Method for simultaneous localization and parameter estimation in particle tracking experiments

Trevor T. Ashley[1,*] and Sean B. Andersson[1,2,†]

[1]*Department of Mechanical Engineering, Boston University, Boston, Massachusetts 02215, USA*

[2]*Division of Systems Engineering, Boston University, Boston, Massachusetts 02215, USA*

(Received 31 July 2015; published 5 November 2015)

We present a numerical method for the simultaneous localization and parameter estimation of a fluorescent particle undergoing a discrete-time continuous-state Markov process. In particular, implementation of the method proposed in this work yields an approximation to the posterior density of the particle positions over time in addition to maximum likelihood estimates of fixed, unknown parameters. The method employs sequential Monte Carlo methods and can take into account complex, potentially nonlinear noise models, including shot noise and camera-specific readout noise, as well as a wide variety of motion models and observation models, including those representing recent engineered point spread functions. We demonstrate the technique by applying it to four scenarios, including a particle undergoing free, confined, and tethered diffusions.

DOI: 10.1103/PhysRevE.92.052707 PACS number(s): 87.80.Nj, 05.40.Jc, 02.50.Tt, 05.45.Tp

I. INTRODUCTION

The tracking of fluorescent particles, commonly referred to as single-particle tracking (SPT), has become a widely used tool since its initial development in the late 1980s [1,2]. Since then, various improvements, including brighter fluorophores (e.g., quantum dots) and sensitive photodetectors (e.g., electron multiplying charge-coupled devices), have significantly improved the conclusions that may be drawn from SPT experiments; some examples include the elucidation of clathrin-mediated endocytosis of influenza [3] and the traversal of myosin VI along actin filaments [4] (see also recent survey articles such as Refs. [5–7]). Within this paradigm, the feature of interest, often smaller than that which can be resolved by an optical microscope, is labeled with a nanometer-scale fluorescent particle and observed during a time period of interest. Typically, one postprocesses the data to localize the particle within each image frame and then analyzes the resulting trajectory to infer various statistics regarding its motion. The most commonly estimated statistic is the diffusion coefficient, as this parameter characterizes a large range of physical properties, including size, shape, and viscosity. Although the estimation of diffusion coefficients has come to maturity over the past several years [8–10], the diffusion coefficient is only one of many parameters that may be of interest. It is well known in practice that particle motion is not limited to pure diffusion or any other single model. Motion may be confined, tethered, directed, or anomalous and may even switch between different modes [11,12]. To the best of our knowledge, there is no body of work that simultaneously considers the localization problem jointly with optimal estimation of parameters describing arbitrary stochastic motion and observation models; leveraging from the field of nonlinear system identification [13], we describe and demonstrate such a method in this work.

After images of the fluorescent particle have been acquired, the analyst usually estimates the position of the center of the particle within each frame. This process, known as localization, produces a sequence of estimated positions over time; as was shown in Ref. [14], the key feature here is that, given a sufficient number of photons, the position of the particle can be estimated with subnanometer resolution. A vast number of localization algorithms have been proposed, ranging in complexity, speed, and performance [15–18]. A commonality among the majority of these algorithms is that they rely solely on intraframe information; in other words, they assume the position of the particle in one image yields no information regarding the preceding or succeeding estimates. In contrast, one approach, proposed in Ref. [19], incorporated model-based sequential Monte Carlo methods (i.e., "particle filtering") that utilized the interframe information to estimate an approximate probability density function of the particle's location given the images. The applicability of this method in an experimental setting is limited, however, due to the fact that its implementation requires prior knowledge of the true motion parameters (e.g., diffusion coefficients) while in practice such values are typically estimated from the data.

Once the particle has been localized, parameters describing its motion may be inferred. In the case of diffusion coefficients, the most common method of estimation is via least-squares fit to the mean-square displacement (MSD). The relationship among the MSD, the diffusion coefficient, and the localization uncertainty is now common knowledge within the particle tracking community. Previously considered an art [20], the optimal estimation of these parameters has been extensively studied [8] and a computationally efficient algorithm for the maximum likelihood estimation of the diffusion coefficient developed. The algorithm begins with the estimated trajectory of the particle and makes the somewhat limiting assumption that the localization uncertainty can be modeled by independent and identically distributed Gaussian noise. While a common assumption, uncertainty in the localization arises from a variety of sources, including shot noise in the photon generation process, background noise, read-out noise in the detectors, and algorithmic error, and the Gaussian assumption is hard to justify theoretically. Further, its use precludes multimodal distributions for the position of the particle; such distributions (i.e., ones that exhibit multiple peaks) can be useful for detection schemes (such as wide-field imaging) that have an inherent symmetry (such as reflection through the focal plane).

