined from a Gaussian fitting algorithm. On the other hand, the right side shows the case when the trajectory is sampled continuously. Such

in-frame motion can blur and distort the molecule image. Thus, when in-frame motion is dominant, this distortion prevents a Gaussian fitting algorithm from identifying molecular positions. Because SPT creates a trajectory from a series of single-molecule fits, such a failed fit decreases the percent recovered displacements. SPT, then, is a method that tolerates only subtle in-frame motion. In Figure 3.3.1A, fewer than 50% of the 1 frame displacements were recoverable when M_{C} is larger than 20, even though 20 is a reasonable value for subcellular diffusion in a typical single-molecule microscope [9]. For instance, $M_C = 20$ could correspond to a molecule diffusing with a rate of 2.88 $\mu\mathrm{m}^2\mathrm{s}^{-1}$ measured with a microscope that has a point spread function standard deviation of 98 nm, a camera integration time of 50 ms and pixel width of 49 nm in the object plane. Experimentally, in-frame motion can be minimized for even extremely fast diffusers with stroboscopic illumination (e.g., pulsed sample illumination with pulse widths that are shorter than tframe; left side of Figure 3.3.1B) [69] or very high imaging frame rates [79].

3.3.2 Effect of in-frame motion on spatio-temporal image correlation spectroscopy (STICS)

STICS has the potential to estimate the diffusion coefficient in the regime of large M_{C} (i.e., significant image blurring) that precludes SPT analysis. However, a description of the accuracy and precision of this method are required to validate its use in this regime. To investigate the accuracy of STICS in different diffusion regimes, we first simulated unconfined, single-molecule diffusion with ten diffusion coefficients between 0.1 $\mu\mathrm{m}^2\mathrm{s}^{-1}$ and 20 $\mu\mathrm{m}^2\mathrm{s}^{-1}$ (Figure 3.3.2). Each diffusion coefficient was considered for $t_{\mathrm{frame}} = 10$ ms and 50 ms and for point spread function standard deviations, $\sigma_{\mathrm{PSF}} = 49$ nm, 98 nm, and 147 nm. No noise was

included in this set of simulations. Per Eq. 3.2.4, different combinations of these parameters can lead to the same degree of in-frame motion, so each measured diffusion coefficient, D, was converted to the corresponding value of the unit-free parameter M_{C}. Simulated movies were analyzed by STICS, and the diffusion coefficient measurement bias, $(D_{measured}-D_0)/D_0$, is plotted as function of M_C in Figure 3.3.2A. According to Figure 3.3.2A, STICS very accurately estimates D at low M_C, but overestimates the diffusion coefficient as M_C increases. This positive measurement bias increases monotonically with increasing in-frame motion, with a bias of about 10% when $M_C = 100$. The measurement bias for each set of parameters (t_{frame}, σ_{PSF}) is assigned a unique color in Figure 3.3.2A (3.3.3). The different color sets overlap and show the same trend, demonstrating that the characteristic motion, M_{C} in Eq. 3.2.4, is indeed the right variable for considering diffusion at different rates under different experimental conditions.

We generated the images of a diffusing fluorescent molecule with in-frame motion analyzed in Figure 3.3.2A from a series of images of a stationary molecule that diffuses between subframes (right panel in Figure 3.3.1B). We hypothesized that the measurement bias in Figure 3.3.2A can be attributed to in-frame motion because this bias disappears when M_{C} is small. To examine the role of in-frame motion, we excluded this motion from the simulation by excluding all subframes but the first in each image (left panel in Figure 3.3.1B). We repeated STICS analysis on this stroboscopic data set, which is free of in-frame motion (Figure 3.3.2B), and found that the measurement bias is removed when no in-frame motion is allowed. Thus, the bias in Figure 3.3.2A is attributed to in-frame motion.

For the conventional STICS analysis in Figures 3.3.2A and 3.3.2B, the correlation function variance is estimated as the variance parameter of a fit to a Gaussian function [70]. While curve fitting to the correlation function is a powerful method of increasing the diffusion

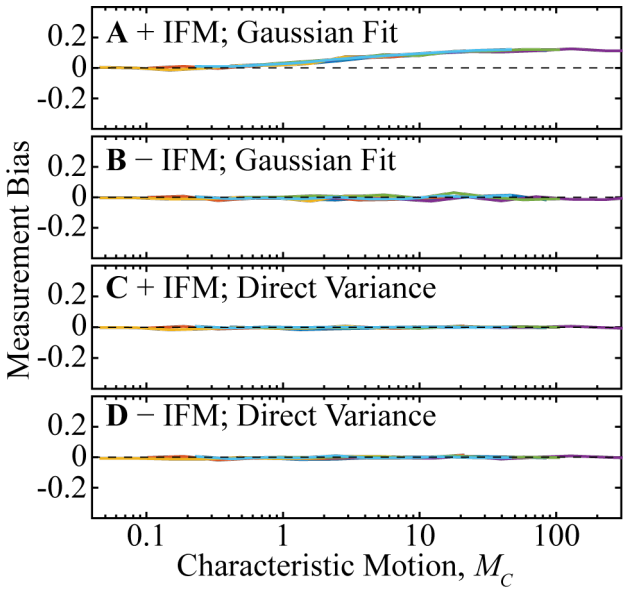

Figure 3.3.2: Diffusion coefficient measurement bias estimated from simulations of unconfined diffusion of a fluorescent molecule with no noise. Colors indicate the set of parameters (t_{frame}, σ_{PSF}) (3.3.3) and M_{C} is defined according to Eq. 3.2.4. (A) STICS analysis by Gaussian fitting of simulations that contain in-frame motion (IFM). Significant bias is measured for $M_C > 1$. (B) STICS analysis by Gaussian fitting of simulated movies that exclude in-frame motion. (C) and (D) The results of analyzing the same data in (A) and (B), but determining the step size distribution variance in the STICS protocol by directly calculating the variance rather than by fitting the correlations to Gaussians functions. Both the removal of in-frame motion in (B) and the direct calculation of variance in (C) and (D) eliminate the measurement bias.

coefficient measurement precision, as described in the Theory section, in the case of in-frame motion, the step sizes will not be normally distributed. Therefore, we hypothesized that the diffusion coefficient measurement bias in Figure 3.3.2A comes from fitting a non-Gaussian correlation function to a Gaussian fitting function. We therefore calculated the correlation function variances directly instead of by a fit to a Gaussian (Eqs. 3.2.10, 3.2.11). The measurement biases from this method are shown in Figures 3.3.2C and 3.3.2D for simulated data with and without in frame motion, respectively. This direct variance calculation allows for bias-free measurements of the diffusion coefficient even when in-frame motion is included

	τ (ms)	σ (nm)	L (μm)	Line Color
Figure 2	10	49	NA	
	10	98	NA	
	10	146	NA	
	50	49	NA	
	50	98	NA	
	50	146	NA	
Figures 3, 6, and S1	10	98	2	
	10	98	3	
	10	98	4	
	50	98	2	
	50	98	3	
	50	98	4	

Figure 3.3.3: Line colors for the simulated data in Figures 3.3.2, 3.3.4, 3.3.10 and 3.3.5.

in the simulated movies (compare Figure 3.3.2C to Figure 3.3.2A). The bias due to in-frame motion in Figure 3.3.2A is therefore the result of fitting STICS correlation functions to a fit function which incorrectly assumes an absence of in-frame motion, and the discrepancy in shape between the actual correlation function and this assumed Gaussian function increases with increasing in-frame motion.

3.3.3 Effect of confinement on STICS

In addition to the artifacts that arise from fitting fast diffusion to the wrong function in the STICS method (Figure 3.3.2A), we hypothesized that confinement also gives rise to a diffusion coefficient measurement bias in the case of fast diffusing molecules (high M_C). We considered the diffusion of a fluorescent molecule in the cytoplasm of a bacterial cell by simulating diffusion of a molecule in cylinders of diameter 1 μm and lengths $L = 2$, 3 and 4 μm with ten diffusion coefficients between 0.1 $\mu m^2 s^{-1}$ and 20 $\mu m^2 s^{-1}$. We considered $t_{\text{frame}} = 10$ and 50 ms, and σ_{PSF} was 98 nm. No noise was included in this set of simulations. In Figure 3.3.4, each set of parameters is scaled to the unit-free parameter M_C as described in Eq. 3.2.5.

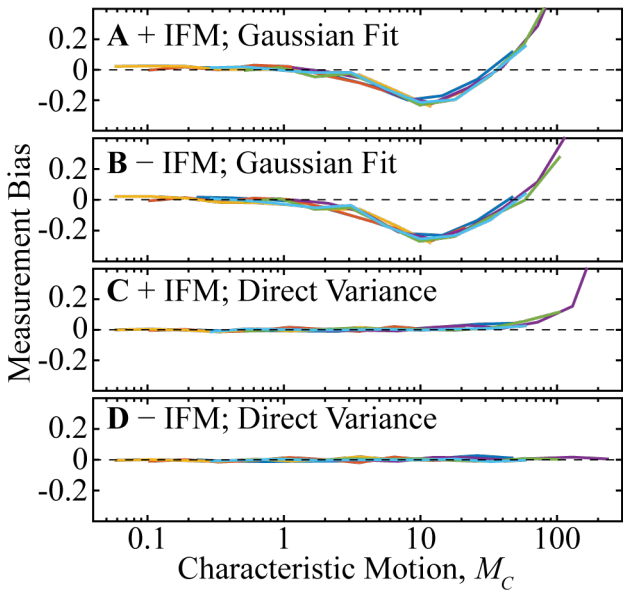

Figure 3.3.4: Diffusion coefficient measurement bias estimated from simulations without noise of a fluorescent molecule confined to a cylinder. Colors indicate the set of parameters (t_{frame}, σ_{PSF}) (3.3.3) and M_{C} is defined according to Eq. 3.2.5. (A) STICS analysis by Gaussian fitting of simulations that contain in-frame motion (IFM). Significant, non-monotonically varying bias occurs for fast characteristic motion, $M_C > 1$. (B) STICS analysis by Gaussian fitting of simulated movies that exclude in-frame motion. The non-monotonically varying bias from (A) is still present at all values of M_{C} considered, though the bias due to in-frame motion is removed. (C) and (D) The results of analyzing the same data in (A) and (B), but determining the step size distribution variance in the STICS protocol by directly calculating the variance rather than by fitting the correlations to Gaussians functions. The combination of direct variance calculation and removal of in-frame motion in (D) removes the bias completely.

Conventional STICS analysis of these simulations of confined motion (Figure 3.3.4A revealed a large, non-monotonically varying bias when $M_C > 1$, with biases as large as 50% for $M_C = 100$. This effect persists even when in-frame motion is eliminated from the simulations (Figure 3.3.4B) by excluding all subframes but the first in each image as described in Figure 3.3.2B. Comparing Figures 3.3.4A and 3.3.4B (see 3.3.3) indicates that in-frame motion does contribute slightly to the bias in Figure 3.3.4A, but the persistence of a significant bias in Figure 3.3.4B indicates that the effect of in-frame motion is minimal relative to the effect of confinement on the measurement bias.

We hypothesized that this confinement artifact arose from using an incorrect fit function to determine the correlation function variance. To circumvent this fit function, the correlation function variances were calculated directly as for Figures 3.3.2C and 3.3.2D. Again, the direct variance calculation allows a bias-free measurement of D: directly calculating the variance in this manner removed nearly all of the measurement bias for simulated data even with in-frame motion (Figure 3.3.4C). The fact that direct variance calculation can remove the large biases in estimation of D indicates that, as was true for in-frame motion (Figure 3.3.2A), confinement leads to a large discrepancy between the assumed normal step size distribution (Eq. 3.2.1) and the actual observed distribution. As confinement (and therefore M_C) increases, a Gaussian fitting function no longer correctly describes the STICS correlation function, and the resulting bias increases with increasing M_C (Figure 3.3.5B). There remains in Figure 3.3.4C a large positive bias at large M_C that is caused by in-frame motion and is removed upon excluding in-frame motion (Figure 3.3.4D).

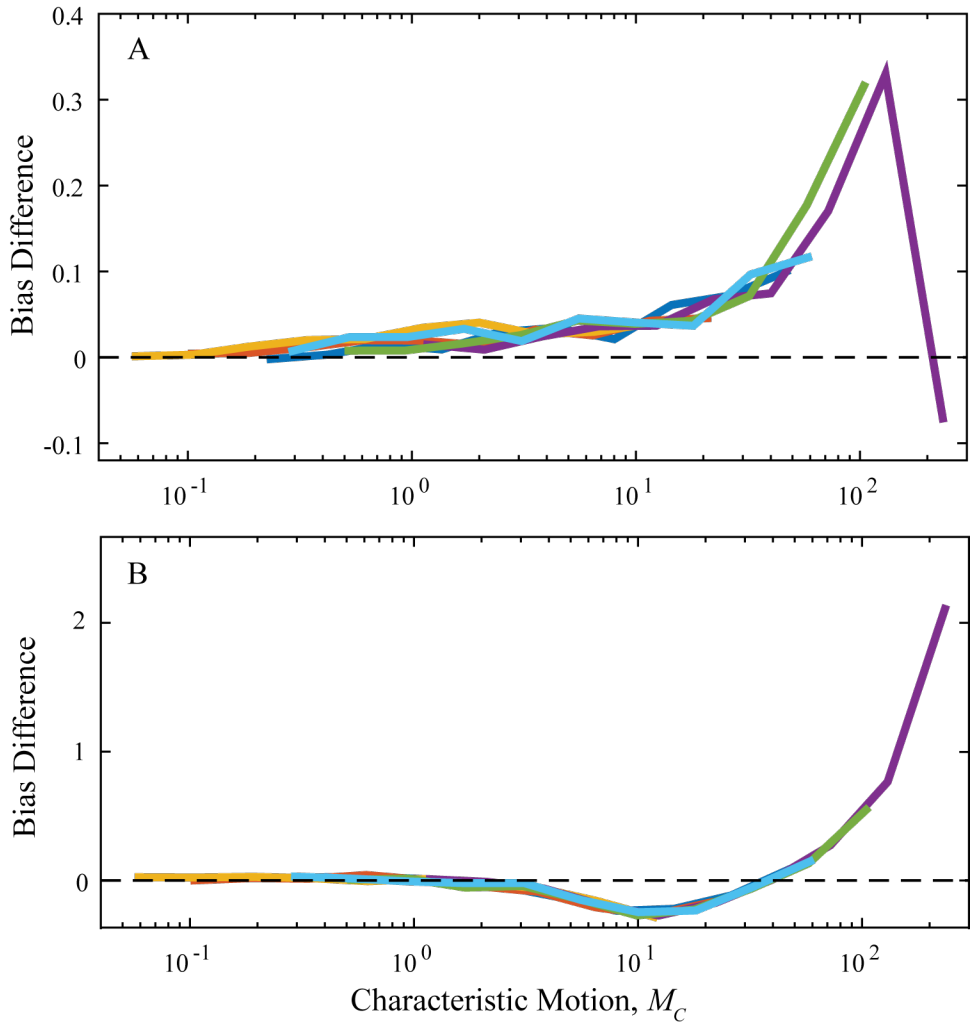

Figure 3.3.5: (A) Figures 3.3.4A and 3.3.4B showed the bias in determination by Gaussian fitting STICS of the diffusion coefficient, D, in the presence and absence of in-frame motion, respectively. Here, the difference between these two biases shows that removing in-frame motion does decrease the bias, especially at $M_C > 10$. (B) Figures 3.3.4A and 3.3.4C showed the bias in determination of the diffusion coefficient, D, by Gaussian fitting STICS and by STICS with direct computation of the variance, respectively. Here, the difference between these two biases highlights the large magnitude of the confinement artifact in Gaussian fitting STICS, which overwhelms the in-frame motion artifact in magnitude. Colors indicate the set of parameters (t_{frame}, σ_{PSF}) (3.3.3) and M_C is defined according to Eq. 3.2.5.

3.3.4 Comparing the precision of STICS and single-particle tracking

Given the protocol established above to remove measurement bias artifacts from STICS analysis through direct calculation of the variance, we next considered the measurement errors incurred in estimating single-molecule diffusion coefficients by either STICS or SPT in the presence of realistic noise and experimental parameters. We examined the ability of STICS to precisely determine D in simulated 1200-frame movies of a fluorescent molecule diffusing inside a cylinder of diameter 1 μm and $L = 3$ μm with SNR = 6, $\sigma_{\mathrm{PSF}} = 98$ nm, $t_{\mathrm{frame}} = 50$ ms, and ten diffusion coefficients between 0.1 $\mu\mathrm{m}^2\mathrm{s}^{-1}$ and 20 $\mu\mathrm{m}^2\mathrm{s}^{-1}$. In Figure 3.3.6, the percent error in the determination of the diffusion coefficient is plotted as a function of M_{C} (as defined in Eq. 3.2.5). STICS and SPT analysis are performed on two identical data sets: movies simulated with and without in-frame motion. For minimal in-frame motion and minimal confinement (small M_{C}), the precision of SPT is higher than that of STICS. Yet, when $M_C > 50$ (arrow in Figure 3.3.6), in-frame motion causes SPT (yellow curve) to be significantly less precise than STICS (blue curve). This is consistent with the data loss during tracking (Figure 3.3.1A). However, in the absence of in-frame motion, SPT outperforms STICS under all experimental conditions considered in this work. The error in determining D for both methods increases with increasing M_{C} because the effective SNR of each simulated image decreases with increasing M_{C} as is the case when the number of photons emitted by a molecule is kept constant but these are spread over more pixels due to image blur.

By directly considering the correlation function variance rather than assuming a Gaussian functional form for this distribution, the diffusion of even fast moving, highly confined

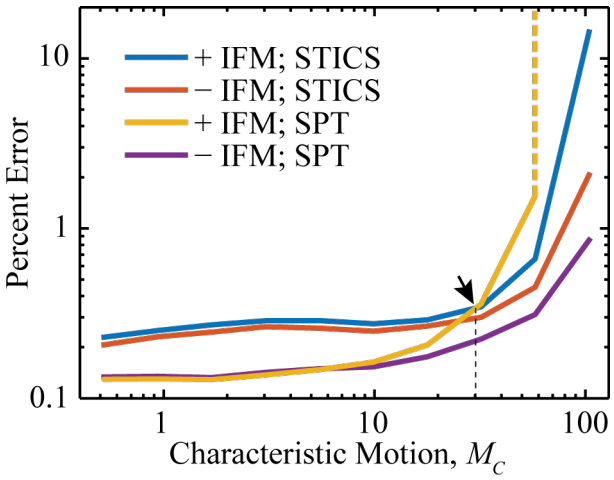

Figure 3.3.6: Simulated movies (see text for parameters) of confined diffusion with and without in-frame motion (IFM) were analyzed with the SPT and STICS methods to compare the precision of the two methods. The diffusion coefficient determination by STICS was done by direct calculation of the variance. SPT has higher precision at slow speeds, but becomes less precise for fast characteristic motion. The inclusion of in-frame motion causes the precision of SPT to become worse than that of STICS at $M_C \sim 30$ (arrow), which, for this parameter set, corresponds to a diffusion coefficient of 5.76 $\mu m^2 s^{-1}$. The tenth data point of the yellow curve (SPT with in-frame motion) was infinite due to data loss described in Figure 3.3.1A; this singularity is indicated by a dashed vertical line beginning at the ninth data point.

molecules can be estimated with STICS. Indeed, because of in-frame motion, when $M_C > 50$, one can no longer track diffusing molecules at all with SPT (yellow curve in Figure 3.3.6), whereas STICS can still determine the diffusion coefficient with < 1 % error. Overall, Figure 3.3.6 can indicate the appropriate analytical method for a given single-molecule diffusion measurement based on the experimental parameters.

3.3.5 Cytosolic mMaple3 diffusion in *E. coli*

Though single-molecule imaging has successfully answered a host of questions in cell biology [9, 87–89], one current challenge is a limited ability to characterize the dynamics of

the fastest molecules—especially when confinement precludes stroboscopic illumination-for instance freely diffusing proteins in the bacterial cytoplasm [85]. We imaged *E. coli* cells expressing the photoswitchable fluorescent protein mMaple3, which is switched from a green state to a red state upon activation with 406-nm light [90]. Before activation, the cells were non-fluorescent (Figure 3.3.7B). We exposed the cells to 406-nm laser pulses to photoswitch one mMaple3 molecule at a time (Figure 3.3.7C), and we sequentially imaged dozens of mMaple3 molecules in each of 87 different cells. The cells were imaged in the red channel (imaging wavelength: 561 nm) with $t_{\text{frame}} = 40$ ms, $\sigma_{\text{PSF}} = 98$ nm, and pixel width = 49 nm (Figure 3.3.7). We measured the cell lengths, which varied from 1 - 10 μm. Single-molecule photoswitching was evident as the average fluorescence intensity of the cell increased after each photoswitching pulse and decreased a short time later due to photobleaching, but the large degree of in-frame motion due to the fast diffusion prevented single molecules from being visualized as punctate fluorescent spots and localized with Gaussian fitting in the majority of the images. This is consistent with the low data recovery rate for simulations of molecules with high M_{C} in Figure 3.3.1A. The diffusion of free mMaple3 under these conditions could therefore not be analyzed by SPT. Though the method of direct variance determination removes the systematic error (compare Figures 3.3.2C and 3.3.4C to Figures 3.3.2A and 3.3.4A) and explains the source of the bias, this method has extremely low precision (light blue and green curves in Figure 3.3.8). This approach is therefore not used to analyze the experimental results that follow.

Figure 3.3.9 illustrates our implementation of Gaussian-fitting STICS for one representative cell ($L = 3.1$ μm). A phase contrast image of the cell (Figure 3.3.9A) provided the boundaries for a cell mask (Figure 3.3.9B), and a single $iMSD$ curve was compiled for each cells axial dimension based on Gaussian fits to the correlation function within this mask

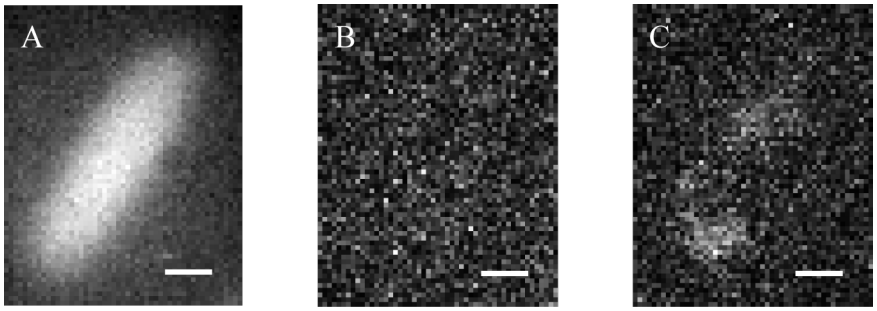

Figure 3.3.7: (A) Time-averaged fluorescence image obtained from summing a time series of fluorescence images of mMaple3 diffusion inside an *E. coli* cell. (B) A single image ($t_{\text{frame}} = 40$ ms) from that time series shows a cell after the mMaple3 molecule has bleached. (C) An image of the same *E. coli* cell in (B) two frames earlier, showing a typical mMaple3 molecule diffusing so rapidly that it is diffuse over nearly the entire bacterium. Scale bars: 0.5 μm.

(Figure 3.3.9C). Each $iMSD$ was fit to the square-confined diffusion model in Eq. 3.3.9 to estimate the coefficient of diffusion of mMaple3 in that cell (Figure 3.3.9D). Gaussian fitting was used to calculate the $iMSD$ with increased precision in our experimental regime of relatively noisy data. Also, because transverse and longitudinal diffusion must be considered independently, here due to the different confinement lengths in the two directions, the correlation function variances cannot be calculated directly without interpolating the raw data onto a rotated pixel array which would impose additional errors.

We analyzed 87 unique bacterial cells, and used the small molecule cephalexin to generate cells with $L = 1 - 10$ μm. The effect of cephalexin on the diffusion of mMaple3 in the cytoplasm of *E. coli* is expected to be minimal [64]. The simulations described in Figure 3.3.4A indicate the biases that are expected, and the expected bias from Figure 3.3.4A can be used to convert biased experimental diffusion measurement made with Gaussian-fitting STICS to the corresponding unbiased value (Figure 3.3.10A). For instance, in the example of the representative cell in Figure 3.3.9, Gaussian-fitting STICS measured $D = 5.20$ $\mu\text{m}^2\text{s}^{-1}$

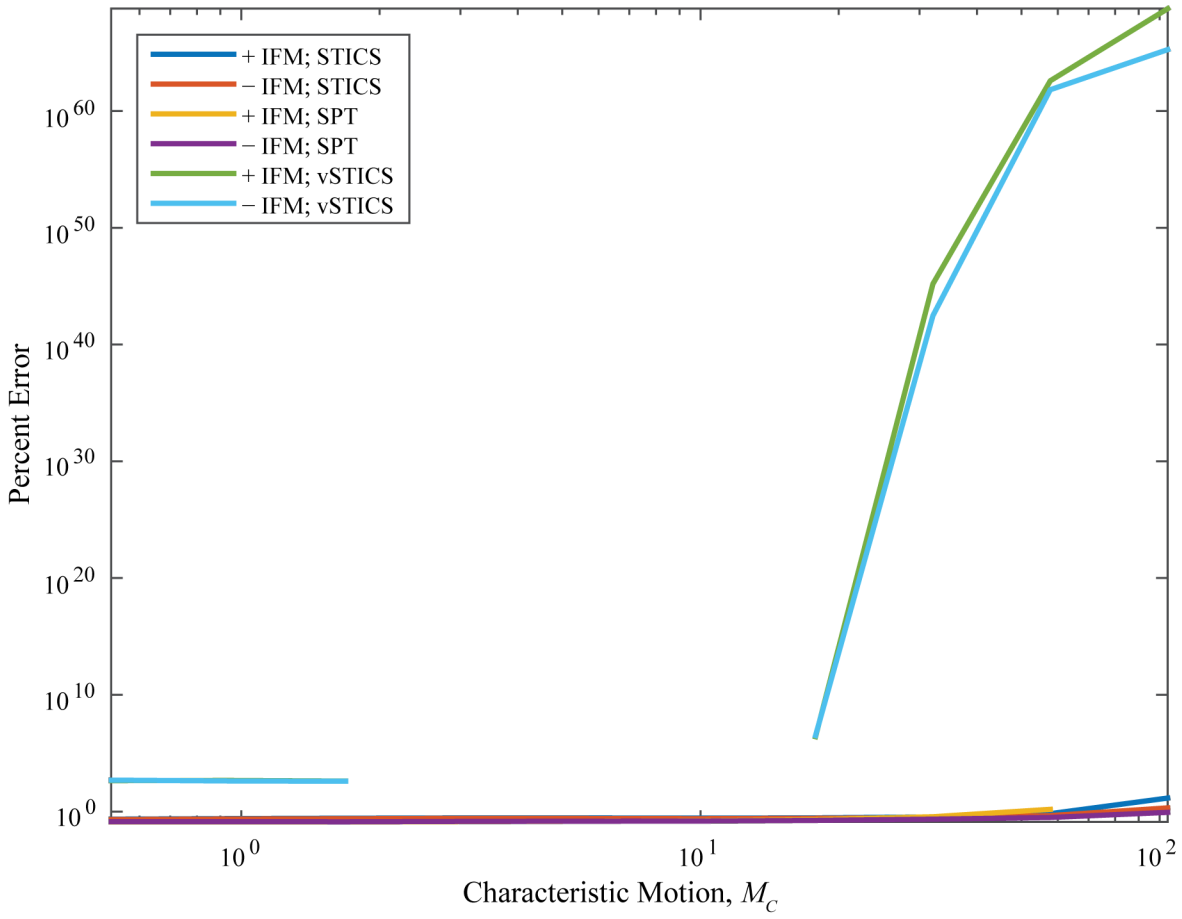

Figure 3.3.8: Measurement variance of STICS with the two methods of variance computation. Dark blue, red, yellow, and purple curves as in Figure 3.3.6. The green and light blue lines give the percent error when STICS is used to analyze data with and without in-frame motion (IFM), when the Gaussian-fitting step is replaced by direct variance computation. The missing points in the light blue and green curves (direct variance method, vSTICS) represent a complete failure to estimate the value, and even when the analysis algorithm had sufficient data to succeed the percent error was never below 1000%. The M_C is defined according to Eq. 3.2.5.

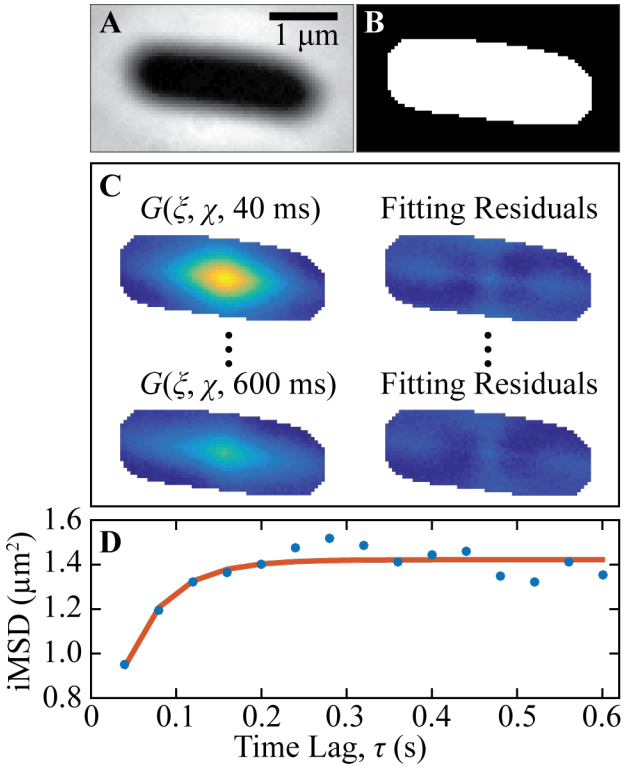

Figure 3.3.9: Analysis procedure for estimating by STICS the diffusion coefficient of free cytosolic mMaple3 inside a single *E. coli* cell. (A) A phase contrast image of the *E. coli* cell provides (B) a mask of the cell location and orientation. (C) The STICS correlation function is computed and then fit to a Gaussian function, G. The phase mask in (B) includes only the correlation amplitudes that correspond to displacements inside the cell. (D) The long-axis variances of the Gaussian fits to the correlation functions ($iMSD$) were plotted as a function of time lag, τ (dots), and this $iMSD$ curve was fit to a model for square-confined diffusion (Eq. ; red curve) to obtain a single diffusion coefficient measurement for each cell.

in a cell with $L = 2.41$ μm. This measurement was taken with $\sigma_{\mathrm{PSF}} = 98$ nm, $t_{\mathrm{frame}} = 40$ ms, and pixel width $= 49$ nm, so Eq. 3.2.5 gives $M_C = 33.6$, for which value Figure 3.3.4A predicts a bias of -0.01 (arrow in Figure 3.3.10A). Interestingly, for the representative cell in Figure 3.3.9, ($M_C = 33.6$), the measured D (5.20 $\mu\mathrm{m}^2\mathrm{s}^{-1}$) is converted to an unbiased D of 5.15 $\mu\mathrm{m}^2\mathrm{s}^{-1}$; i.e., Gaussian-fitting STICS produces a nearly unbiased estimate of D for the amount of in-frame motion and confinement in this regime.

Based on the interpolated curve in Figure 3.3.10A, all D measurements were converted to unbiased D, and Figure 3.3.10B shows a histogram of the average diffusion coefficient estimates for single mMaple3 molecules in each of the 87 different *E. coli* cells. The unbiased mean diffusion coefficient of 9.6 ± 1.0 $\mu\mathrm{m}^2\mathrm{s}^{-1}$ agrees with the range of free protein diffusion inside the cytoplasm of *E. coli* reported elsewhere [66, 80], indicating that Gaussian-fitting STICS is an appropriate analysis method for obtaining the average D of a cytoplasmic fluorescent protein in *E. coli*. We found no correlation between the measured D and L, indicating that the degree of confinement did not strongly affect our measurements (Figure 3.3.11). Overall, these experiments demonstrate that the diffusion of fluorescent molecules in small volumes with a significant amount of in-frame motion that precludes SPT can be characterized by the STICS method.

3.4 Conclusion

In this chapter, we have extended STICS to the regime of rapidly moving molecules in highly confined environments by considering the motion of freely diffusing fluorescent proteins inside living *E. coli* bacterial cells. In this regime, such fast, confined motion can still be characterized with negligible bias, though we have also identified regimes where fast, con-

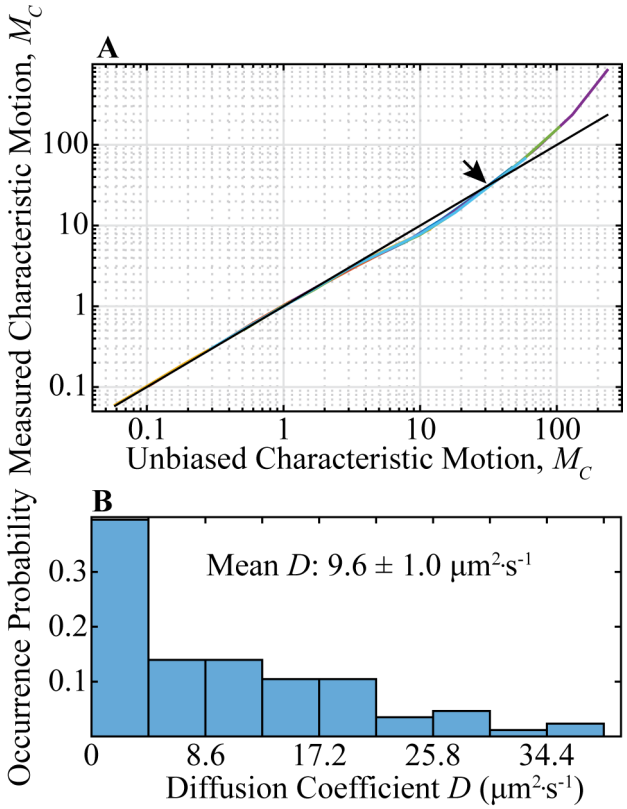

Figure 3.3.10: Analysis of the diffusion coefficients of mMaple3 in 87 *E. coli* cells measured with STICS. (A) Measured M_C may be converted to unbiased M_C based on the simulations in Figure 3.3.4A, which are used here to create a look-up table. The overlapping colored lines represent the same experimental parameter sets as in Figure 3.3.4A (Figure 3.3.3). The black line indicates 1:1 correspondence (no bias). (B) Histogram of the unbiased average mMaple3 diffusion coefficient measured in each of 87 different *E. coli* cells as estimated by the STICS method. 5 - 15 mMaple3 fluorescent proteins are photoswitched one-at-a-time in each cell, and the biased measurements were corrected by cubic interpolation of the curve in panel (A).

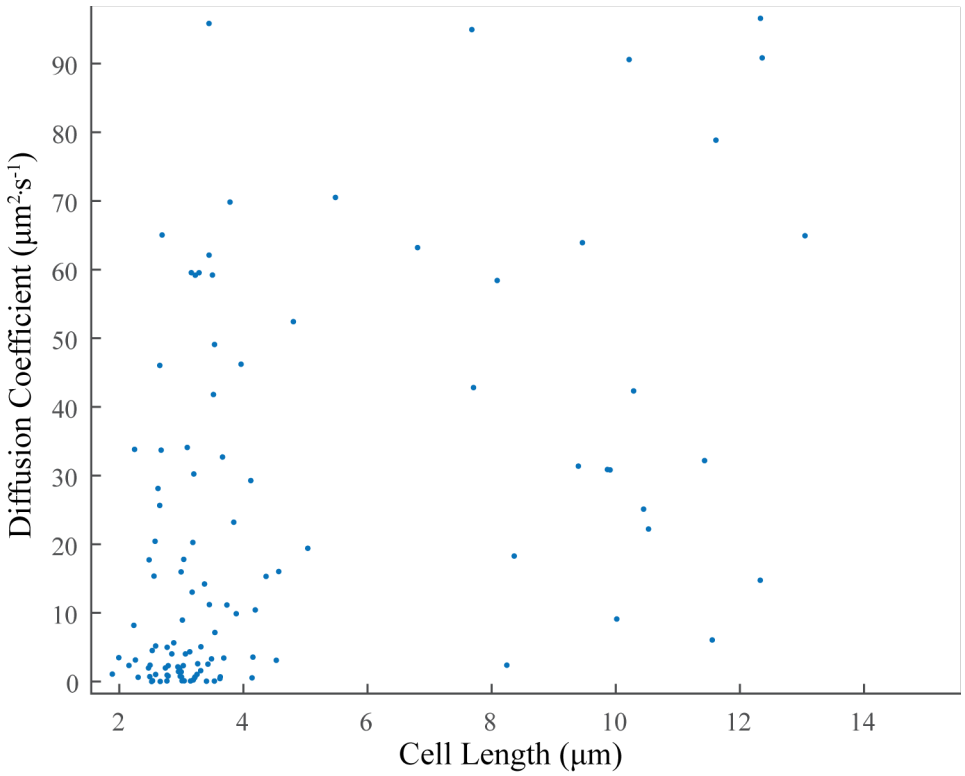

Figure 3.3.11: Diffusion coefficients vs *E. coli* cell length. The measured diffusion coefficient for mMaple3 molecules in each *E. coli* cell did not strongly depend on the cell length calculated from the phase contrast image of that cell.

fined motion results in a significant bias in diffusion coefficient measurement by STICS. Simulations of this bias indicate that it stems from a STICS correlation function that is not well approximated by a Gaussian function (Figures 3.3.2A and 3.3.2A); this bias increases with increasing diffusion coefficient, D, and increasing confinement, L^{-2}, as described here by the unit-free parameter characteristic motion (M_{C}). Thus, when M_{C} becomes large due to in-frame motion or confinement, Gaussian-fitting STICS is a biased method for estimating the diffusion coefficient. Interestingly, this bias can be removed by directly calculating the variances of the correlation function rather than estimating the variance by Gaussian fitting (Figures 3.3.2C and 3.3.4C), and we provide an analytical description of this process. Furthermore, because this unbiased direct calculation of the correlation function variances is less precise than Gaussian fitting and therefore not desirable for the treatment of noisy experimental data, the simulations in this chapter provide a way to remove the bias when Gaussian-fitting STICS is used. In particular, the plot of measured M_{C} vs. unbiased M_{C} in Figure 3.3.10A, together with the definition of M_{C} in Eq. 3.2.5B, provides a look-up table for all experimental parameter sets in future experiments where in-frame motion and confinement causes Gaussian-fitting STICS to measure a biased value of M_{C}. In this way, some of the precision of Gaussian-fitting STICS is preserved while the measurement bias is eliminated. Still, in the future, it would be desirable to solve Eqs. 3.2.8 and 3.2.9 and replace the Gaussian fit function entirely with a more accurate description of a correlation function computed from a movie of a highly confined diffusing fluorescent molecule that exhibits in-frame motion; this is beyond the scope of the current work.

Overall, Figure 3.3.6 illustrates one of the primary strengths of STICS: when $M_C > 50$, in-frame motion dominates and SPT fails entirely to estimate the diffusion coefficient because of data loss as shown in Figure 3.3.1A. Thus, though the experimental data in Figures 3.3.9 and

3.3.10 were not analyzable by SPT, diffusion coefficients were still recovered by STICS. Based on this chapter, independent researchers in single-molecule fields should be able to determine the value of characteristic motion in their experiments and decide which analysis method (STICS or SPT) produces the most precise or least biased estimate of the diffusion coefficient for the relevant experimental regime. This decision can be made based on performance or pragmatism: for instance, the timing and optics of stroboscopic illumination can be difficult to implement, especially in commercial single-molecule instruments, and current EMCCD camera technology requires a very small imaging area for the highest accessible frame rates. The relative precision of STICS compared to SPT at large M_C makes STICS a promising and facile method for the estimation of diffusion coefficients in experimental measurements of fast diffusion that is confined within small cells and organelles, within nanomaterials, and in any other highly confined system where the imaging camera frame rate is longer than the time that the molecule takes to explore the confinement volume.

CHAPTER IV

Global fitting of single-particle tracking data improves estimation precision and fitting robustness

The work presented in this chapter has been submitted to the journal Nano Letters:

Rowland, D. J. and Biteen, J. S. Global fitting of single-particle tracking data improves estimation precision and fitting robustness. *Submitted.*

4.1 Introduction

Subcellular dynamics vary in time and over small size scales due to spatial and temporal variations such as transient interactions with molecular partners, [8] crowding by the nucleoid in bacteria, [79] or the presence of different lipid domains in membranes. [91] To measure the motions of biological molecules such as proteins and lipids in the face of this complicated local environment, single-molecule super-resolution fluorescence microscopy data must be analyzed with a method that systematically accounts for heterogeneity. One approach to single-particle tracking (SPT) derives the apparent diffusion coefficient from each individual

trajectory from the mean square displacement (MSD) vs. time lag curve. [65,92,93] With this MSD approach, heterogeneous diffusion is described by dividing the collection of MSD curves into diffusive populations; the number of trajectories in each population of curves may be taken as an estimate of the relative proportions of these diffusive populations. [36,40] However, heterogeneous diffusion can be observed even over the course of the trajectory of a single molecule, and this single-track MSD analysis specifically disallows the case where a single molecular trajectory experiences multiple diffusive modes by providing only the average diffusion coefficient for each track.

An approach that accounts explicitly for such heterogeneous motion considers the entire collection of single-molecule steps instead of dividing these steps into individual tracks. This collection of step data can then be quantified based on the cumulative probability distribution (CPD) of the total collection of squared step sizes to explicitly account for heterogeneous motion and increase the signal-to-noise ratio. Single-step analysis with CPD is therefore a diffusion estimation technique that has had impact in numerous disparate fields such as artificial membranes, leukocytes, bacterial membranes, neurons and artificial materials. [37,41,94–104]

Alternatively, a number of Bayesian [87,105–107] and machine learning [108] algorithms can be used to estimate the number of diffusive components and measure their properties, but the complexity of these methods poses a significant barrier to intuitive understanding of the underlying modes of heterogeneous motion. Fluorescence correlation spectroscopy (FCS) and the related methods of spatiotemporal image correlation spectroscopy (STICS), raster image correlation spectroscopy (RICS), or particle image correlation spectroscopy (PICS) can also be used here; these approaches all employ spatial or temporal correlation functions which can also be fit to multi-component diffusion models, [30,32,70,75,76] but it is rare for the signal-

to-noise to be high enough for the analysis of complex heterogeneous motion in bacterial systems. [34] The ability of single-particle tracking to isolate high quality trajectories from noisy single molecule data can present a more attractive conduit for analysis.