[*] tashley@bu.edu

[†] sanderss@bu.edu

Although the aforementioned localization and parameter estimation methods have been widely used with success in practice, their applicability is limited to a subset of potential situations. To see the limitation, consider a "corral-like" confinement of proteins on a plasma membrane [21] where the analyst is not only interested in the optimal diffusion coefficient estimate of a protein on the surface but an optimal estimate for the area of confinement. By assuming free diffusion, as the methods mentioned above do, the analyst has already biased the resulting diffusion coefficient estimate since the model is inaccurate; moreover, those methods provide no estimate for the size of confinement. While MSD methods can be used on more general models, this approach relies on *ad hoc* fitting choices and is sensitive to measurement noise.

In this work, we present an inference method which allows for the optimal estimation of parameters describing reasonably arbitrary motion and observation models. In addition, the method couples both the localization and parameter estimation steps, thereby taking into account both interframe and intraframe information. The method, taken from recent work developed in the field of nonlinear system identification, employs the widely known expectation maximization (EM) algorithm in conjunction with sequential Monte Carlo methods. In Sec. II we present background theory for this method, connecting the general theory to the specifics of the single-particle tracking problem, and in Sec. III we describe a framework that can be used for tracking individual fluorescent particles from a sequence of fluorescence microscopy images. To demonstrate its applicability, we present four demonstrations in Sec. IV; these demonstrations include the simultaneous localization and parameter estimation of freely diffusing, confined, and tethered fluorescent particles observed by a wide-field fluorescence microscope.

II. THEORETICAL BACKGROUND

In this section, we review a numerical technique that, given a sequence of discrete-time observations, simultaneously infers a hidden state driven by a fixed-structure discrete-time motion model in addition to fixed parameters that characterize that model. This technique is based on the EM algorithm and was originally proposed by Schön, Wills, and Ninness in the context of nonlinear system identification [13]. While the technique and the description given here is quite general, throughout the section we will connect it to the SPT setting. A detailed description of the application of the method to the tracking problem is given in Sec. III.

A. The EM algorithm for parameter identification

Consider the problem of identifying an unknown fixed parameter $\theta \in \mathbb{R}^{n_\theta}$ for the state and observation models

$$x_1 \sim p_\theta^{\text{Init}}(x_1), \tag{1}$$

$$x_{k+1} \sim p_\theta^{\text{Mot}}(x_{k+1}|x_k), \tag{2}$$

$$y_k \sim p_\theta^{\text{Obs}}(y_k|x_k), \tag{3}$$

with $x_k \in \mathbb{R}^{n_x}$, $y_k \in \mathbb{R}^{n_y}$, and the distributions $p_\theta(\cdot|\cdot)$ of known form. (Here $\mathbb{R}^n$ denotes the vector space of real numbers in n dimensions.) We assume only N values $Y_N \triangleq \{y_1,\ldots,y_N\}$ are available to derive an estimate of θ, namely $\hat{\theta}$.

In the context of SPT, the state x_k denotes the location of the particle in either two or three dimensions at a discrete time index k. The distribution p_θ^{Init} describes the probability distribution of the particle location at the start of the experiment. The motion model p_θ^{Mot} is a probability distribution describing how the particle evolves in time and may be given by a diffusion, directed motion, or other dynamic process. The observation model describes the fluorescent measurement. For example, assuming a wide-field imaging modality, y_k signifies a pixilated image acquired at discrete time k. The observation model $p_\theta^{\text{Obs}}(\cdot)$ is a probability distribution capturing the statistics of the measurement process itself and may involve, for example, Poisson-distributed shot noise, Gaussian read-out noise, or other features specific to the experimental setting. The parameter θ may describe properties of the motion (such as diffusion coefficients) as well as of the observation process (such as peak intensity). Section III, as well as the examples in Sec. IV, describe this context in greater detail.