Despite its advantages, the commonly employed CPD analysis method requires a two-step fitting process where the CPD and MSD curves are fit sequentially. [94,96] We present here a method that combines this two-step fitting into a single-step multi-domain global fit algorithm. We analyze simulated trajectories of multiple diffusive components to measure the improvements in the diffusion coefficient estimation error and find that global fitting is superior to the traditional local fitting CPD analysis algorithm. We then consider the diffusion of 80 nm and 200 nm gold spheres in glycerol water solution to show that global fitting outperforms local fitting in a real system. We report improvements in precision, robustness, and simplicity of use.

4.2 Theory and Experimental

4.2.1 Imaging and tracking

Slides were imaged at room temperature using wide-field epifluorescence microscopy in an Olympus IX71 inverted microscope with a 100×, 1.40 NA oil immersion objective (in Zeiss Immersol 518F immersion oil) and appropriate excitation, emission, and dichroic filters (Semrock LL01-488, Semrock BLP01-488 and Semrock Di01-R488, respectively). After a 3× beam expander, a Photometrics Evolve EMCCD camera with $> 90\%$ quantum efficiency captured the images at 100 frames per second. Each camera pixel corresponds to a 49 nm × 49 nm area of the sample. The gold spheres were illuminated with a 488 nm laser (Coherent

Sapphire 488-50), which was circularly polarized with a quarter waveplate (Tower Optical AO15Z 1/4). Single molecule positions were associated into tracks with the Hungarian algorithm [109] according to an exponential merit function. [35]

4.2.2 Diffusion of gold spheres in glycerol

Gold nanoparticles with diameters 80 and 200 nm (BBI Solutions) were dispersed in 50% glycerol. 5 L of the mixture was sandwiched between two glass coverslips. The second through fifth time lags were used for both the global and local fitting algorithms to reduce the magnitude of the fitting residuals. Unweighted least squares fitting was performed with the Matlab built-in function *lsqnonlin*.

4.2.3 Simulations

Diffusion was simulated by generating 10^3 steps from a zero-mean normal distribution with variance equal to $2Dt_{\text{frame}}$, where D is the desired diffusion coefficient and t_{frame} is the simulated camera exposure time which was set to 0.04 seconds. Localization precision was simulated by adding zero-mean Gaussian-distributed random numbers to the simulated trajectories; the localization precision, or the standard deviation of the random numbers, was varied from 4.9 nm to 73.5 nm. Each simulation was repeated 10^4 times. The first 10 time lags were used for both the global and local fitting algorithms and unweighted least squares was performed with the Matlab built-in function *lsqnonlin*.

4.2.4 Bootstrapping

For the analysis of the tracks of gold spheres, histograms of estimated diffusion coefficients and population weights were produced by bootstrapping the fitting procedure. The total set of 13232 squared step sizes was sampled with replacement 300 times to produce 300 unique data sets each with as many values as the original data set. These bootstrapped data sets were then fit with either the global or local fitting method.

4.2.5 Local CPD fitting

To probe heterogeneous diffusion, the cumulative probability distribution (CPD) of squared step sizes (Δr^2) was calculated from the tracks of diffusing molecules at each time lag (τ) between frames in the trajectory. There is one CPD curve, CPD_i, for each time lag considered, and each CPD_i was fit to the multi-term exponential fit [94] with the appropriate number of terms (three terms shown here for instance):

$$CPD_i = 1 - \alpha_1 \times \exp\left(\frac{-\Delta r_i^2}{MSD_{1,i}}\right) - \alpha_2 \times \exp\left(\frac{-\Delta r_i^2}{MSD_{2,i}}\right) - (1 - \alpha_1 - \alpha_2) \times \exp\left(\frac{-\Delta r_i^2}{MSD_{3,i}}\right) \tag{4.2.1}$$

This series of fits, where i, runs from 1 to the number of time lags considered, N_τ, estimates three mean squared displacements, MSD_1, MSD_2, and MSD_3, as a function of time lag for $N_D = 3$ diffusive populations with weights α_1, α_2, and $(1-\alpha_1-\alpha_2)$, respectively. Each of the three MSD curves is then fit to a model—here of 2D unconfined diffusion—to extract the diffusion coefficient of the respective population of molecules. For example, for

population α_1, the second-step fitting function is:

$$MSD_1 = 4D_1\tau + 4\sigma_1^2, \tag{4.2.2}$$

where D_1 is the diffusion coefficient of population α_1, τ is the domain of time lags and σ_1 is the localization precision for population α_1. If one uses the first 5 time lags to estimate the diffusion coefficients of three populations, the total number of fitting parameters in this local CPD fitting approach is: $5 \times N_\tau + 2 \times N_D = 31$ here with $N_\tau = 5$, and $N_D = 3$.

4.2.6 Global CPD fitting

Instead of fitting in separate steps, the set of empirical CPDs may be fit all at once by incorporating the MSD functions (Equation 4.2.2 into Equation 4.2.1). Conceptually, this can be understood as the sharing of redundant parameters, such as the weight of population 1, α_1. The free parameters in the combined fitting function now include only N_D diffusion coefficients (one for each population), a single localization precision, σ, shared among all populations, and all but one of the population weights because one is estimated using the others. For instance, if one wishes to estimate the diffusion coefficients of three populations, the total number of fitting parameters is 6 (three diffusion coefficients, one localization precision parameter, and two population weight parameters). This number of fitting parameters in the global fitting algorithm is hugely improved from the 31 parameters necessary for local fitting.

We implemented Global Fit with a Matlab-specific formulation that exchanges several nonlinear least squares problems for a single larger nonlinear least squares problem. See Appendix B for complete code.

4.3 Results

To illustrate the power of the method, we implemented the CPD global fit algorithm in Matlab (Methods) and applied this analysis to simulated diffusive 2D trajectories (Figure 4.3.1A) with a combination of diffusion coefficients ($D_1 = 1\ \mu\text{m}^2\text{s}^{-1}$ and $D_1 = 0.01\ \mu\text{m}^2\text{s}^{-1}$). For each trajectory, the collection of squared displacements given a certain time lag are used to compute the empirical cumulative probability distribution for that time lag. Figure 4.3.1B compares the global fitting method to the traditional local fitting method. In both methods, the CPD of squared step sizes is computed at each of the first 10 time lags (black and grey curves). The local fitting method (green arrows) first fits each of the CPD curves independently to estimate the mean squared displacements (MSDs) of the two diffusive populations for each curve as well as to provide an average estimate of the population weights, α_1 and $(1 - \alpha_1)$. The two resultant MSD vs. time lag curves (blue and orange) are then fit in a second step to estimate the diffusion coefficients of each of the two populations. Global fitting, on the other hand, combines all of these fitting operations into a single least squares minimization step (blue arrow). This global fit process greatly reduces the total number of fitting parameters. In Figure 1B, the local fitting procedure requires 3 free parameters ($MSD_{1,i}$, $MSD_{2,i}$ and $\alpha_{1,i}$) for each CPD curve fit and 2 for each MSD curve (D and the localization precision, σ) for a total of 34 parameters. The global fitting method, which does not separate the CPD and MSD fits, in this case requires only 4 fit parameters (D_1, D_2, α_1, and σ).

Based on the analysis of simulations of unconfined 2D diffusion of two diffusive populations as in Figure 4.3.1A, Figure 4.3.2 describes the expected behavior of the global and local fitting methods. At each condition, 10^4 trajectories of 10^3 steps were randomly gener-

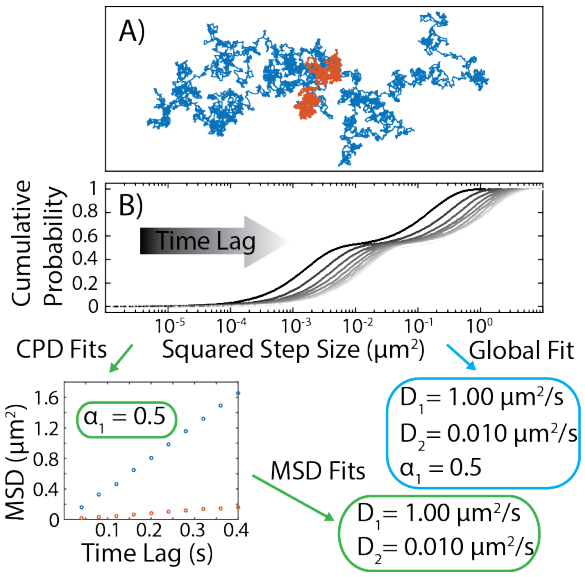

Figure 4.3.1: Schematic diagram of Local and Global fitting algorithms. (A) An example pair of diffusive 2D trajectories with 10^3 steps each and diffusion coefficients of 1 $\mu m^2 s^{-1}$ and 0.01 $\mu m^2 s^{-1}$ (blue and orange trajectories, respectively). The cumulative probability distribution (CPD) approach analyzes data from trajectories like these to extract the diffusion coefficients, D_1 and D_2, and relative weights, α_1 and $\alpha_2 = (1 - \alpha_1)$, of each diffusive population. (B) The CPD of squared step sizes is computed at each of the first 10 time lags. The resulting curves are plotted here with decreasing intensity from the first time lag (black) to the tenth (light gray). For local fitting (green arrows), first each CPD is fit to a two-population model yielding two series of mean squared displacement (MSD) values and a population weighting factor, α_1, that is the average from all the fits. The color of the each resultant MSD vs. time lag curve (bottom left) corresponds to the color of the corresponding trajectory in (A). Next, the two MSD curves are fit separately to yield D_1 and D_2. Alternatively, for global fitting (blue arrow), all 10 CPD curves are fit at once to yield α_1, D_1 and D_2. Local fitting here has a total of 34 fitting parameters whereas global fitting has only 4 (two diffusion coefficients, a population weight, and a localization precision).

ated with localization precision, σ (Methods). Figure 4.3.2A shows the standard deviation (S.D.) of the diffusion coefficient estimates for trajectories simulated with σ that varies from 4.9 nm to 73.5 nm. At all values of σ, global fitting (solid lines) outperforms local fitting (dotted lines) for both diffusive components (blue and orange). Notably, once σ approaches 70 nm (black arrow in Figure 4.3.2A), local fitting completely fails to estimate the diffusion coefficient of the faster diffusive population. This breakdown is caused when numerical instability due to poor data quality produces mostly meaningless diffusion coefficient estimates. The increased numerical stability of the global fit algorithm enables a precise and unbiased estimation of the diffusion coefficients even in the presence of 4 diffusive population; the local fitting algorithm fails completely to measure motion in this situation (Figure 4.3.3). Overall, by constraining all CPD curves to the same diffusion model parameters, the global fitting algorithm is more robust to increased complexity and decreased data quality. The histograms in Figures 4.3.2B and 4.3.2C present the results of the simulations used in Figure 4.3.2A at $\sigma = 4.9$ nm and 38 nm. For this two-population simulation, at both low and high levels of localization precision, the diffusion coefficients of both populations are measured without bias by both fitting methods. However, the decreased standard deviations for diffusion coefficient estimations indicate that the results from global fitting (solid curves) are more rigidly centered on their mean value. In other words, sampling error is less likely to skew the result of the global fitting when a two-term CPD model was used.

The global fit algorithm also shows improved performance in applications to experimental measurements where the ground truth is not known. We imaged suspensions of spherical gold nanoparticles of diameter 80 nm or 200 nm dispersed in 50% glycerol (Methods) and tracked the particles. Figure 4.3.4A reports the bootstrapped results of global and local fitting of a one-population diffusion model to movies of samples containing only 200 nm

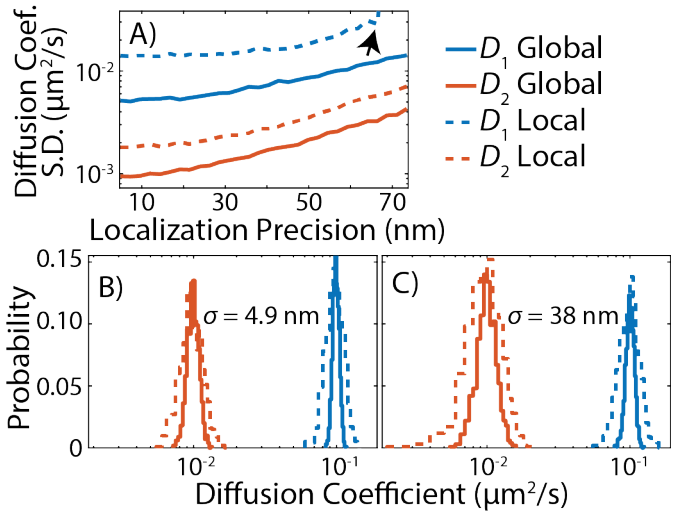

Figure 4.3.2: (A) Standard deviation of diffusion coefficient estimations for simulations of unconfined 2D diffusion of two diffusive populations. Trajectories were simulated as in Figure 4.3.1A with 10^3 time steps for each of two diffusion coefficients, 0.1 $\mu m^2 s^{-1}$ and 0.01 $\mu m^2 s^{-1}$, and white noise was added to simulate a range of localization precisions. The trajectories of both diffusive populations at each noise level were combined then analyzed with the global (blue) and local (orange) CPD fitting methods using a two-population model (Equation 4.2.1 with only the first two terms). 10^4 trajectories were randomly generated, each was analyzed; the standard deviations of the estimated diffusion coefficients, D_1 and D_2, are shown. The black arrow corresponds to the point at which numerical instability due to poor data quality causes the local fitting algorithm to fail to estimate the faster of the two diffusion coefficients. (B) and (C) Histograms of the estimated diffusion coefficients in (A) at localization precisions of 4.9 nm and 38 nm, respectively.

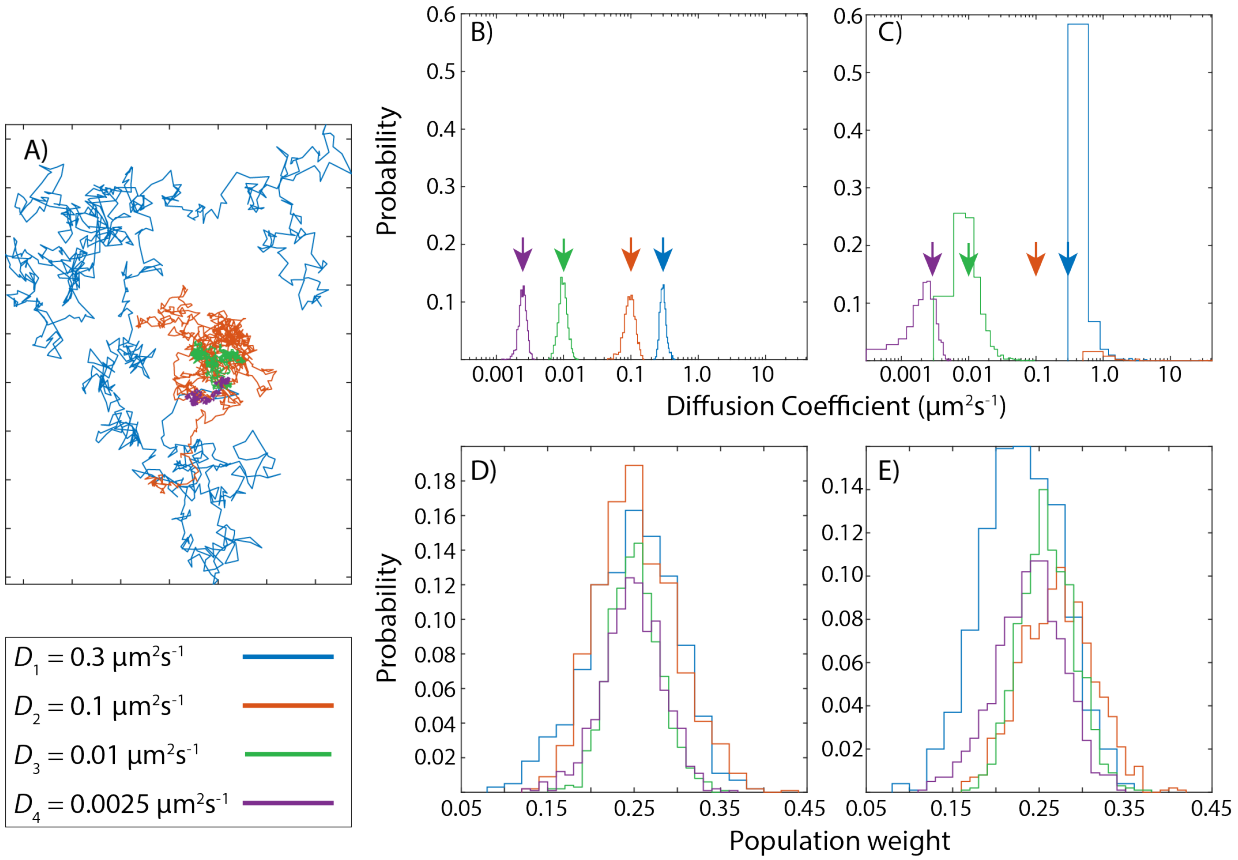

Figure 4.3.3: A) Sample simulated trajectories with four different diffusion coefficients as given in the legend at the bottom left. B) Histograms of diffusion coefficients estimated with the global CPD algorithm from 10^4 simulated trajectories, each consisting of 10^3 steps belonging to each diffusive population. C) Histograms of diffusion coefficients estimated with the local CPD algorithm for the same data used in B). D) and E) Estimated population weights for the diffusive populations using the global and local CPD algorithms, respectively. The nominal values here for the weights are $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$.

gold spheres. Global fitting results in a histogram of diffusion coefficient estimates with mean 0.24 $\mu m^2 s^{-1}$ and standard deviation 0.0061 $\mu m^2 s^{-1}$, whereas local fitting of the same data set estimates a larger diffusion coefficient of 0.27 $\mu m^2 s^{-1}$ and a smaller standard deviation of 0.0055 $\mu m^2 s^{-1}$. Figure 4.3.4B shows the estimated diffusion coefficients of 80 nm gold spheres in 50% glycerol. Again, the mean diffusion coefficient estimate increases and standard deviation decreases when the local fitting method is used instead of global fitting. Here, the diffusion coefficient increases from 0.74 $\mu m^2 s^{-1}$ to 0.82 $\mu m^2 s^{-1}$ and the standard deviation decreases from 0.026 $\mu m^2 s^{-1}$ to 0.022 $\mu m^2 s^{-1}$ when comparing the global and local fitting methods, respectively. In this case of single-component diffusion, we found that local fitting resulted in a more precise measurement of the diffusion coefficient. This result is likely due to the increased number of fitting parameters in local fitting causing the fitting residuals to be smaller. This increased precision should not be taken to imply an increase in accuracy, however, because there are no guarantees that the minima found in least squares minimization is the global minimum.

Single-particle tracks within mixtures of both nanoparticle sizes require a two-population diffusion model. Here, global fitting more precisely estimates the diffusion coefficients. Figure 4.3.4C shows histograms of the diffusion coefficient estimates for this scenario, where global fitting estimates the two diffusion coefficients to be 0.33 $\mu m^2 s^{-1} \pm 0.0074$ $\mu m^2 s^{-1}$ and 0.76 $\mu m^2 s^{-1} \pm 0.047$ $\mu m^2 s^{-1}$ corresponding to the 80 nm and 200 nm gold spheres, respectively. Local fitting, on the other hand, estimates the diffusion coefficients to be 0.37 $\mu m^2 s^{-1} \pm 0.021$ $\mu m^2 s^{-1}$ and 0.98 $\mu m^2 s^{-1} \pm 0.15$ $\mu m^2 s^{-1}$. In this scenario where multiple diffusive populations are present, the improved parameter rigidity of global fitting is readily apparent. As the complexity of the model increases, global fitting offers a more robust method for estimating diffusion coefficients. It should be noted, however, that both global

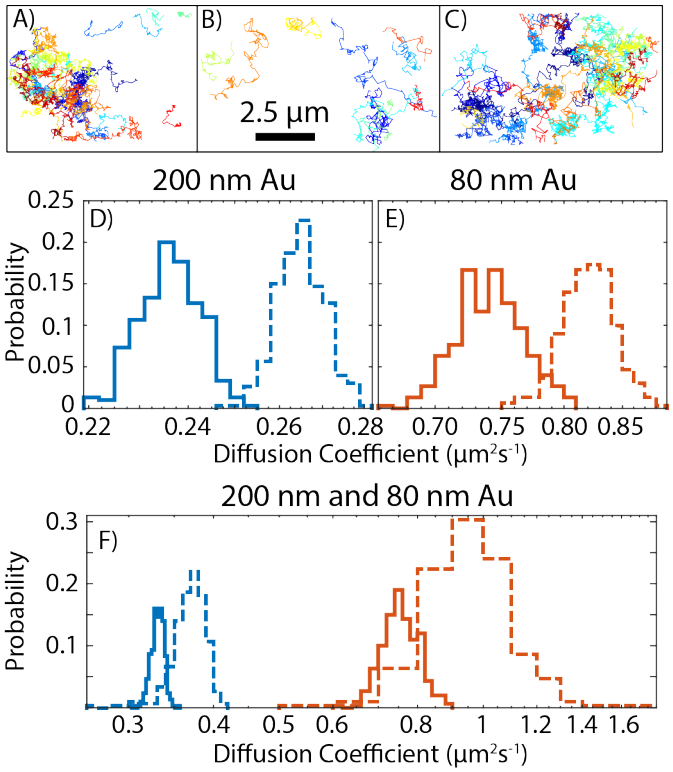

Figure 4.3.4: (A), (B), (C) 40 tracks of 200 nm gold spheres, 13 tracks of 80 nm gold spheres, and 28 tracks of a mixture of both sizes of gold spheres, all diffusing in 50% glycerol, respectively. (D) Histograms of the estimated diffusion coefficient for 200 nm gold spheres in 50% glycerol in water solution from fits to a one-component model with the Global Fit (solid line) and Local Fit (dashed line) methods. (E) Histograms of the estimated diffusion coefficient for 80 nm gold spheres under the same conditions. (F) Estimated diffusion coefficients of a mixture of 200 nm and 80 nm gold spheres in a 50% glycerol in water solution fit using a two-component model by Global Fit (solid lines) and Local Fit (dashed lines). The blue and orange line colors correspond to the sphere sizes in panels D and E, respectively.

and local algorithms performed equally well in the estimation of the population weights, α_1 and $(1 - \alpha_1)$ (Figure 4.3.5).

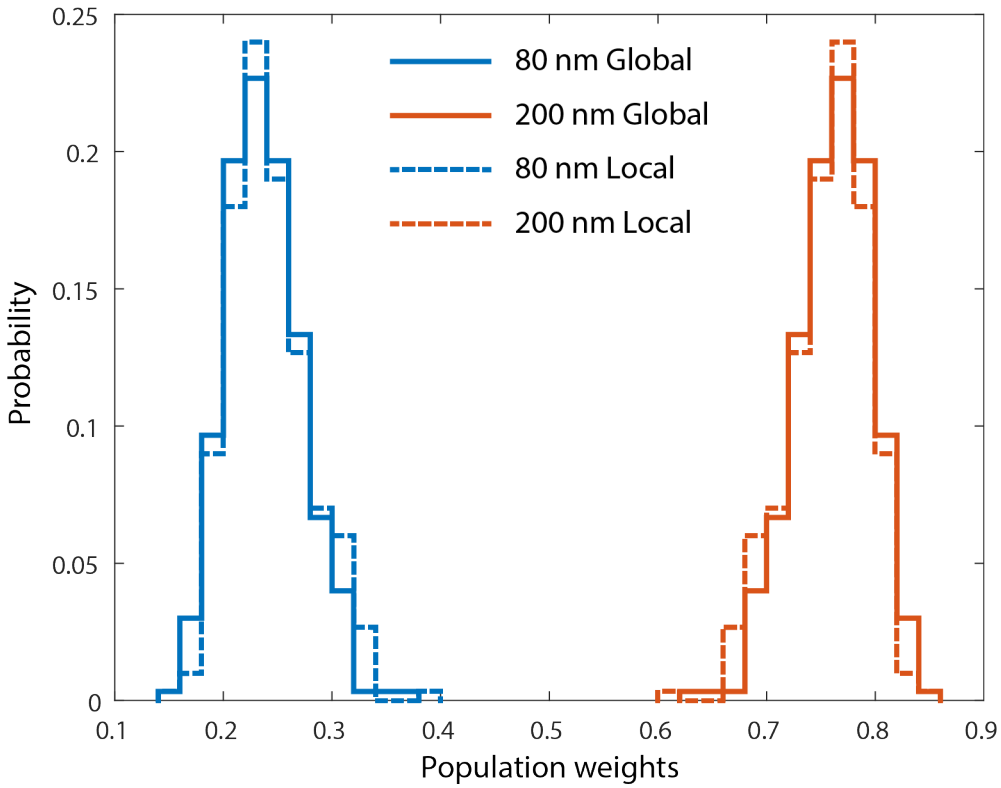

Figure 4.3.5: Population weights for the global and local CPD algorithm for the mixture of diffusing 80 nm and 200 nm gold spheres. Although the two algorithms estimate different values and precisions for the mean diffusion coefficients (see Figure 4.3.4, they both estimate the ratio of steps taken by the 80 nm spheres to the steps taken by the 200 nm spheres to be $1 : 3$. Both algorithms similarly estimate the population proportions to equivalent precision.

Alternatively, we investigated the diffusion of single DiI fluorescent molecules in supported lipid bilayers (Figure 4.3.6; see caption for methods), and uncovered sample inhomogeneities that are outside of the scope of this work. Figure 4.3.7 compares local and global fitting methods with one and two-term fits to trajectories of and demonstrates the presence of two diffusive populations. DiI molecules in a homogeneous lipid environment should diffuse at

	Global	Local
D_1 ($\mu m^2 s^{-1}$)	2.5 ± 0.09	2.6 ± 0.2
D_2 ($\mu m^2 s^{-1}$)	0.46 ± 0.04	0.41 ± 0.05
α_1 (%)	50 ± 1	48 ± 2
α_2 (%)	50 ± 1	52 ± 2

Table 4.2: Diffusion coefficients and population amplitudes from the data presented in Figure 4.3.8: DiI diffusing in a POPC bilayer. Both Global and Local fitting estimated similar diffusion coefficients and population amplitudes, but the standard deviations were considerably smaller for Global fitting.

around 2 $\mu m^2 s^{-1}$, and indeed, the faster population has $D_1 = 2.5$ $\mu m^2 s^{-1}$, which is near to this value (Figure 4.3.8). However, the analysis also reveals a slower diffusive population ($D_2 = 0.4$ $\mu m^2 s^{-1}$; Table 4.2), due to sample inhomogeneities. Regardless of the origin of these inhomogeneities that gave rise to the surprising second diffusive populations, global fitting outperformed local fitting in terms of diffusion coefficient estimation precision.

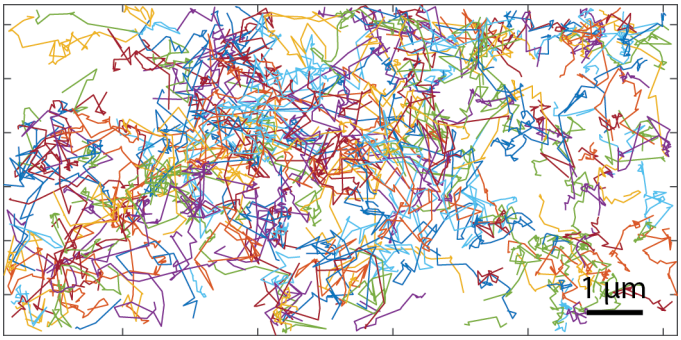

Figure 4.3.6: 1-palmitoyl-2-oleoyl-sn-glycero-3-phosphocholine (POPC, Avanti) and the lipid dye DiI C12 (ThermoFisher) were mixed together in chloroform at final concentrations of 5 mg mL^{-1} and $1 : 4 \times 10^{-5}$ from stock concentration, respectively. 1.5 wt% agarose and then 30 μL of lipid/dye solution were spin coated onto a coverslip at 5000 rpm. 30 μL PBS was then placed on top of the dried lipids and a top coverslip was then added. The sparse DiI was imaged with a 561 nm fluorescence excitation laser (Coherent Sapphire 560-50) with a power density of 0.01 mW μm^{-2}. The sample was imaged on a Photometrics Evolve EMCCD camera and the diffusing DiI was then localized and tracked. Shown here are the 336 trajectories used in the analysis shown in Figures 4.3.7 and 4.3.8.

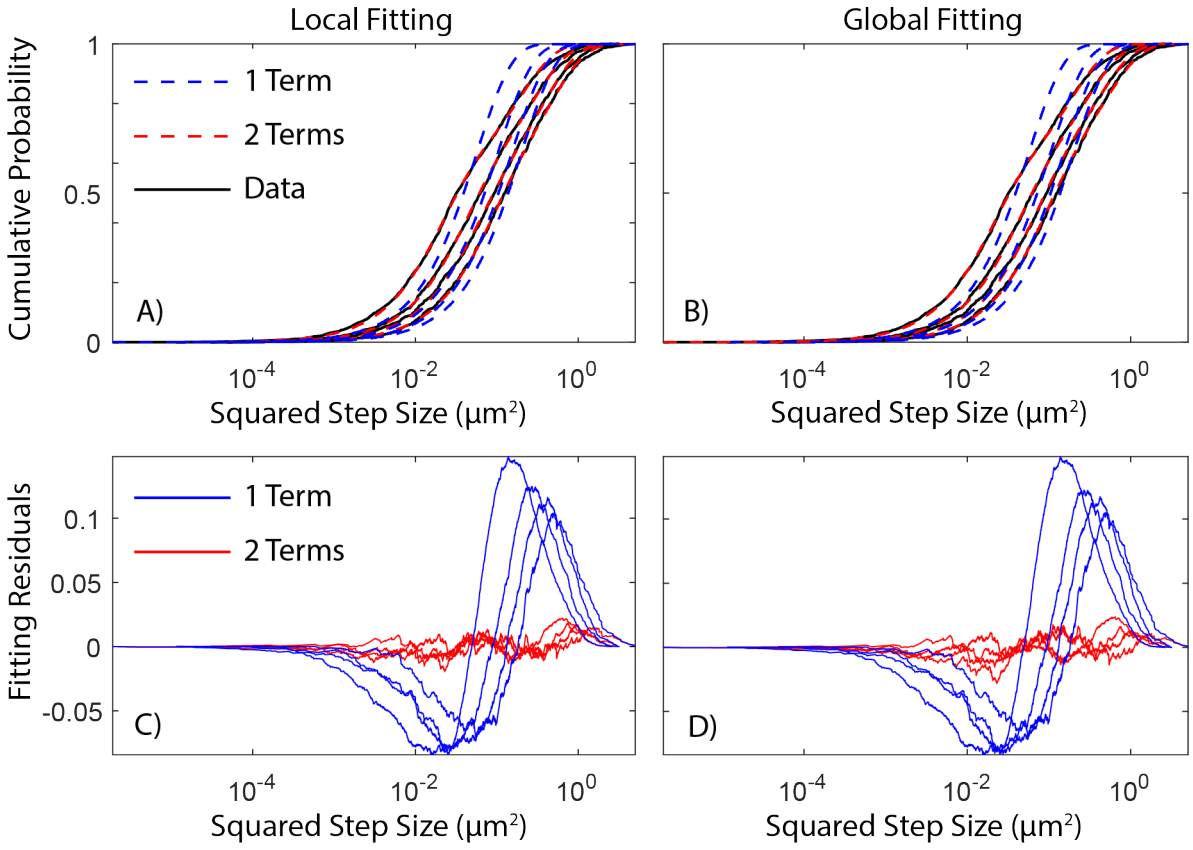

Figure 4.3.7: A) and B) Fitting results of the Local and Global fitting algorithms applied to the trajectories of DiI shown in Figure 4.3.6. The data (solid lines, time lags $\tau = 0.01$ s, 0.02 s, 0.03 s, and 0.04 s) are overlain with the dashed fitting results according to the legend. The red dashed lines (fits using two diffusive terms) conform tightly to the data, while a single term does not fit well. C) and D) Residuals from the fits in A) and B). For both the local and global algorithms, the results of the fitting are very similar and it is clear from the residuals that two diffusive terms are required to explain the data.

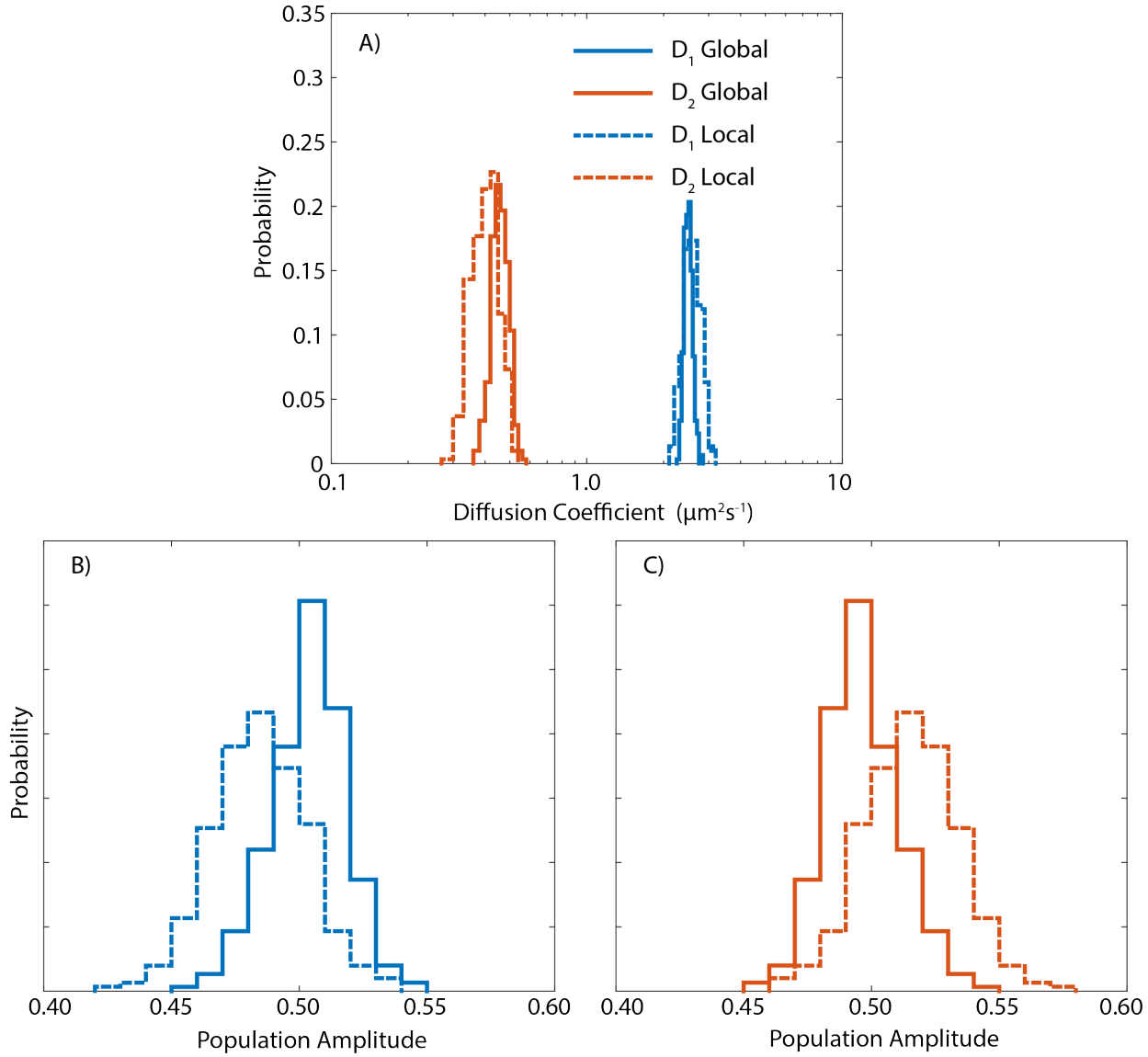

Figure 4.3.8: A) Distributions of diffusion coefficients of DiI diffusing on a POPC bilayer. The Local fitting algorithm (dashed lines) resulted in less precise estimations of two diffusive populations, corresponding to the relative populations of fast and slow modes of DiI diffusion. B) and C) Population weights of the first and second diffusive populations, respectively, measured with the Global and Local CPD algorithms. All methods resulted in similarly precise estimations of a 50% distribution of fast and slow DiI. Table 4.2 shows the mean and standard deviations of the diffusion coefficients and population amplitudes.

4.4 Conclusions

In this work, we introduced global CPD fitting, a straightforward method to enhance the precision and simplicity of a commonly used diffusion coefficient estimation technique by combining redundant parameters. Instead of estimating the mean squared displacements of diffusing particles only to then fit those results in a second curve fitting step (Figure 4.3.1B), we presented a multi-domain fit that accomplishes both steps at once. The resulting reduction in degrees of freedom of the fitting operation increases the estimation precision and robustness by increasing the rigidity of the fitting parameters. The enhanced precision and robustness of global fitting introduced here enables precise, quantitative investigations of systems with increased complexity, for instance motion that includes more than one or two diffusive components. This global fitting method is not restricted to freely diffusing molecules, and can be extended to any type of diffusion such as confined diffusion, diffusion with flow, or anomalous diffusion.

CHAPTER V

Conclusions and Future Directions

In preceding chapters, I dissected the commonly used biophysical analysis techniques to innovate methods of improving the quantity and quality of analytical results produced by those techniques. In this chapter, I will review the ramifications of these foundational improvements and propose possible extensions. First, we showed how refinements to the analytical interpretation of point spread functions (PSFs) can simplify data analysis of directed motion of point light sources (Chapter 2). [29] Further work could investigate directed motion that incorporates in-frame diffusion along that directed path. Second, even though we showed for the first time that STICS could be used to measure diffusion of single molecules inside bacteria (Chapter 3), [34] the Gaussian fitting function in the spatiotemporal image correlation spectroscopy (STICS) analysis protocol inaccurately describe the motion of quickly diffusing molecules confined to small volumes. Future work for this project could include modifications to the Gaussian fitting function used in STICS so that in-frame motion and confinement does not introduce biases. Third, we combined the multi-step fitting procedure required by the commonly used cumulative probability distribution (CPD) method of estimating the diffusion coefficients into a single fitting step (Chapter 4). This consolidation increases the robustness and precision of the diffusion coefficient estimates because of the

reduced number of fitting parameters. Further work for this project could involve applying the global fitting method to the STICS algorithm, that also makes use of a multi-step fitting process and could similarly benefit from a reduction in the number of free fitting parameters.

5.1 Maximizing the information content of point spread functions

Successful implementation of single-molecule super-resolution microscopy requires several important assumptions including particle immobility during the integration time of the camera. [34] By imposing these assumptions, information about the system is lost; this concept was impetus for the work reported in Chapter 2. There, I exploited a more accurate description of the point spread function of a point light source undergoing directed motion as a way to glean information directly from individual image frames that otherwise must be calculated as an average value from multiple image frames.

This new fitting function produced only a marginal improvement over the asymmetric Gaussian method of estimating directed in-frame motion reported in Yüce et al. [55] Their method, however, requires an empirical calibration curve and this increases the complexity of the experiment. In addition, any errors present in the calibration curve will degrade the performance of the algorithm. By instead appropriately modifying the fitting function used for the localization of the molecules, we found that the in-frame displacement can be estimated directly without requiring a separate calibration step.

The modified PSF, Sum of Error functions (SErf), accounts for in-frame directed motion but not diffusion. Interesting further work might include using the work done by Schuster et

al. [25] that analyzed the isotropic broadening of the PSF of diffusive molecules to calculate diffusion coefficients from single images. The path taken by a molecule undergoing directed motion might also include diffusion that would broaden the PSF. If one were to calibrate PSF broadening as a function of diffusion coefficient, the transverse width fitting parameter of the SErf PSF fitting function could be used in conjunction with the displacement fitting parameter to estimate both the diffusion coefficient and the velocity of motion in each image frame.

5.2 Maximizing the information content of fluorescence data

Localizing particles with PSF fitting restricts the data to be analyzed to include only regions with high SNR. By selecting only the ∼500 nm region around the location of a point light source, vast regions of imaging data empty of meaningful signal are excluded completely from the analysis. This selection process is a double-edged sword, however, because the selection process either requires image analysis algorithms or human selection that prioritizes the brightest or most obvious molecules. Chapter 3 investigated spatiotemporal image correlation spectroscopy [30] (STICS) as a potential method for unbiased measurement of diffusion inside bacteria because STICS does not preferentially select regions of interest. Localization-based algorithms, in contrast, preferentially select regions of interest around molecules that do not move much during the image capture—thereby excluding images of the molecule when it takes larger than average steps. STICS removes this selection bias and further work may uncover a more appropriate fitting function that also does not suffer from in-frame motion and confinement artifacts.

In the end, we found in Chapter 3 that STICS produced two biases, and we further discovered that these biases are caused independently by in-frame motion and confinement. Both of these biases could be removed by using a modified STICS fitting function that explicitly accounts for the two situations.

The first bias, due to in-frame motion, is caused by the finite image capture time of the EMCCD camera. STICS assumes that all of the collected photons were emitted at the same time, e.g., at the single time-point recorded for that particular image. We know, however, that the molecule moved around during a range of times that span the camera integration time. Therefore, when the correlation function of a pair of images separated by a time-lag of two frames, the result is a correlation function that includes a range of time-lags instead of a single time-lag of two frames (see Figure 5.2.1). The result of this is a correlation function calculated from data including in-frame diffusion and so will include a linear combination of Gaussian correlation functions that correspond to a range of time-lags. In the case where the molecule is diffusing slowly, the in-frame trajectory can be approximated as a single point, which obviates the fact that the photons were collected during a range of times and the correlation function will assume the expected Gaussian shape. If there is in-frame motion, however, the step size distributions from a range of time-lags are included in the correlation function for a single time-lag. Equation 5.2.1 describes this idea by the integration of step-size distributions over the range of time lags brought on by the finite camera integration time. A future investigator could investigate this equation further and perhaps account for the in-frame motion bias apparent when using STICS to estimate the diffusion coefficient of

a molecule diffusing so quickly so as to become blurred in a single image (see Figure 5.2.2).

$$G_{\text{blur}} = \frac{\gamma}{N}\left[\left(\int_{\tau_0-t_{\text{frame}}}^{\tau_0+t_{\text{frame}}} g(\tau_{\text{eff}})d\tau_{\text{eff}}\right)^{-1} \times \int_{\tau_0-t_{\text{frame}}}^{\tau_0+t_{\text{frame}}} g(\tau_{\text{eff}})N(0,4D\tau_{\text{eff}})d\tau_{\text{eff}}\right] \otimes W_{\text{PSF}}(\xi,\chi). \tag{5.2.1}$$

Confinement to the small volume of a bacterium also leads to a second bias in the estimation of the diffusion coefficient by STICS. Even without in-frame motion, confinement was found to cause bias and the likely reason for this is shown in Figure 5.2.3 to be the effect that reflective boundary conditions have on the actual distribution of step sizes. Further work on this subject would entail solving Equation 5.2.2

$$G_{\text{conf}} = \frac{\gamma}{N}\left[2\sum_{n,\text{int.}} p(\xi+nL,\chi+nL,\tau)\right] \otimes W(\xi,\chi), -\frac{L}{2} \leq \xi,\chi \leq \frac{L}{2} \tag{5.2.2}$$

to find a modified fitting function for the correlation functions (Equation 5.2.3)

$$G(\xi,\chi,\tau) = \frac{\gamma}{N}p(\xi,\chi,\tau) \otimes W_{\text{PSF}}(\xi,\chi) \tag{5.2.3}$$

calculated by STICS. For instance, the derivation of the STICS analysis procedure assumes that the step size distribution is Gaussian. [70] However, because the steps taken inside a small bacterium are curtailed by the boundaries of the cell, the Gaussian approximation no longer holds (see Figure 5.2.3). With the solution described here, the STICS correlation functions may be fit with the correct fitting function and the bias due to tight confinement might be eliminated.

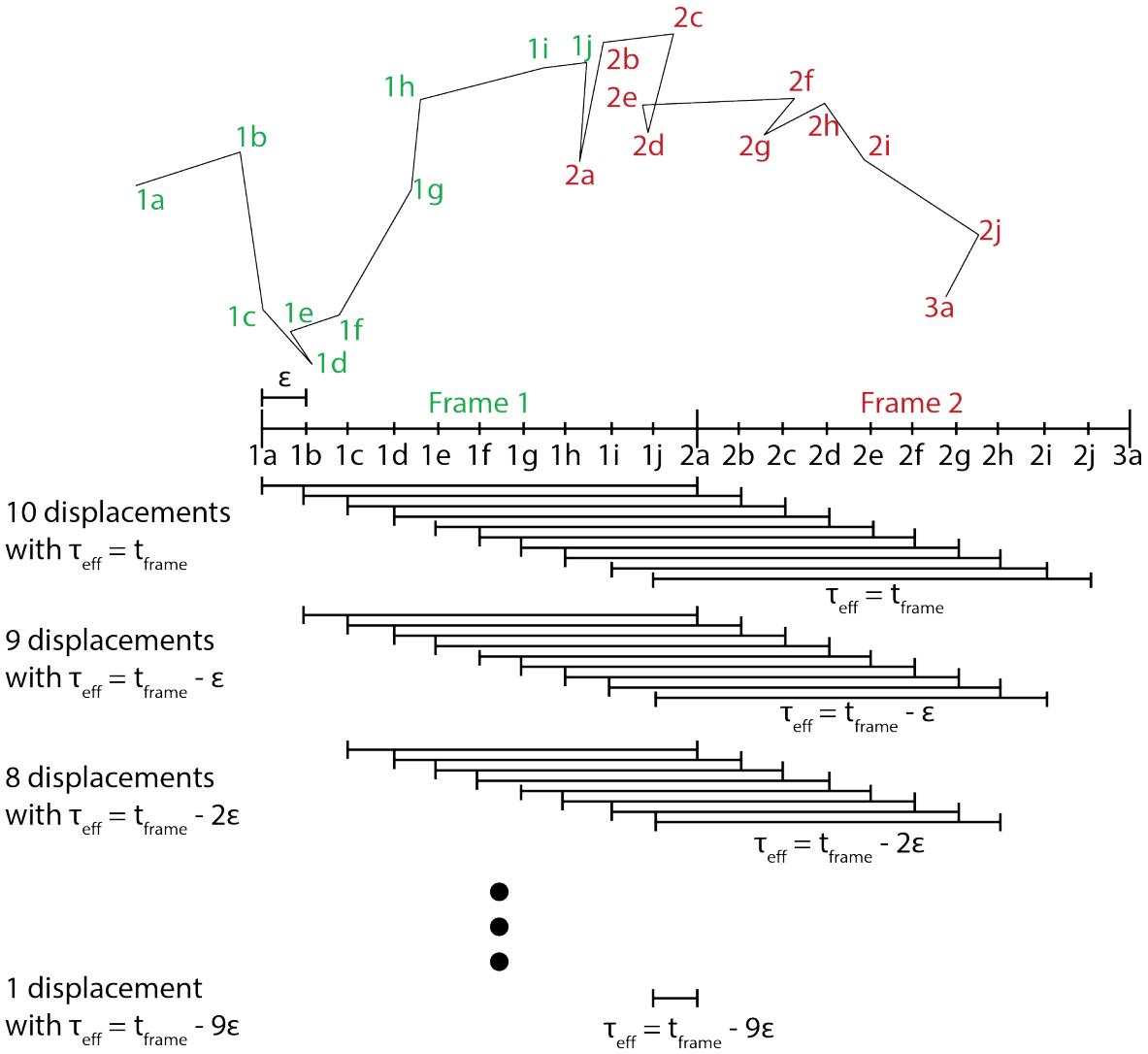

Figure 5.2.1: Top, a diffusive trajectory that spans two imaging frames with 10 sub-steps shown, although this argument can be made with any number of sub-steps. The correlation function for the actual motion includes correlations with a time lag of not only tframe, but also smaller time lags (such as from time point 1b to 2a) and larger time lags (such as from time point 1a to 2b) The step size distribution of normal diffusion given a time lag of $t_{\text{frame}} - 9\epsilon$, which corresponds to the sub-frame motion from the final time step in the first frame to the first time step in the second frame occurs least frequently and so should not contribute as much to the final correlation function.

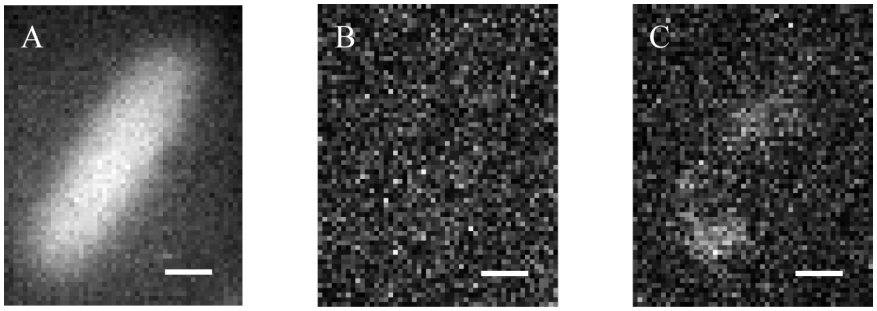

Figure 5.2.2: (A) Time-averaged fluorescence image obtained from summing a time series of fluorescence images of mMaple3 diffusion inside an *E. coli* cell. (B) A single image ($t_{\text{frame}} = 40$ ms) from that time series shows a cell after the mMaple3 molecule has bleached. (C) An image of the same *E. coli* cell in (B) two frames earlier, showing a typical mMaple3 molecule diffusing so rapidly that it is diffuse over nearly the entire bacterium. Scale bars: 0.5 μm. [34]

5.3 Maximizing the precision of existing analysis methods

Chapter 4 analyzed the effect of combining the multiple steps used to estimate diffusion coefficients from single particle trajectories in the cumulative probability distribution (CPD) fitting method into a single fitting step. By fitting all of the CPDs at once, the total number of fitting parameters can be reduced from dozens to fewer than ten. This reduction in the degrees of freedom in the fitting procedure was shown to both increase the estimation precision and the numerical stability of the fit. In other words, diffusion parameters could be estimated more precisely and the fitting algorithm was less likely to estimate non-physical values.

It is generally known in the scientific community that, given enough fitting parameters, one may be able to successfully fit an elephant. By *removing redundant fitting parameters*, more accurate, robust, and physically meaningful results are expected because the parameter

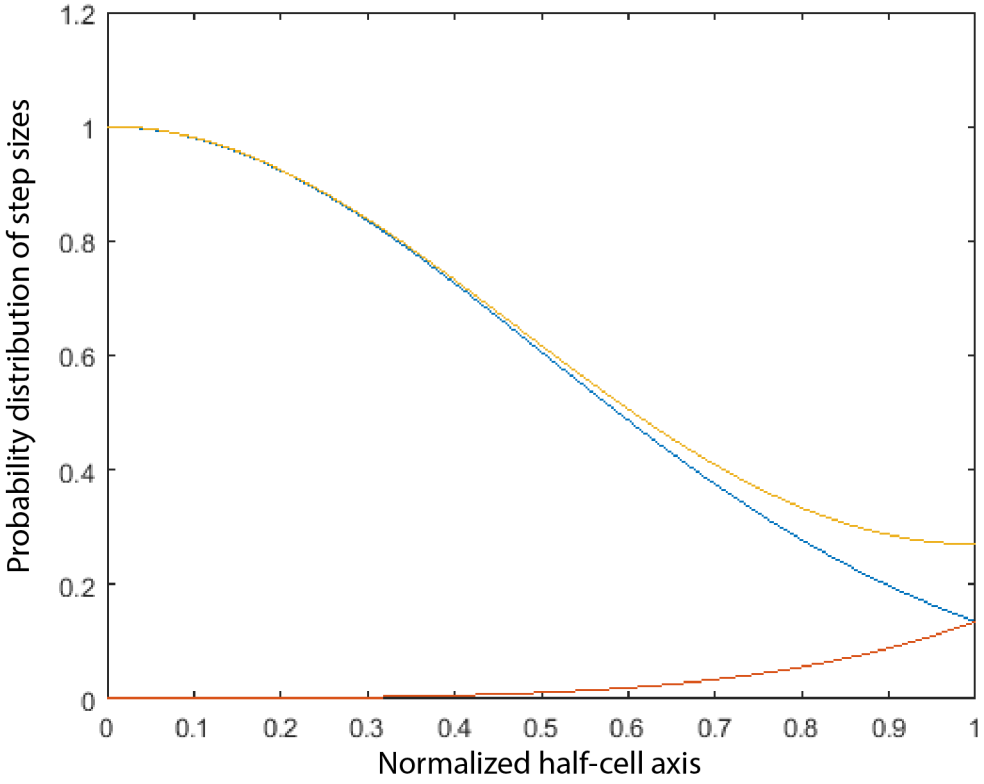

Figure 5.2.3: The effect of a reflective boundary at the end of the long axis of a bacterial cell on the actual distribution of step sizes taken by diffusing particles. The blue line is the normal (Gaussian) distribution of step sizes. When steps at the tail of the blue line attempt to pass the reflective boundary at 1 on the x axis, the steps are reflected as shown by the red line. These retrograde steps are indistinguishable from the smaller steps present in the normal distribution of steps, and so the blue and red lines add to become the yellow line. This yellow distribution is the effective step size distribution of step sizes given a reflective boundary at 1 on the x axis. It is for this reason that the step size distributions estimated by STICS are non-Gaussian in shape.

space of possible local minima will be restricted to a smaller hypervolume. For instance, this global fitting method could also be applied to the fitting procedure used in STICS. To estimate diffusion coefficients and diffusive population weights, STICS uses a two-step fitting process where first the correlation functions are fit to Gaussians and then the widths of the Gaussians are fit to a particular model of diffusion, such as a linear relationship between mean squared displacement and time lag for unconfined diffusion. Along the same lines as the global fitting method used in Chapter 4, the collection of Gaussians may be fit in a single step. This will reduce the number of fitting parameters and will likely increase the fitting precision and will certainly increase the robustness of the fit in the same way as for the CPD analysis.

5.4 Future Directions

This dissertation has focused on improvements of data analysis methods that have already been utilized in simplified forms. These improvements lead to important questions that span the scientific community. The design and implementation of obscure analysis algorithms will always be able to be improved, but as complexity increases, the size of the population of experts with general working knowledge will decrease. Will computer automation lead to dozens of analysis software packages that are treated more or less as black boxes into which data is injected and out of which come results that merely use cited sources as proof of validity? Additionally, as the precision of analysis methods increase, previously unconsidered anomalies will inevitable come to light, and this presents many interesting new areas of research. For example, diffusion in the crowded environment [110] may push the results of the current state of diffusion coefficient estimation to the limits of interpretability.

Those using techniques described in this thesis should recognize some notable caveats. Firstly, experimental design must be carefully optimized in order to use the fitting function for directed motion described in Chapter 2. The derivation of the fitting function assumes that the path taken by the fluorescent molecule is straight and that the velocity is constant during the image acquisition. The camera frame rate may be reduced to lessen the likelihood that the molecule deviates appreciably from a straight path, or increased to improve the precision of the displacement estimation, for example.

Further caveats include the loss of position information when using the STICS and CPD analysis methods. STICS completely obscures the original location of pixel intensities. A data set consisting of large images may be split into many regions and analyzed independently thereby retaining coarse-grained spatial information, but the edge effects associated with such Fourier methods dictate a strong correlation between region size and signal-to-noise ratio. CPD analysis also obscures position information by reducing all molecule trajectories to a single list of step sizes. To assign a diffusion population identity to a particular step in a trajectory is an intriguing problem. It could be very useful to know when a molecule is diffusing according to its fast or slow diffusive state, for example.

The work in this thesis makes exclusive use of a single non-linear least-squares minimization algorithm provided with Matlab, but the reader should recognize that there are methods that directly compete and cab serve as drop-in improvements while sacrificing only computational speed (maximum likelihood estimation [18] is one). Furthermore, one should be curious about the commonly implemented Gaussian approximation of the Airy disk point spread function. This approximation affects SPT and STICS differently, however. Both Gaussian and Airy disk functions are symmetric, so particle localization will be unbiased regardless of the localization fitting function used. The difference in peak width between the

two functions could conceivably affect the precision of localization, but I think the signal-to-noise ratios typically seen in single-molecule fluorescence data are much too low for this to be noticable. STICS, on the other hand, directly relies on the peak width of correlation functions that are affected by the shape of the microscope point spread function. It is therefore likely that the Gaussian approximation used in the derivation of STICS negatively affects the method's estimation precision.

Finally, one might also ask whether the future developments in the field of single-molecule super-resolution microscopy will lie on the data acquisition or data analysis side. There was an explosion of analysis methods in this field over the past 10 years and this collection needs to be combined and simplified. The fluorophores used are a severe limitation inherent to this field because they dictate a finite information budget that corresponds to the number of photons that can be emitted by a single fluorophore. Imaging systems are approaching perfect photon sensitivity and so the greatest improvements will be found mostly on the analysis side from more accurate and holistic algorithms. When analyzing data, the experimental parameters should be specified—not the analysis method.

APPENDICES

APPENDIX A
Code for Chapter 3: bpassDJR, dataGen, dataParse, gaussFit

```
%
% NAME:
%               bpassDJR
% PURPOSE:
%               Implements a real-space bandpass filter that suppresses
%               pixel noise and long-wavelength image variations while
%               retaining information of a characteristic size.
%
% CATEGORY:
%               Image Processing
% CALLING SEQUENCE:
%               res = bpass( image_array, lnoise, lobject, threshold, lzero )
% INPUTS:
%               image: The two-dimensional array to be filtered.
%               lnoise: Characteristic lengthscale of noise in pixels.
%                       Additive noise averaged over this length should
%                       vanish. May assume any positive floating value.
%                       May be set to 0 or false, in which case only the
%                       highpass "background subtraction" operation is
%                       performed.
%               lobject: (optional) Integer length in pixels somewhat
%                       larger than a typical object. Can also be set to
%                       0 or false, in which case only the lowpass
%                       "blurring" operation defined by lnoise is done,
%                       without the background subtraction defined by
%                       lobject. Defaults to false.
%               threshold: (optional) By default, after the convolution,
%                       any negative pixels are reset to 0. Threshold
%                       changes the threshhold for setting pixels to
%                       0.  Positive values may be useful for removing
```

```
%                     stray noise or small particles. Alternatively, can
%                     be set to -Inf so that no threshholding is
%                     performed at all.
%
% OUTPUTS:
%               res:   filtered image.
% PROCEDURE:
%               simple convolution yields spatial bandpass filtering.
% NOTES:
% Performs a bandpass by convolving with an appropriate kernel. You can
% think of this as a two part process. First, a lowpassed image is
% produced by convolving the original with a gaussian. Next, a second
% lowpassed image is produced by convolving the original with a boxcar
% function. By subtracting the boxcar version from the gaussian version, we
% are using the boxcar version to perform a highpass.
%
% original - lowpassed version of original => highpassed version of the
% original
%
% Performing a lowpass and a highpass results in a bandpassed image.
%
% Converts input to double. Be advised that commands like 'image' display
% double precision arrays differently from UINT8 arrays.

% MODIFICATION HISTORY:
%               Written by David G. Grier, The University of Chicago, 2/93.
%
%               Greatly revised version DGG 5/95.
%
%               Added /field keyword JCC 12/95.
%
%               Memory optimizations and fixed normalization, DGG 8/99.
%               Converted to Matlab by D.Blair 4/2004-ish
%
%               Fixed some bugs with conv2 to make sure the edges are
%               removed D.B. 6/05
%
%               Removed inadvertent image shift ERD 6/05
%
%               Added threshold to output. Now sets all pixels with
%               negative values equal to zero. Gets rid of ringing which
%               was destroying sub-pixel accuracy, unless window size in
```

```
%              cntrd was picked perfectly. Now centrd gets sub-pixel
%              accuracy much more robustly ERD 8/24/05
%
%              Refactored for clarity and converted all convolutions to
%              use column vector kernels for speed. Running on my
%              macbook, the old version took ~1.3 seconds to do
%              bpass(image_array,1,19) on a 1024 x 1024 image; this
%              version takes roughly half that. JWM 6/07
%
%      This code 'bpass.pro' is copyright 1997, John C. Crocker and
%      David G. Grier. It should be considered 'freeware'- and may be
%      distributed freely in its original form when properly attributed.
%
%
% 2016 edits by David J. Rowland, The University of Michigan:
% added lzero to the input list instead of just declaring it inside the function.

if nargin < 3, lobject = false; end
if nargin < 4, threshold = 0; end

normalize = @(x) x/sum(x);

image_array = double(image_array);

if lnoise == 0
  gaussian_kernel = 1;
else
  gaussian_kernel = normalize(...
    exp(-((-ceil(5*lnoise):ceil(5*lnoise))/(2*lnoise)).^2));
end

if lobject
  boxcar_kernel = normalize(...
      ones(1,length(-round(lobject):round(lobject))));
end

% JWM: Do a 2D convolution with the kernels in two steps each. It is
% possible to do the convolution in only one step per kernel with
%
  % gconv = conv2(gaussian_kernel',gaussian_kernel,image_array,'same');
  % bconv = conv2(boxcar_kernel', boxcar_kernel,image_array,'same');
%
```

```
% but for some reason, this is slow. The whole operation could be reduced
% to a single step using the associative and distributive properties of
% convolution:
%
  % filtered = conv2(image_array,...
  %   gaussian_kernel'*gaussian_kernel - boxcar_kernel'*boxcar_kernel,...
  %   'same');
%
% But this is also comparatively slow (though inexplicably faster than the
% above). It turns out that convolving with a column vector is faster than
% convolving with a row vector, so instead of transposing the kernel, the
% image is transposed twice.

gconv = conv2(image_array',gaussian_kernel','same');
gconv = conv2(gconv',gaussian_kernel','same');

if lobject
  bconv = conv2(image_array',boxcar_kernel','same');
  bconv = conv2(bconv',boxcar_kernel','same');

  filtered = gconv - bconv;
else
  filtered = gconv;
end

% commented out because why do it in the first place?
% % Zero out the values on the edges to signal that they're not useful.
% lzero = max(lobject,ceil(5*lnoise));
% lzero=0;

filtered(1:(round(lzero)),:) = 0;
filtered((end - lzero + 1):end,:) = 0;
filtered(:,1:(round(lzero))) = 0;
filtered(:,(end - lzero + 1):end) = 0;

% JWM: I question the value of zeroing out negative pixels. It's a
% nonlinear operation which could potentially mess up our expectations
% about statistics. Is there data on 'Now centroid gets subpixel accuracy
% much more robustly'? To choose which approach to take, uncomment one of
% the following two lines.
% ERD: The negative values shift the peak if the center of the cntrd mask
% is not centered on the particle.
```

```
% res = filtered;
filtered(filtered < threshold) = 0;
res = filtered;
```

```
function [v, simProps] = dataGen(varargin)
%
% NAME:
%      dataGen
% PURPOSE:
%      Generates space and time resolved single-molecule imaging data of a
%       single diffuser.
% CATEGORY:
%      Data Simulation
% CALLING SEQUENCE:
%      [v, simProps] = dataGen(boundaryCondition, blurFlag);
% INPUTS:
%      varargin: use paired inputs to set the property (input 1) to the
%          value (input 2) desired.
%
%      Properties:          Descriptions:
%
%      D                    diffusion coefficient in microns^2/s
%
%      tFrame               image frame integration time in seconds
%
%      pixSize              Pixel size in micrometers
%
%      psfSize              standard deviation (width) of the microscope's
%                           point spread function in micrometers. i.e.
%                           FWHM = sqrt(2*log(2)) * psfSize
%
%      celSize              1x2 vector: [width (diameter), height
%                           (length)] of the bounding box (cylinder)
%                           depending on the confinement condition
%
%      nFrames              number of frames to be simulated
%
%      SNR                  signal to noise ratio (ratio of maximum
%                           signal amplitude to standard deviation of
%                           background noise)
%
%      confBool             1 for confined to the interior of a
%                           cylinder, and 0 for free diffusion.
%
%      blurFlag             1 for blurry motion, 0 for 'stroboscopic
```

```
%                           illumination'
%
% OUTPUTS:
%      v:               simulated image time sequence
%      simProps:        properties of the simulation in the Matlab structure
   format
% PROCEDURE:
%      1. Simulate molecular trajectory
%      2. Evaluate pixel intensities
%    3. Add noise
% MODIFICATION HISTORY:
%      Written by David J. Rowland, The University of Michigan, 3/16.
% NOTES:
%      This code 'dataGen.m' should be considered 'freeware'- and may be
%      distributed freely in its original form when properly attributed.
%
%      For testing purposes, this line makes a movie of confined, blurry
%      diffusion.
%
%      v=dataGen('confined', 1)


%% Default simulation parameters
D = .01;              % diffusion coefficient in microns^2/s
tFrame = .05;      % frame integration time in seconds
pixSize = .049;       % width of pixels in microns
psfSize = .098;       % s.d. of the psf in microns
celSize = [1,3];      % [width, length] of confinement cylinder in microns
nFrames = 1e3;     % number of frames in the simulated movie
SNR = 20;          % signal to noise ratio for added white noise (0:inf)
confBool = 1;         % 'confined' or 'unconfined'
blurFlag = 1;         % include blur subframes or not

% algorithmically-determined image size designed to disallow edge effects
imSize = ceil(celSize/pixSize+4*ceil(psfSize/pixSize));

% check variable size
if prod([imSize(1:2),nFrames])*8/1e9>1
    warning('video is over a GB. manually pass this block if you wish to continue')
end

% minimum increment of speed in units of microns^2/subframe
```

```
dtRef = 0.0001;

% initialize simulation properties structure
simProps.D = D;
simProps.tFrame = tFrame;
simProps.pixSize = pixSize;
simProps.psfSize = psfSize;
simProps.celSize = celSize;
simProps.nFrames = nFrames;
simProps.SNR = SNR;
simProps.blurFlag = blurFlag;
simProps.nSubs = [];
simProps.dtRef = dtRef;
simProps.confBool = confBool;

% if any sim parameters are included as inputs, change the simulation
% parameters mentioned
if ~rem(nargin,2)
    fNames=fieldnames(simProps);
    for ii=1:2:nargin
        whichField = strcmp(fNames,varargin{ii});

        if all(~whichField)
            warning('Check spelling. Parameter change may have not occurred')
        end

        eval([fNames{whichField} ' = varargin{ii+1}'])
        eval(['simProps.' fNames{whichField} ' = ' fNames{whichField},';'])
    end

elseif ~rem(nargin,1)
    warning('use paired inputs')
    v=[];
    return
end

% number of subframes required
nSubs = ceil(D*tFrame/dtRef);

% update the value of the diffusion coefficient since rounding may change it.
D = nSubs*dtRef/tFrame;
```

```
%% Trajectory generation
if confBool
    % confined particle trajectory
    mLocs = zeros(3, nFrames);
    for ii = 1 : nFrames*nSubs-1 % this loop can be compiled to mex64 to increase
        its speed
        % three 1d steps pulled from normal distribution with variance 2*dtRef
        step = sqrt(2*dtRef) * randn(3,1);

        candPos = mLocs(:,ii) + step;
        prevPos = mLocs(:,ii);

        % the ordering in the celSize vector matters because of this line:
        r = celSize(1)/2;

        % if the candidate position is outside of the cylinder in the x/z
            dimensions, reflect the step
        % against the inside of the cylinder. path length is preserved.
        if sqrt(sum(candPos([1,3]).^2)) > celSize(1)/2
            m = (candPos(3)-prevPos(3)) / (candPos(1)-prevPos(1));
            b = candPos(3)-m*candPos(1);

            xi=[(-m*b+sqrt(-b^2+r^2+m^2*r^2))/(1+m^2),...
                (-m*b-sqrt(-b^2+r^2+m^2*r^2))/(1+m^2)];
            yi=m*xi+b;

            % there are two solutions. the one closest to the candidate position is
                chosen. the farther
            % one is on the other side of the cell.
            whichone = (xi-candPos(1)).^2 + (yi-candPos(3)).^2 + (xi-prevPos(1)).^2
                + (yi-prevPos(3)).^2;
            xip = [xi(find(whichone == min(whichone))),yi(find(whichone ==
                min(whichone)))];

            normv = -xip/sqrt(sum(xip.^2));
            l = sqrt(sum((candPos([1,3])'-xip).^2));
            pf = 2*sum((prevPos([1,3])'-xip).*normv)*normv-(prevPos([1,3])'-xip);
            pf = pf/sqrt(sum(pf.^2))*l+xip;

            % replace x/z components of the position with the reflected x/z
                components
            out=candPos;
```

```
            out([1,3])=pf;

            candPos=out;
        end

        % if the candidate position is outside of the cylinder in the y dimension
            (cell's long axis)
        if candPos(2) < -celSize(2)/2
            candPos(2) = 2*-celSize(2)/2 - candPos(2);
        end
        if candPos(2) > celSize(2)/2
            candPos(2) = 2*celSize(2)/2 - candPos(2);
        end

        mLocs(:, ii+1) = candPos;
    end
else

    % unconfined particle trajectory
    mLocs=cumsum(sqrt(2*dtRef) * randn(3,nFrames*nSubs),2);
end

%% movie generation

if blurFlag
    % arrange tracks for subframe averaging
    tr_x = zeros(nSubs, nFrames);
    tr_y = zeros(nSubs, nFrames);
    for ii = 1:nFrames
        tr_x(:, ii) = mLocs(1, 1+(ii-1)*nSubs : ii*nSubs);
        tr_y(:, ii) = mLocs(2, 1+(ii-1)*nSubs : ii*nSubs);
    end
else
    % just use the first subframe from each frame
    tr_x = mLocs(1, 1:nSubs:end);
    tr_y = mLocs(2, 1:nSubs:end);
end

% shift to positive values
tr_x=tr_x+celSize(1)/2;
tr_y=tr_y+celSize(2)/2;
```

```
% noiseless pixel intensities
v=zeros([imSize(1:2),nFrames]); cx = 0; cy = 0;
for ii=-2*psfSize:pixSize:celSize(1)+2*psfSize   % x pixel locations with padding
    cx = cx+1;
    for jj=-2*psfSize:pixSize:celSize(2)+2*psfSize % y pixel locations with padding
        cy = cy+1;

        % symmeteric gaussian function approximation of Airy Disk
        v(cx,cy,:) = mean(exp(-((ii-tr_x).^2+(jj-tr_y).^2)/2/psfSize^2),1);
    end
    cy = 0;
end

% add white noise
v = v + 1/SNR*randn(size(v));

simProps.D = D;
simProps.tFrame = tFrame;
simProps.pixSize = pixSize;
simProps.psfSize = psfSize;
simProps.celSize = celSize;
simProps.nFrames = nFrames;
simProps.SNR = SNR;
simProps.blurFlag = blurFlag;
simProps.nSubs = nSubs;
simProps.dtRef = dtRef;
simProps.confBool = confBool;
end
```

```
function resStruct = dataParse(data, mask)
%
% NAME:
%      dataParse
% PURPOSE:
%      Analyze single-molecule imaging data with tracking or STICS
% CATEGORY:
%      Image Processing
% CALLING SEQUENCE:
%      resStruct = dataParse(data,flag);
% INPUTS:
%      data:             x by y by t movie of fluorescence data
%      mask:             (optional) 2d binary roi selection mask
% OUTPUTS:
%      resStruct:        fitting result 'structure'
% PROCEDURE:
%      1. Peak fitting, then MSD calculation a la single molecule analysis
%      2. Correlation function calculation and width estimation
%      3. Diffusion coefficient estimation by MSD fitting
% MODIFICATION HISTORY:
%      Written by David J. Rowland, The University of Michigan, 3/16.
% NOTES:
%      This code 'dataParse.m' should be considered 'freeware'- and may be
%      distributed freely in its original form when properly attributed.
%
%      For testing purposes, run this script:
%
%      [v,sP]=dataGen('D',.1,'tFrame',.05,'nFrames',500);
%      r = dataParse(v)
%      plot(r.iMSDs); hold all
%      plot(r.MSDs)
%      plot(r.MSDd); hold off


%% analysis parameters
tFrame = 0.05;        % camera integration time in seconds
nTau = 5;            % number of smallest time lag values to use (excluding 0)
pixelSize = .049;     % pixel size in microns
nFrames = size(data,3); % number of frames in the data
yesOverlap = 1;       % use overlapping time lags?
isConfined = 1;       % use confined MSD curve fit?
```

```
padStyle = 1;          % padding style for STICS analysis
                       % 1 : pad zeros, 2: pad mean, 3: pad mean outside mask

%% single molecule tracking analysis

% spot fitting.
fitP = zeros(nFrames, 6);
parfor ii = 1:nFrames
    % this loop may be parallelized by simply replaceing 'for' with 'parfor'
    % and starting a parallel pool before running the code.
    fitP(ii,:) = gaussFit(data(:,:,ii));
end

% 'tracking'. missed spots will register as nans.
tr = fitP(:,1:2) * pixelSize;

% calculate mean squared displacements, one for each time lag value
MSDs=zeros(nTau,2);
for ii=1:nTau
    if yesOverlap                   % overlapping frame pairs
        indvec1=ii+1:nFrames;
        indvec2=1:nFrames-ii;
    elseif ~yesOverlap              % nonoverlapping frame pairs
        indvec2=1:ii:nFrames;
        indvec1=indvec2(1:end-1);
        indvec2=indvec2(2:end);
    end

    % mean squared displacements vs time lag
    MSDs(ii,:)=nanmean((tr(indvec2,:)-tr(indvec1,:)).^2,1);
end

%% STICS analysis

% pad images
switch padStyle
    case 1
        data = padZeros(data);
    case 2
        data = padMean(data);
    case 3
        data = padMask(data,mask);
```

```
end

% time-space correlation function calculation
if yesOverlap                          % overlapping frame pairs
    famps=abs(fft(fft(fft(data,[],1),[],2),[],3)).^2;
    STCorr = fftshift(fftshift(real(ifft(ifft(ifft(famps...
        ,[],1),[],2),[],3)),1),2)/numel(famps)/mean(data(:))^2-1;
    STCorr = STCorr(:,:,2:nTau+1);

elseif ~yesOverlap                     % nonoverlapping frame pairs
    vFft=fft2(data);
    STCorr=zeros(size(vFft,1),size(vFft,2),nTau+1);

    STCorr(:,:,1)=mean(bsxfun(@times,real(fftshift(fftshift(...
        ifft2(vFft.*conj(vFft)),1),2)),1./(mean(mean(data)).^2)),3);

    for kk=1:nTau
        ind1 = 1:kk:vidsize(3);
        ind2 = ind1(2:end);
        ind1 = ind1(1:end-1);
        STCorr(:,:,kk+1) = mean(bsxfun(@times,real(fftshift(fftshift( ...
            ifft2(vFft(:,:,ind2).*conj(vFft(:,:,ind1))),1),2)), ...
            1./mean(mean(data(:,:,ind1))./mean(mean(data(:,:,ind2)))),3);
    end
    STCorr=STCorr/numel(vFft(:,:,1))-1;
end

% estimate the widths of the correlation function
iMSDs=zeros(nTau,2); MSDd=iMSDs;
[x,y]=ndgrid(1:size(STCorr,1),1:size(STCorr,2));
for ii = 1:nTau
    fitP = gaussFit(STCorr(:,:,ii),'widthGuess',5,'nPixels',...
        min(size(STCorr(:,:,1))),'findTheSpot',0);
    iMSDs(ii,:) = fitP(3:4).^2 * pixelSize^2;

    % discrete variance calculation for x dimension
    pmf=sum(STCorr(:,:,ii)/sum(sum(STCorr(:,:,ii))),2);
    MSDd(ii,1) = sum(pmf.*(x(:,1)-mean(x(:))).^2) * pixelSize^2;

    % discrete variance calculation for x dimension
    pmf=sum(STCorr(:,:,ii)/sum(sum(STCorr(:,:,ii))),1);
    MSDd(ii,2) = sum(pmf.*(y(1,:)-mean(y(:))).^2) * pixelSize^2;
```

```
end

%% MSD Fitting

% choose fitting function
if isConfined
    f=@(p,X)sqconfMSD1D(p,X);
    pStart = [.1, .1, 0];
    lb = [0, 0, -inf];
    ub = [inf, inf, inf];
else
    f=@(p,X) 2*p(2)*X+p(1);
    pStart = [0, .1];
    lb = [-inf, 0];
    ub = [inf, inf];
end

% time lag domain vector
tau = (1:nTau)'*tFrame;

% D from tracking
pT(:,1)=lsqcurvefit(f,pStart,tau,MSDs(:,1),lb,ub);
pT(:,2)=lsqcurvefit(f,pStart,tau,MSDs(:,2),lb,ub);

% D from STICS
pS(:,1)=lsqcurvefit(f,pStart,tau,iMSDs(:,1),lb,ub);
pS(:,2)=lsqcurvefit(f,pStart,tau,iMSDs(:,2),lb,ub);

% D from 'discrete variance'
pD(:,1)=lsqcurvefit(f,pStart,tau,MSDd(:,1),lb,ub);
pD(:,2)=lsqcurvefit(f,pStart,tau,MSDd(:,2),lb,ub);

resStruct.Dtracking = pT(2,:);
resStruct.Dstics = pS(2,:);
resStruct.Dvar = pD(2,:);
resStruct.MSDs = MSDs;
resStruct.iMSDs = iMSDs;
resStruct.MSDd = MSDd;

end

function zi=sqconfMSD1D(p,X)
```

```
l=p(1);
d=exp(p(2));
ns=p(3);
tau=X;

summedTerm=@(t,d,l,n)1/n^4*exp(-(n*pi/l).^2*d*t);

temp=eps*ones(size(tau));
for ii=1:2:2*400-1
    s=summedTerm(tau,d,l,ii);
    if sum(s./temp)<1e-10
        break
    end
    temp=temp+s;
end
zi=l^2/6*(1-96/pi^4*temp)+ns;
zi(isnan(zi))=eps;
zi(isinf(zi))=eps;
end

function imStack=padMask(imStack,mask)
% replace pixels outside the mask with the average value inside the mask in
% each frame

imsize=size(imStack);
mMean=mean(reshape(imStack(mask(:,:,ones(1,imsize(3)))),[],imsize(3)));
imStack(~mask(:,:,ones(1,imsize(3))))=mMean(ones(1,sum(~mask(:))),:);
end

function imstack=padMean(imstack)
% pad the first two dimensions to double size with the mean of each image
imstack=mat2cell(imstack,size(imstack,1),size(imstack,2),ones(1,size(imstack,3)));
imm=cellfun(@(x)mean(mean(x(mask))),imstack,'uniformoutput',false);

imstack=cellfun(@(x,y)padarray(x,floor(size(x)/2),y),imstack,imm,...
    'uniformoutput',false);
imstack=cat(3,imstack{:});
end

function imstack=padZeros(imstack)
% pad the first two dimensions to double size with zeros
imstack=mat2cell(imstack,size(imstack,1),size(imstack,2),ones(1,size(imstack,3)));
```

```
imstack=cellfun(@(x,y)padarray(x,floor(size(x)/2),y),...
    imstack,repmat({0},1,1,size(imstack,3)),'uniformoutput',false);
imstack=cat(3,imstack{:});
end
```

```
function [fitPars, conf95]=gaussFit(img, varargin)
%
% NAME:
%      gaussFit
% PURPOSE:
%      Fits a generalized gaussian function to 2d imaging data. This code
%      produces results in units of pixels for the center position and
%      widths.
% CATEGORY:
%      Image Processing
% CALLING SEQUENCE:
%      [fitPars, conf95] = gaussFit(img,findTheSpot);
% INPUTS:
%      img:           The two-dimensional array to be fit to a gaussian
%
%      varargin:      use paired inputs to set the property (input 1) to the
%          value (input 2) desired.
%
%      Properties:    Descriptions:
%
%      findTheSpot:   1 or 0. Default behavior is to fit an
%          ROI in the center of the image. If the spot is not near the
%          center or the image is very large, findTheSpot enables the code
%          to first roughly locate the spot and then use that location as
%          the ROI center.
%
%      plottingFlag:  1 or 0. show plotting output. default is 0.
%
%      widthGuess:    set the starting value for the width of the
%                     Gaussian in units of pixels.
%
%      nPixels        pixel width of ROI to be selected from img. default
%                     is 11. the value should be odd.
%
% OUTPUTS:
%      fitPars:       fitting coefficient vector, all units are pixels.
%      fitCI:         95% confidence interval of fitting coefficients at
%                     end of fitting
% PROCEDURE:
%      1. Peak guessing and/or data ROI selection of local area inside img
%      2. Non-linear least squares minimization for 7 (or 6) - parameter
```

```
%          Gaussian function on the ROI selected.
% MODIFICATION HISTORY:
%      Written by David J. Rowland, The University of Michigan, 3/16.
% NOTES:
%      This code 'gaussFit.m' should be considered 'freeware'- and may be
%      distributed freely in its original form when properly attributed.
%
%      For testing purposes, run this script:
%
%      img = exp(-x.^2/2/2^2-y.^2/2/3^2)+.02*randn(size(x));
%      p = gaussFit(img,'widthGuess',2);

% if any sim parameters are included as inputs, change the simulation
% parameters mentioned
if nargin>1
    fNames={'findTheSpot', 'plottingFlag', 'widthGuess', 'nPixels'};
    for ii=1:2:nargin-1
        whichField = strcmp(fNames,varargin{ii});

        if all(~whichField)
            warning('Check spelling. Parameter change may have not occurred.')
        end

        eval([fNames{whichField} ' = varargin{ii+1};'])
    end

elseif rem(nargin,1)
    warning('use paired inputs if using varargin.')

    % empty output. size must change if the gaussian fitting function is changed.
    fitPars = nan(1,6);
    conf95 = nan(1,6);
    return
end

%% declaring fitting predicates

if ~exist('findTheSpot','var')
    findTheSpot = 1;
end
if ~exist('plottingFlag','var')
    plottingFlag = 0;
```

```
end
if ~exist('widthGuess','var')
    widthGuess = 2;
end
if ~exist('nPixels','var')
    nPixels = 11;
end

% freely rotating bivariate gaussian function for least squares minimization
% parameters: [xCenter, yCenter, angle, xSD, ySD, amplitude, offset]
% xR=@(x,y,xc,yc,th)(x-xc)*cos(th)-(y-yc)*sin(th);
% yR=@(x,y,xc,yc,th)(x-xc)*sin(th)+(y-yc)*cos(th);
% f=@(p,X) exp( -xR(X(:,1), X(:,2), p(1), p(2), p(3)).^2/2/p(4)^2 + ...
%     -yR( X(:,1), X(:,2), p(1), p(2), p(3)).^2/2/p(5)^2 ) *p(6) + p(7);

% fixed angle fit
% parameters: [xCenter, yCenter, xSD, ySD, amplitude, offset]
th=0;
xR=@(x,y,xc,yc)(x-xc)*cos(th)-(y-yc)*sin(th);
yR=@(x,y,xc,yc)(x-xc)*sin(th)+(y-yc)*cos(th);
f=@(p,X) exp( -xR(X(:,1), X(:,2), p(1), p(2)).^2/2/p(3)^2 + ...
    -yR( X(:,1), X(:,2), p(1), p(2)).^2/2/p(4)^2 ) *p(5) + p(6);

% bounds
lb=[-inf, -inf, 0, 0, -inf, -inf];
ub=[inf, inf, inf, inf, inf, inf];


%% data selection

if findTheSpot
    % select the local area around a bright spot in a larger image

    % bandpass and threshold
    LP=1;          % low pass value
    HP=10;         % high pass value
    intThresh=0.1; % intensity threshold. set to zero and then check by inspection
    hMax=0.1;      % larger if the dynamic range of your data is larger
    lzero=4;       % this squelches a 5 pixel boundary around the filtered image

    bIm=bpassDJR(img, LP, HP, intThresh, lzero);
```

```
    % watershed algorithm
    extImg=imextendedmax(bIm,hMax);

    % failed watershed algorithm can result in all ones
    if all(extImg(:))
        extImg=extImg-1;
    end

    % shrink to a point. this is the estimated location of the spot
    sIm=bwmorph(extImg,'shrink',inf);

    % the index of the one pixel is a good guess for the particle location
    [locInds(:,1),locInds(:,2)]=find(sIm);

    % temporally coincdident guesses are not treated with this code.
    if size(locInds(:,1))~=1
        fitPars=nan(1,6);
        conf95=nan(1,6);
        return
    end

else
    % otherwise, assume the spot is in near the center of the image
    locInds=round(size(img)/2);
end

% pad the img(s) with nans (removed later).
padsize=[nPixels,nPixels];
padVal=nan;
direction='both';
img=padarray(img,padsize,padVal,direction);
locInds=locInds+nPixels;

% find the selection domain
[sDom1,sDom2]=ndgrid(locInds(1)-(nPixels-1)/2:locInds(1)+(nPixels-1)/2, ...
    locInds(2)-(nPixels-1)/2:locInds(2)+(nPixels-1)/2);
inds=sub2ind(size(img),sDom1(:),sDom2(:));

% select the data
truImg=reshape(img(inds),[nPixels,nPixels]);

%% starting parameter selection for 6-parameter Gaussian Fit
```

```
% x, y centers starting guess
pStart(1)=0;
pStart(2)=0;

% xSD, ySD in units of pixels
pStart(3)=widthGuess;
pStart(4)=widthGuess;

% amplitude, offset
mVals=[max(truImg(:)),min(truImg(:))];
pStart(5)=mVals(1)-mVals(2);
pStart(6)=mVals(2);

%% fitting the data
[x,y]=ndgrid(1:nPixels,1:nPixels);
X=cat(2,x(:),y(:)) - nPixels/2;
[fitPars, ~, residual, ~, ~, ~,jacobian] = ...
    lsqcurvefit(f,pStart,X(~isnan(truImg(:)),:),truImg(~isnan(truImg(:))),...
    lb,ub);

% confidence intervals
conf95 = nlparci(fitPars, residual,'jacobian',jacobian);

%% plot the output
if plottingFlag
    fVals=reshape(f(fitPars,X),[nPixels,nPixels]);
    dVals=truImg;
    sVals=reshape(f(pStart,X),[nPixels,nPixels]);

    subplot(221)
    title('Data')
    pcolor(kron(dVals,ones(10)))
    shading flat; axis image; colorbar

    subplot(222)
    title('starting values')
    pcolor(kron(sVals,ones(10)))
    shading flat; axis image; colorbar

    subplot(223)
    title('fit result')
```

```
    pcolor(kron(fVals,ones(10)))
    shading flat; axis image; colorbar

    subplot(224)
    title('residuals')
    pcolor(kron(dVals - fVals,ones(10)))
    shading flat; axis image; colorbar
end

% shift center back to lab frame
fitPars([1,2])=fitPars(1:2)+locInds-nPixels;
end
```

APPENDIX B
Code for Chapter 4: simpleDiffusion, cpdFunFinder, cpdGlobal

For the most up-to-date version of the following code, please visit

https://github.com/BiteenLab/SingleMoleculeDataAnalysis

```
function tr =
    simpleDiffusion(diffusionCoefficient,integrationTime,numberOfFrames,micronsPerPixel)
% diffusionCoefficient in microns per second^2
% integration time in seconds

tr = cumsum(cat(1,[0,0],sqrt(2*diffusionCoefficient*integrationTime)*...
    randn(numberOfFrames-1,2)),1);

tr(:,4:5) = tr(:,1:2)/micronsPerPixel;
tr(:,1) = 1;
tr(:,2) = 1:numberOfFrames;
tr(:,3) = nan;
end
```

```
function outStruct = cpdFunFinder(nMobile)
%
% NAME:
%      cpdFunFinder
% PURPOSE:
%      Based on the number of diffusive terms, determine the fitting
%      function structure required for use in the CPDGlobal. code.
% CATEGORY:
%      Data fitting
% CALLING SEQUENCE:
%      outStruct = cpdFunFinder(nMobile);
% INPUTS:
%      nMobile:       Integer number of expected diffusive components
%
% OUTPUTS:
%      outStruct:     Matlab structure used inside CPDGlobal to construct
%                     the required fitting function.

% MODIFICATION HISTORY:
%      Written by David J. Rowland, The University of Michigan, 11/16.
% NOTES:
%      This code 'cpdFunFinder.m' should be considered 'freeware'- and may be
%      distributed freely in its original form when properly attributed.


% partial 2d cpd function
c2=@(x,y,p)p*exp(-x./y);

% 2d confined msd function
m2=@(t,p)4*p(1)*t+p(2);

% starting values
pStart = [.9,0,.1, .01, .0025, .00001, .2, .2, .2, .2];

% bounds for the fit
LB=-inf(1,numel(pStart));
LB(7:10) = 0;
UB=inf(1,numel(pStart));
UB(7:10) = 1;

switch nMobile
```

```
case 1
    msdFun=@(tau,p) ...
        m2(tau,p([1,2]));
    cpdFun=@(x,y,p)1-...
        c2(x,y(1),1);
    pID = 1:2;

case 2
    msdFun=@(tau,p)cat(2,...
        m2(tau,p([1,2])),...
        m2(tau,p([3,2])));
    cpdFun=@(x,y,p)1-...
        c2(x,y(1),p(4))-...
        c2(x,y(2),1-p(4));
    pID = [1:3,7];

case 3
    msdFun=@(tau,p)cat(2,...
        m2(tau,p([1,2])),...
        m2(tau,p([3,2])),...
        m2(tau,p([4,2])));
    cpdFun=@(x,y,p)1-...
        c2(x,y(1),p(5))-...
        c2(x,y(2),p(6))-...
        c2(x,y(3),1-p(5)-p(6));
    pID = [1:4,7:8];

case 4
    msdFun=@(tau,p)cat(2,...
        m2(tau,p([1,2])),...
        m2(tau,p([3,2])),...
        m2(tau,p([4,2])),...
        m2(tau,p([5,2])));
    cpdFun=@(x,y,p)1-...
        c2(x,y(1),p(6))-...
        c2(x,y(2),p(7))-...
        c2(x,y(3),p(8))-...
        c2(x,y(4),1-p(6)-p(7)-p(8));
    pID = [1:5,7:9];

case 5
    msdFun=@(tau,p)cat(2,...
```

```
            m2(tau,p([1,2])),...
            m2(tau,p([3,2])),...
            m2(tau,p([4,2])),...
            m2(tau,p([5,2])),...
            m2(tau,p([6,2])));
        cpdFun=@(x,y,p)1-...
            c2(x,y(1),p(7))-...
            c2(x,y(2),p(8))-...
            c2(x,y(3),p(9))-...
            c2(x,y(4),p(10))-...
            c2(x,y(5),1-p(7)-p(8)-p(9)-p(10));
        pID = [1:10];
end

pStart={pStart(pID)};
bounds=[{LB(pID)},{UB(pID)}];
dID = find(ismember(pID,[1,3:6]));
aID = find(ismember(pID,7:10));

outStruct.cpdFun = cpdFun;
outStruct.msdFun = msdFun;
outStruct.pStart = pStart;
outStruct.bounds = bounds;
outStruct.dID = dID;
outStruct.aID = aID;
end
```

```
function out = CPDGlobal(tr)
%
% NAME:
%      CPDGlobal
% PURPOSE:
%      Given trajectories of diffusing molecules fit all Cumulative
%      probability distributions (CPDs) of step sizes given a particular model of
%      diffusion all at once. This is a multi-domain fit, where several of
%      the  domains are the squared step size domains of the CPDs, and the
%      other is the time lag domain typically used in MSD fits to the
%      model of diffusion.
% CATEGORY:
%      Data fitting
% CALLING SEQUENCE:
%      outStruct = CPDGlobal(tr);
% INPUTS:
%      tr:            is a cell array with each element being n X 4,
%                     where n is the number of frames. The first column
%                     is the trajectory id number (integers), the second
%                     column is the time step id (integers), and the 3rd
%                     and 4th columns are the x and y positions of the
%                     trajectory
%
% OUTPUTS:
%      out:           Matlab structure used inside CPDGlobal to construct
%                     the required fitting function.
% USAGE:
%      Configure the analysis parameters below to correspond to your
%      particular experimental setup.
% MODIFICATION HISTORY:
%      Written by David J. Rowland, The University of Michigan, 11/16.
% NOTES:
%      This code 'CPDGlobal.m' should be considered 'freeware'- and may be
%      distributed freely in its original form when properly attributed.

%% Default analysis parameters
anProp.nMobile = 1;   % number of diffusive populations
anProp.tFrame = .0104; % camera integration time in seconds
anProp.pixSize = .049; % pixel size in microns
anProp.minTau = 1;    % minimum time lag in frames
anProp.maxTau = 5;    % maximum time lag in frames
```

```
anProp.overBool = 0;  % use overlapping or non-overlapping displacements?
anProp.bootNum = 300; % number of bootstraps

% partial 2d cpd function
c2=@(x,y,p)p*exp(-x./y);

% 2d confined msd function
m2=@(t,p)4*p(1)*t+p(2);

% linearized cell arrays of similar dimension
linCell=@(x)cat(1,x{:});

% disable fitting routine textual output
opts = optimset('Display','off');

%% calculate and collect all the squared step sizes for each time lag considered
for kk=1:numel(tr)
    if isempty(tr{kk})
        continue
    end
    trackNums = unique(tr{kk}(:,1))';

    if ~isempty(trackNums)
        for ii = trackNums
            tracks = tr{kk}(tr{kk}(:,1)==ii,[2,3,4]);

            % fill in the time holes with nans
            fixedTrack = nan(max(tracks(:,1)),size(tracks,2));
            fixedTrack(tracks(:,1),:) = tracks;

            % remove leading nans
            fixedTrack(1:find(all(isnan(fixedTrack),2)==0,1,'first')-1,:) = [];

            nLocs = size(fixedTrack,1);
            for jj=1:anProp.maxTau
                if anProp.overBool    % overlapping displacements
                    indvec1=jj+1:nLocs;
                    indvec2=1:nLocs-jj;
                elseif ~anProp.overBool % non-overlapping displacements
                    indvec2=1:jj:nLocs;
                    indvec1=indvec2(1:end-1);
                    indvec2=indvec2(2:end);
```

```
                end

                % calculate squared step sizes
                allSqSteps{kk,ii,jj}=nansum( (fixedTrack(indvec1,[2,3]) - ...
                    fixedTrack(indvec2,[2,3])).^2, 2);
            end
        end
    end
end

%% compile the cumulative probability distributions for each time lag
sqSteps=cell(anProp.maxTau,1);
for ii=1:anProp.maxTau
    wSteps = cat(1,allSqSteps{:,:,ii});
    sqSteps{ii}=sort(wSteps(wSteps > eps)); % nansum puts zeros where there were
        nans
end
oRanks=cellfun(@(x)linspace(0,1,numel(x))',sqSteps,'uniformoutput',0);
sqSteps = cellfun(@(x)x*anProp.pixSize.^2,sqSteps,'uniformoutput',0);
nSteps = cellfun(@numel,sqSteps,'uniformoutput',0);

%% fitting function selection
funFinds = cpdFunFinder(anProp.nMobile);
cpdFun = funFinds.cpdFun;
msdFun = funFinds.msdFun;
pStart = funFinds.pStart;
bounds = funFinds.bounds;
dID = funFinds.dID;
aID = funFinds.aID;

% uninformed amplitude guesses
pStart{1}(aID) = 1/(numel(aID)+1);

%% GLOBAL FITTING
fHandle=@(p,tau,sqSteps,ranks)linCell(...
    cellfun(@(x,y)x-y,...
    cellfun(@(x,y)cpdFun(x,y,p),...
    sqSteps,num2cell(msdFun(tau,p),2),'uniformoutput',0),...
    ranks,'uniformoutput',0));
eHandle=@(p,tau,sqSteps)cellfun(@(x,y)cpdFun(x,y,p),...
    sqSteps,num2cell(msdFun(tau,p),2),'uniformoutput',0);
```

```
% fit to original data
y=sqSteps;
r=oRanks;

% time lag domain in seconds
tau = (1:anProp.maxTau)'*anProp.tFrame;

% fitting to the original data
[fP_nB,~,r_nB] = lsqnonlin(@(p)fHandle(...
    p,tau(anProp.minTau:end),y(anProp.minTau:end),r(anProp.minTau:end)),...
    pStart{1},bounds{1},bounds{2},opts);

% split up the residuals into into 1 for each cpd curve
rTemp = r_nB;
for ii = anProp.minTau:anProp.maxTau
    residCell{ii} = rTemp(1:nSteps{ii});
    rTemp(1:nSteps{ii}) = [];
end

% bootstrap
fP_B = zeros(anProp.bootNum,numel(pStart{1}));
parfor kk = 1:anProp.bootNum
    % resamples with replacement
    y1 =
        cellfun(@(x,y)sort(x(randsample(y,y,1))),sqSteps,nSteps,'uniformoutput',0);
    r1 = oRanks;

    % fitting to the bootstrapped data
    fP_B(kk,:) = lsqnonlin(@(p)fHandle(...
        p,tau(anProp.minTau:end),y1(anProp.minTau:end),r1(anProp.minTau:end)),...
        pStart{1},bounds{1},bounds{2},opts);
end

%% plot results
c = colormap('lines');
c = c(1:7,:);

if anProp.nMobile > 1
    nPlots = 3;
else
    nPlots = 2;
end
```

```
subplot(nPlots,1,1)
for ii = 1:numel(dID)
    h=histogram(fP_B(:,dID(ii)),'normalization','probability','displaystyle','stairs');
    set(h,'edgecolor',c(ii,:))
    hold on
end
title('Bootstrapped diffusion coefficients')
set(gca,'xscale','log'); hold off

subplot(nPlots,1,2)
for ii = anProp.minTau:anProp.maxTau
    plot(sqSteps{ii},residCell{ii}); hold all
end
title('Original data CPD residuals');
hold off

if anProp.nMobile > 1
    subplot(nPlots,1,3)
    lastAmp = ones(anProp.bootNum,1);
    for ii = 1:numel(aID)
        h=histogram(fP_B(:,aID(ii)),'normalization','probability','displaystyle','stairs');
        set(h,'edgecolor',c(ii,:))
        hold on

        lastAmp = lastAmp-fP_B(:,aID(ii));
    end
    h=histogram(lastAmp,'normalization','probability','displaystyle','stairs');
    title('Bootstrapped population amplitudes')
    hold off
end

% output files
out.fittedParameters_nB = fP_nB;
out.cpdResiduals = residCell;
out.sqSteps = sqSteps;
out.dID = dID;
out.aID = aID;
end
```

Bibliography

[1] L. J. F. Jones, R. Carballido-López, and J. Errington, "Control of cell shape in bacteria: Helical, actin-like filaments in Bacillus subtilis," *Cell*, vol. 104, no. 6, pp. 913–922, 2001.

[2] Z. Gitai, "The new bacterial cell biology: Moving parts and subcellular architecture," *Cell*, vol. 120, no. 5, pp. 577–586, 2005.

[3] Y.-L. Shih and L. Rothfield, "The Bacterial Cytoskeleton," *Microbiology and Molecular Biology Reviews*, vol. 70, no. 3, pp. 729–754, 2006.

[4] J. Lefman, P. Zhang, T. Hirai, M. Robert, J. Juliani, D. Bliss, M. Kessel, E. Bos, P. J. Peters, S. Subramaniam, and R. M. Weis, "Three-Dimensional Electron Microscopic Imaging of Membrane Invaginations in Escherichia coli Overproducing the Chemotaxis Receptor Tsr," *Journal of Bacteriology*, vol. 186, no. 15, pp. 5052–5061, 2004.

[5] W. Baumeister, R. Grimm, and J. Walz, "Electron tomography of molecules and cells," *Trends in Cell Biology*, vol. 9, no. 2, pp. 81–85, 1999.

[6] G. Patterson, M. Davidson, S. Manley, and J. Lippincott-Schwartz, "Superresolution Imaging using Single-Molecule Localization," *Annual Review of Physical Chemistry*, vol. 61, no. 1, pp. 345–367, 2010.

[7] C. Joo, H. Balci, Y. Ishitsuka, C. Buranachai, and T. Ha, "Advances in single-molecule fluorescence methods for molecular biology.," *Annual review of biochemistry*, vol. 77, pp. 51–76, 2008.

[8] B. L. Haas, J. S. Matson, V. J. Dirita, and J. S. Biteen, "Single-molecule tracking in live Vibrio cholerae reveals that ToxR recruits the membrane-bound virulence regulator TcpP to the toxT promoter," *Molecular Microbiology*, vol. 96, no. 1, pp. 4–13, 2015.

[9] H. H. Tuson and J. S. Biteen, "Unveiling the inner workings of live bacteria using super-resolution microscopy," *Analytical Chemistry*, vol. 87, no. 1, pp. 42–63, 2015.

[10] E. Abbe, "Beiträge zur Theorie des Mikroskops und der mikroskopischen Wahrnehmung: I. Die Construction von Mikroskopen auf Grund der Theorie," *Archiv für mikroskopische Anatomie*, vol. 9, no. 1, pp. 413–418, 1873.

[11] E. Betzig, G. H. Patterson, R. Sougrat, O. W. Lindwasser, S. Olenych, J. S. Bonifacino, M. W. Davidson, J. Lippincott-Schwartz, and H. F. Hess, "Imaging intracellular fluorescent proteins at nanometer resolution.," *Science (New York, N.Y.)*, vol. 313, no. 5793, pp. 1642–5, 2006.

[12] S. T. Hess, T. P. Girirajan, and M. D. Mason, "Ultra-High Resolution Imaging by Fluorescence Photoactivation Localization Microscopy," *Biophysical Journal*, vol. 91, no. 11, pp. 4258–4272, 2006.

[13] M. J. Rust, M. Bates, and X. W. Zhuang, "Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM)," *Nature Methods*, vol. 3, no. 10, pp. 793–795, 2006.

[14] M. Heilemann, S. Van De Linde, M. Schüttpelz, R. Kasper, B. Seefeldt, A. Mukherjee, P. Tinnefeld, and M. Sauer, "Subdiffraction-resolution fluorescence imaging with conventional fluorescent probes," *Angewandte Chemie - International Edition*, vol. 47, no. 33, pp. 6172–6176, 2008.

[15] R. E. Thompson, D. R. Larson, and W. W. Webb, "Precise nanometer localization analysis for individual fluorescent probes.," *Biophysical Journal*, vol. 82, no. 5, pp. 2775–83, 2002.

[16] A. J. Berglund, M. D. McMahon, J. J. McClelland, and J. A. Liddle, "Fast, bias-free algorithm for tracking single particles with variable size and shape.," *Optics Express*, vol. 16, no. 18, pp. 14064–14075, 2008.

[17] R. Starr, S. Stahlheber, and A. Small, "Fast maximum likelihood algorithm for localization of fluorescent molecules.," *Optics Letters*, vol. 37, no. 3, pp. 413–5, 2012.

[18] R. J. Ober, S. Ram, and E. S. Ward, "Localization accuracy in single-molecule microscopy.," *Biophysical Journal*, vol. 86, no. 2, pp. 1185–200, 2004.

[19] R. H. Webb, "Confocal optical microscopy," *Reports on Progress in Physics*, vol. 59, no. 3, pp. 427–471, 1999.

[20] P. A. Santi, "Light Sheet Fluorescence Microscopy A Review," *Journal of Histochemistry & Cytochemistry*, vol. 59, no. 2, pp. 129–138, 2011.

[21] S. Hell and E. H. K. Stelzer, "Fundamental improvement of resolution with a 4Pi-confocal fluorescence microscope using two-photon excitation," *Optics Communications*, vol. 93, no. 5-6, pp. 277–282, 1992.

[22] B. Huang, W. Wang, M. Bates, and X. Zhuang, "Three-Dimensional Super-Resolution Imaging by Stochastic Optical Reconstruction Microscopy," *Science*, vol. 319, pp. 810–813, feb 2008.

[23] H. P. Kao and A. S. Verkman, "Tracking of Single Fluorescent Particles in Three Dimensions: Use of Cylindrical Optics to Encode Particle Position," *Biophysical Journal*, vol. 67, no. 3, pp. 1291–1300, 1994.

[24] S. R. P. Pavani, M. A. Thompson, J. S. Biteen, S. J. Lord, N. Liu, R. J. Twieg, R. Piestun, and W. E. Moerner, "Three-dimensional, single-molecule fluorescence imaging beyond the diffraction limit by using a double-helix point spread function.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 9, pp. 2995–2999, 2009.

[25] J. Schuster, F. Cichos, and C. Von Borczyskowski, "Diffusion measurements by single-molecule spot-size analysis," *Journal of Physical Chemistry A*, vol. 106, no. 22, pp. 5403–5406, 2002.

[26] S. K. Zareh, M. C. Desantis, J. M. Kessler, J. L. Li, and Y. M. Wang, "Single-image diffusion coefficient measurements of proteins in free solution," *Biophysical Journal*, vol. 102, no. 7, pp. 1685–1691, 2012.

[27] T. Fazio, M. L. Visnapuu, S. Wind, and E. C. Greene, "DNA curtains and nanoscale curtain rods: High-throughput tools for single molecule imaging," *Langmuir*, vol. 24, no. 18, pp. 10524–10531, 2008.

[28] Y. Liao, S. K. Yang, K. Koh, A. J. Matzger, and J. S. Biteen, "Heterogeneous single-molecule diffusion in one-, two-, and three-dimensional microporous coordination polymers: Directional, trapped, and immobile guests," *Nano Letters*, vol. 12, no. 6, pp. 3080–3085, 2012.

[29] D. J. Rowland and J. S. Biteen, "Top-hat and asymmetric Gaussian-based fitting functions for quantifying directional single-molecule motion," *ChemPhysChem*, vol. 15, no. 4, pp. 712–720, 2014.

[30] C. Di Rienzo, E. Gratton, F. Beltram, and F. Cardarelli, "Fast spatiotemporal correlation spectroscopy to determine protein lateral diffusion laws in live cell membranes," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 15, pp. 12307–12312, 2013.

[31] P. Schwille and R. Rigler, "Dual-Color Fluorescence Cross-Correlation Spectroscopy for Multicomponent Diffusional Analysis in Solution," *Biophysical Journal*, vol. 72, no. 4, pp. 1878–1886, 1997.

[32] D. L. Kolin, S. Costantino, and P. W. Wiseman, "Sampling effects, noise, and photobleaching in temporal image correlation spectroscopy.," *Biophysical Journal*, vol. 90, no. 2, pp. 628–39, 2006.

[33] M. A. Digman, P. Sengupta, P. W. Wiseman, C. M. Brown, A. R. Horwitz, and E. Gratton, "Fluctuation correlation spectroscopy with a laser-scanning microscope: exploiting the hidden time structure.," *Biophysical Journal*, vol. 88, no. 5, pp. L33–6, 2005.

[34] D. J. Rowland, H. H. Tuson, and J. S. Biteen, "Resolving Fast, Confined Diffusion in Bacteria with Image Correlation Spectroscopy," *Biophysical Journal*, vol. 110, no. 10, pp. 2241–2251, 2016.

[35] P. R. Selvin and T. Ha, *Single-Molecule Techniques*. Cold Spring Harbor, New York: Cold Spring Harbor Laboratory Press, 2008.

[36] O. Rossier, V. Octeau, J.-B. Sibarita, C. Leduc, B. Tessier, D. Nair, V. Gatterdam, O. Destaing, C. Albigès-Rizo, R. Tampé, L. Cognet, D. Choquet, B. Lounis, and G. Giannone, "Integrins $\beta 1$ and $\beta 3$ exhibit distinct dynamic nanoscale organizations inside focal adhesions," *Nature Cell Biology*, vol. 14, no. 10, pp. 1057–1067, 2012.

[37] C. K. Johnson and G. S. Harms, "Tracking and localization of calmodulin in live cells," *Biochimica et Biophysica Acta - Molecular Cell Research*, vol. 1863, no. 8, pp. 2017–2026, 2016.

[38] S. Manley, J. M. Gillette, G. H. Patterson, H. Shroff, H. F. Hess, E. Betzig, and J. Lippincott-Schwartz, "High-density mapping of single-molecule trajectories with photoactivated localization microscopy.," *Nature Methods*, vol. 5, no. 2, pp. 155–157, 2008.

[39] C. M. Anderson, G. N. Georgiou, I. E. Morrison, G. V. Stevenson, and R. J. Cherry, "Tracking of cell surface receptors by fluorescence digital imaging microscopy using a charge-coupled device camera. Low-density lipoprotein and influenza virus receptor mobility at 4 degrees C.," *Journal of Cell Science*, vol. 101, no. 2, pp. 415–425, 1992.

[40] M. J. Saxton and K. Jacobson, "Single-Particle Tracking: Applications to Membrane Dynamics," *Annual Review of Biophysics and Biomolecular Structure*, vol. 26, no. 1, pp. 373–399, 1997.

[41] M. A. Deverall, E. Gindl, E. K. Sinner, H. Besir, J. Ruehe, M. J. Saxton, and C. A. Naumann, "Membrane lateral mobility obstructed by polymer-tethered lipids studied at the single molecule level.," *Biophysical Journal*, vol. 88, no. 3, pp. 1875–86, 2005.

[42] J. Gelles, B. J. Schnapp, and M. P. Sheetz, "Tracking kinesin-driven movements with nanometre-scale precision.," *Nature*, vol. 331, no. 6155, pp. 450–453, 1988.

[43] M. K. Cheezum, W. F. Walker, and W. H. Guilford, "Quantitative comparison of algorithms for tracking single fluorescent particles.," *Biophysical Journal*, vol. 81, no. 4, pp. 2378–2388, 2001.

[44] A. Yildiz, J. N. Forkey, S. A. McKinney, T. Ha, Y. E. Goldman, and P. R. Selvin, "Myosin V walks hand-over-hand: Single fluorophore imaging with 1.5-nm localization," *Science*, vol. 300, no. 5628, pp. 2061–2065, 2003.

[45] S. Y. Kim, Z. Gitai, A. Kinkhabwala, L. Shapiro, and W. E. Moerner, "Single molecules of the bacterial actin MreB undergo directed treadmilling motion in Caulobacter crescentus.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 29, pp. 10929–34, 2006.

[46] S. Semrau, A. Pezzarossa, and T. Schmidt, "Microsecond single-molecule tracking (usSMT)," *Biophysical Journal*, vol. 100, no. 4, pp. L19–21, 2011.

[47] L. Xiao, L. Wei, C. Liu, Y. He, and E. S. Yeung, "Unsynchronized translational and rotational diffusion of nanocargo on a living cell membrane," *Angewandte Chemie - International Edition*, vol. 51, no. 17, pp. 4181–4184, 2012.

[48] R. N. Ghosh and W. W. Webb, "Automated detection and tracking of individual and clustered cell surface low density lipoprotein receptor molecules.," *Biophysical Journal*, vol. 66, no. 5, pp. 1301–1318, 1994.

[49] K. I. Mortensen, L. S. Churchman, J. a. Spudich, and H. Flyvbjerg, "Optimized localization analysis for single-molecule tracking and super-resolution microscopy.," *Nature Methods*, vol. 7, no. 5, pp. 377–381, 2010.

[50] J. Engelhardt, J. Keller, P. Hoyer, M. Reuss, T. Staudt, and S. W. Hell, "Molecular orientation affects localization accuracy in superresolution far-field fluorescence microscopy," 2011.

[51] S. M. Kay, *Fundamentals of Statistical Signal Processing*, vol. 1. Upper Saddle River, NJ: Prentice Hall PTR, 2015.

[52] C. S. Smith, N. Joseph, B. Rieger, and K. A. Lidke, "Fast, single-molecule localization that achieves theoretically minimum uncertainty.," *Nature Methods*, vol. 7, no. 5, pp. 373–5, 2010.

[53] F. Aguet, D. Van De Ville, and M. Unser, "A maximum-likelihood formalism for sub-resolution axial localization of fluorescent nanoparticles.," *Optics Express*, vol. 13, no. 26, pp. 10503–10522, 2005.

[54] M. C. DeSantis, S. K. Zareh, X. Li, R. E. Blankenship, and Y. M. Wang, "Single-image axial localization precision analysis for individual fluorophores," *Optics Express*, vol. 20, no. 3, p. 3057, 2012.

[55] M. Y. Yüce, A. Jonás, and A. T. Erdoan, "Video-based tracking of single molecules exhibiting directed in-frame motion.," *Microscopy and microanalysis : the official journal of Microscopy Society of America, Microbeam Analysis Society, Microscopical Society of Canada*, vol. 18, no. 4, pp. 781–92, 2012.

[56] P. A. Chiarelli, M. S. Johal, D. J. Holmes, J. L. Casson, J. M. Robinson, and H. L. Wang, "Polyelectrolyte spin-assembly," *Langmuir*, vol. 18, no. 1, pp. 168–173, 2002.

[57] K. Nienhaus and G. U. Nienhaus, "Fluorescent proteins for live-cell imaging with super-resolution," *Chemical Society Reviews*, vol. 43, no. 4, pp. 1088–1106, 2014.

[58] S. Uphoff, R. Reyes-Lamothe, F. Garza de Leon, D. J. Sherratt, and A. N. Kapanidis, "Single-molecule DNA repair in live bacteria.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 20, pp. 8063–8068, 2013.

[59] P. Hammar, M. Walldén, D. Fange, F. Persson, O. Baltekin, G. Ullman, P. Leroy, and J. Elf, "Direct measurement of transcription factor dissociation excludes a simple operator occupancy model for gene regulation.," *Nature Genetics*, vol. 46, no. 4, pp. 405–8, 2014.

[60] A. Sanamrad, F. Persson, E. G. Lundius, D. Fange, A. H. Gynnå, and J. Elf, "Single-particle tracking reveals that free ribosomal subunits are not excluded from the Escherichia coli nucleoid.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 111, no. 31, pp. 11413–8, 2014.

[61] K. S. Karunatilaka, E. A. Cameron, E. C. Martens, N. M. Koropatkin, and S. Biteen, "in Human Gut Symbionts," *mBio*, vol. 5, no. 6, pp. 1–10, 2014.

[62] D. Axelrod, D. E. Koppel, J. Schlessinger, E. Elson, and W. W. Webb, "Mobility measurement by analysis of fluorescence photobleaching recovery kinetics," *Biophysical Journal*, vol. 16, no. 9, pp. 1055–1069, 1976.

[63] K. A. Sochacki, I. A. Shkel, M. T. Record, and J. C. Weisshaar, "Protein diffusion in the periplasm of E. coli under osmotic stress," *Biophysical Journal*, vol. 100, no. 1, pp. 22–31, 2011.

[64] M. B. Elowitz, M. G. Surette, P. E. Wolf, J. B. Stock, and S. Leibler, "Protein mobility in the cytoplasm of Escherichia coli," *Journal of Bacteriology*, vol. 181, no. 1, pp. 197–203, 1999.

[65] A. Kusumi, Y. Sako, and M. Yamamoto, "Confined lateral diffusion of membrane receptors as studied by single particle tracking (nanovid microscopy). Effects of calcium-induced differentiation in cultured epithelial cells.," *Biophysical Journal*, vol. 65, no. 5, pp. 2021–40, 1993.

[66] Y. Lill, W. A. Kaserer, S. M. Newton, M. Lill, P. E. Klebba, and K. Ritchie, "Single-molecule study of molecular mobility in the cytoplasm of Escherichia coli," *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, vol. 86, no. 2, pp. 1–7, 2012.

[67] M. J. Saxton, "Single-particle tracking: the distribution of diffusion coefficients.," *Biophysical Journal*, vol. 72, no. 4, pp. 1744–53, 1997.

[68] J. E. Donehue, E. Wertz, C. N. Talicska, and J. S. Biteen, "Plasmon-Enhanced brightness and photostability from single fluorescent proteins coupled to gold nanorods," *Journal of Physical Chemistry C*, vol. 118, no. 27, pp. 15027–15035, 2014.

[69] J. Elf, G.-W. Li, and X. S. Xie, "Probing transcription factor dynamics at the single-molecule level in a living cell.," *Science (New York, N.Y.)*, vol. 316, no. 5828, pp. 1191–4, 2007.

[70] B. Hebert, S. Costantino, and P. W. Wiseman, "Spatiotemporal image correlation spectroscopy (STICS) theory, verification, and application to protein velocity mapping in living CHO cells.," *Biophysical Journal*, vol. 88, no. 5, pp. 3601–3614, 2005.

[71] N. O. Petersen, P. L. Höddelius, P. W. Wiseman, O. Seger, and K. E. Magnusson, "Quantitation of membrane receptor distributions by image correlation spectroscopy: concept and application.," *Biophysical Journal*, vol. 65, no. 3, pp. 1135–46, 1993.

[72] D. Pozzi, C. Marchini, F. Cardarelli, A. Rossetta, V. Colapicchioni, A. Amici, M. Montani, S. Motta, P. Brocca, L. Cantù, and G. Caracciolo, "Mechanistic understanding of gene delivery mediated by highly efficient multicomponent envelope-type nanoparticle systems," *Molecular Pharmaceutics*, vol. 10, no. 12, pp. 4654–4665, 2013.

[73] C. M. Brown, B. Hebert, D. L. Kolin, J. Zareno, L. Whitmore, A. R. Horwitz, and P. W. Wiseman, "Probing the integrin-actin linkage using high-resolution protein velocity mapping.," *Journal of Cell Science*, vol. 119, no. Pt 24, pp. 5204–5214, 2006.

[74] A. Abu-Arish, E. Pandzic, J. Goepp, E. Matthes, J. Hanrahan, and P. Wiseman, "Cholesterol Modulates CFTR Confinement in the Plasma Membrane of Primary Epithelial Cells," *Biophysical Journal*, vol. 109, no. 1, pp. 85–94, 2015.

[75] L. Kisley, R. Brunetti, L. J. Tauzin, B. Shuang, X. Yi, A. W. Kirkeminde, D. A. Higgins, S. Weiss, and C. F. Landes, "Characterization of Porous Materials by Fluorescence Correlation Spectroscopy Super-resolution Optical Fluctuation Imaging," *ACS Nano*, vol. 9, no. 9, pp. 9158–9166, 2015.

[76] J. Hendrix, V. Baumgärtel, W. Schrimpf, S. Ivanchenko, M. A. Digman, E. Gratton, H. G. Kräusslich, B. Müller, and D. C. Lamb, "Live-cell observation of cytosolic HIV-1 assembly onset reveals RNA-interacting Gag oligomers," *Journal of Cell Biology*, vol. 210, no. 4, pp. 629–646, 2015.

[77] M. a. Digman, C. M. Brown, P. Sengupta, P. W. Wiseman, A. R. Horwitz, and E. Gratton, "Measuring fast dynamics in solutions and cells with a laser scanning microscope.," *Biophysical Journal*, vol. 89, no. 2, pp. 1317–1327, 2005.

[78] S. Semrau and T. Schmidt, "Particle image correlation spectroscopy (PICS): retrieving nanometer-scale correlations from high-density single-molecule position data.," *Biophysical Journal*, vol. 92, no. 2, pp. 613–621, 2007.

[79] S. Bakshi, B. P. Bratton, and J. C. Weisshaar, "Subdiffraction-limit study of Kaede diffusion and spatial distribution in live Escherichia coli," *Biophysical Journal*, vol. 101, no. 10, pp. 2535–2544, 2011.

[80] S. Deville, R. Penjweini, N. Smisdom, K. Notelaers, I. Nelissen, J. Hooyberghs, and M. Ameloot, "Intracellular dynamics and fate of polystyrene nanoparticles in A549 Lung epithelial cells monitored by image (cross-) correlation spectroscopy and single particle tracking," *Biochimica et Biophysica Acta - Molecular Cell Research*, vol. 1853, no. 10, pp. 2411–2419, 2015.

[81] C. L. Chiu, J. S. Aguilar, C. Y. Tsai, G. K. Wu, E. Gratton, and M. A. Digman, "Nanoimaging of focal adhesion dynamics in 3D," *PLoS ONE*, vol. 9, no. 6, 2014.

[82] B. Storti, C. Di Rienzo, F. Cardarelli, R. Bizzarri, and F. Beltram, "Unveiling TRPV1 spatio-temporal organization in live cell membranes.," *PLoS ONE*, vol. 10, no. 3, pp. 1–17, 2015.

[83] S. Meucci, M. Travagliati, O. Vittorio, G. Cirillo, L. Masini, V. Voliani, N. Picci, F. Beltram, A. Tredicucci, and M. Cecchini, "Tubeless biochip for chemical stimulation of cells in closed-bioreactors: anti-cancer activity of the catechindextran conjugate," *RSC Advances*, vol. 4, no. 66, pp. 35017–35026, 2014.

[84] W. E. Moerner and D. P. Fromm, "Methods of single-molecule fluorescence spectroscopy and microscopy," *Review of Scientific Instruments*, vol. 74, no. 8, pp. 3597–3619, 2003.

[85] B. L. Haas, J. S. Matson, V. J. DiRita, and J. S. Biteen, "Imaging live cells at the nanometer-scale with single-molecule microscopy: Obstacles and achievements in experiment optimization for microbiology," *Molecules*, vol. 19, no. 8, pp. 12116–12149, 2014.

[86] D. Greenwood, F. O. Grady, and O. Grady, "The comparative performance of beta-lactam antibiotics against ampicillin sensitive Escherichia coli in conditions simulating those of the infected urinary bladder," *The British Journal of Experimental Pathology*, vol. 55, no. 3, pp. 245–250, 1974.

[87] F. Persson, I. Barkefors, and J. Elf, "Single molecule methods with applications in living cells," *Current Opinion in Biotechnology*, vol. 24, no. 4, pp. 737–744, 2013.

[88] D. I. Cattoni, J. B. Fiche, and M. Nöllmann, "Single-molecule super-resolution imaging in bacteria," *Current Opinion in Microbiology*, vol. 15, no. 6, pp. 758–763, 2012.

[89] T. Xia, N. Li, and X. Fang, "Single-Molecule Fluorescence Imaging in Living Cells," *Annual Review of Physical Chemistry*, vol. 64, no. 1, pp. 459–480, 2013.

[90] S. Wang, J. R. Moffitt, G. T. Dempsey, X. S. Xie, and X. Zhuang, "Characterization and development of photoactivatable fluorescent proteins for single-molecule-based superresolution imaging.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 111, no. 23, pp. 8452–7, 2014.

[91] S. L. Veatch, B. B. Machta, S. A. Shelby, E. N. Chiang, D. A. Holowka, and B. A. Baird, "Correlation functions quantify super-resolution images and estimate apparent clustering due to over-counting," *PLoS ONE*, vol. 7, no. 2, p. e31457, 2012.

[92] H. Qian, M. P. Sheetz, and E. L. Elson, "Single particle tracking. Analysis of diffusion and flow in two-dimensional systems.," *Biophysical Journal*, vol. 60, no. 4, pp. 910–21, 1991.

[93] A. D. Douglass and R. D. Vale, "Single-Molecule Microscopy Reveals Plasma Membrane Microdomains Created by Protein-Protein Networks that Exclude or Trap Signaling Molecules in T Cells," *Cell*, vol. 121, no. 6, pp. 937–950, 2010.

[94] G. J. Schütz, H. Schindler, and T. Schmidt, "Single-molecule microscopy on model membranes reveals anomalous diffusion.," *Biophysical Journal*, vol. 73, no. 2, pp. 1073–1080, 1997.

[95] G. J. Bakker, C. Eich, J. A. Torreno-Pina, R. Diez-Ahedo, G. Perez-Samper, T. S. van Zanten, C. G. Figdor, A. Cambi, and M. F. Garcia-Parajo, "Lateral mobility of individual integrin nanoclusters orchestrates the onset for leukocyte adhesion," *Proceedings of the National Academy of Sciences*, vol. 109, no. 13, pp. 4869–4874, 2012.

[96] F. Oswald, A. Varadarajan, H. Lill, E. J. G. Peterman, and Y. J. M. Bollen, "MreB-Dependent Organization of the E. coli Cytoplasmic Membrane Controls Membrane Protein Diffusion," *Biophysical Journal*, vol. 110, no. 5, pp. 1139–1149, 2016.

[97] C. Hellriegel, J. Kirstein, C. Bräuchle, V. Latour, T. Pigot, R. Olivier, S. Lacombe, R. Brown, V. Guieu, C. Payrastre, A. Izquierdo, and P. Mocho, "Diffusion of single streptocyanine molecules in the nanoporous network of sol-gel glasses," *Journal of Physical Chemistry B*, vol. 108, no. 38, pp. 14699–14709, 2004.

[98] M. L. Renner, L. Cognet, B. Lounis, A. Triller, and D. Choquet, "The excitatory postsynaptic density is a size exclusion diffusion environment," *Neuropharmacology*, vol. 56, no. 1, pp. 30–36, 2009.

[99] P. H. M. Lommerse, G. A. Blab, L. Cognet, G. S. Harms, B. E. Snaar-Jagalska, H. P. Spaink, and T. Schmidt, "Single-Molecule Imaging of the H-Ras Membrane-Anchor Reveals Domains in the Cytoplasmic Leaflet of the Cell Membrane," *Biophysical Journal*, vol. 86, no. 1, pp. 609–616, 2004.

[100] M. Vrljic, S. Y. Nishimura, S. Brasselet, W. E. Moerner, and H. M. McConnell, "Translational diffusion of individual class II MHC membrane proteins in cells.," *Biophysical Journal*, vol. 83, no. 5, pp. 2681–92, 2002.

[101] S. de Keijzer, A. Serge, F. van Hemert, P. H. Lommerse, G. E. Lamers, H. P. Spaink, T. Schmidt, and B. E. Snaar-Jagalska, "A spatially restricted increase in receptor mobility is involved in directional sensing during Dictyostelium discoideum chemotaxis," *Journal of Cell Science*, vol. 121, no. 10, pp. 1750–1757, 2008.

[102] J. M. Crane and A. Verkman, "Long-Range Nonanomalous Diffusion of Quantum Dot-Labeled Aquaporin-1 Water Channels in the Cell Plasma Membrane," *Biophysical Journal*, vol. 94, no. 2, pp. 702–713, 2008.

[103] C. L. Hsieh, S. Spindler, J. Ehrig, and V. Sandoghdar, "Tracking single particles on supported lipid membranes: Multimobility diffusion and nanoscopic confinement," *Journal of Physical Chemistry B*, vol. 118, no. 6, pp. 1545–1554, 2014.

[104] C. Jung, J. Kirstein, B. Platschek, T. Bein, M. Budde, I. Frank, K. Müllen, J. Michaelis, and C. Bräuchle, "Diffusion of oriented single molecules with switchable mobility in networks of long unidimensional nanochannels," *Journal of the American Chemical Society*, vol. 130, no. 5, pp. 1638–1648, 2008.

[105] R. Das, C. W. Cairo, and D. Coombs, "A hidden Markov model for single particle tracks quantifies dynamic interactions between LFA-1 and the actin cytoskeleton," *PLoS Computational Biology*, vol. 5, no. 11, 2009.

[106] A. Robson, K. Burrage, and M. C. Leake, "Inferring diffusion in single live cells at the single-molecule level.," *Philosophical transactions of the Royal Society of London. Series B, Biological Sciences*, vol. 368, no. 1611, 2013.

[107] N. Monnier, Z. Barry, H. Y. Park, K.-C. Su, Z. Katz, B. P. English, A. Dey, K. Pan, I. M. Cheeseman, R. H. Singer, and M. Bathe, "Inferring transient particle transport dynamics in live cells," *Nature Methods*, vol. 12, no. 9, pp. 838–840, 2015.

[108] P. K. Koo, M. Weitzman, C. R. Sabanaygam, K. L. van Golen, and S. G. J. Mochrie, "Extracting Diffusive States of Rho GTPase in Live Cells: Towards In Vivo Biochemistry," *PLoS Computational Biology*, vol. 11, no. 10, pp. 1–26, 2015.

[109] J. Munkres, "Algorithms for the Assignment and Transportation Problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[110] I. M. Sokolov, "Models of anomalous diffusion in crowded environments," *Soft Matter*, vol. 8, no. 35, pp. 9043–9052, 2012.