The primary goal is to estimate the parameter θ. While many different techniques could be used, we focus on the maximum likelihood estimator due to its well-characterized properties, including asymptotic consistency, efficiency, and normality [22]. Deriving the maximum likelihood estimate involves maximizing the joint density (i.e., the likelihood) of the observations conditioned on the parameter θ,

$$\hat{\theta} = \arg\max_{\theta} \; \log p_\theta(Y_N). \tag{4}$$

It is most often the case that this density is unknown or intractable, and, consequently, an ML estimate cannot be derived analytically. For example, in the SPT application, (4) corresponds to attempting to estimate motion and observation parameters directly from the image data, without knowledge of the particle trajectory. When the density in (4) is unknown or intractable, the EM algorithm provides an alternative for finding the ML estimate.

EM, originally proposed in Ref. [23], is a numerical, iterative method for calculating ML estimates that is typically used when the joint density $p_\theta(X_N,Y_N)$ is known, where $X_N \triangleq \{x_1,\ldots,x_N\}$ is denoted as the *latent* set of random variables. Instead of calculating the full likelihood by integrating the joint density over the full domain of the N-dimensional latent state, the EM algorithm approximates the likelihood by conditionally averaging over the latent state; mathematically, this is given by

$$Q(\theta,\hat{\theta}_e) \triangleq \int \log\left[p_\theta(X_N,Y_N)\right] p_{\hat{\theta}_e}(X_N|Y_N)\, dX_N. \tag{5}$$

For the SPT problem, the latent state X_N is the trajectory of the particle, and thus EM takes advantage of the coupling between the trajectory and the image data as encapsulated by the motion (2) and observation models (3).

Note that the expectation in (5) is with respect to the conditional density $p_{\hat{\theta}_e}(X_N|Y_N)$ with $\hat{\theta}_e$ defined to be some estimate for θ at iteration e. It was shown in Ref. [23] that for each choice of $\hat{\theta}_{e+1}$ such that $Q(\hat{\theta}_{e+1},\hat{\theta}_e) > Q(\hat{\theta}_e,\hat{\theta}_e)$, the likelihood $p_{\hat{\theta}_{e+1}}(Y_N)$ subsequently increases. One method to guarantee an increase is to update the new value so

$$\hat{\theta}_{e+1} = \arg\max_{\theta} \quad Q(\theta,\hat{\theta}_e). \tag{6}$$

By alternating between the expectation and maximization steps, Q as defined in (5) will converge to a (local) maximum as e becomes large, and, consequently, $\hat{\theta}_e$ will converge to a (local) ML estimate of θ.

Implementation of the EM algorithm requires evaluation of $Q(\theta,\hat{\theta}_e)$. A tedious but straightforward calculation (see Ref. [13]) allows (5) to be rewritten as

$$Q(\theta,\hat{\theta}_e) = I_1 + I_2 + I_3, \tag{7}$$

where

$$I_1 \triangleq \int \log\left[p_\theta^{\mathrm{Init}}(x_1)\right] p_{\hat{\theta}_e}(x_1|Y_N)dx_1, \tag{8a}$$

$$I_2 \triangleq \sum_{k=1}^{N-1} \iint \log\left[p_\theta^{\mathrm{Mot}}(x_{k+1}|x_k)\right] \times p_{\hat{\theta}_e}(x_{k+1},x_k|Y_N)dx_k dx_{k+1}, \tag{8b}$$

$$I_3 \triangleq \sum_{k=1}^{N} \int \log\left[p_\theta^{\mathrm{Obs}}(y_k|x_k)\right] p_{\hat{\theta}_e}(x_k|Y_N)dx_k. \tag{8c}$$

Here (8a) corresponds to the initial distribution (1), (8b) corresponds to the motion model (2), and (8c) corresponds to the observation model (3). The calculation of these expressions require evaluation of the posterior density $p_{\hat{\theta}_e}(x_k|Y_N)$ and the sequential-pairwise joint posterior density $p_{\hat{\theta}_e}(x_k,x_{k+1}|Y_N)$; these are, unfortunately, challenging to compute in most cases. For the case where both the motion and observation models constitute a linear, time-invariant system, the posterior densities are Gaussian with sufficient statistics calculated by optimal estimation methods such as the Kalman filter and smoother [24]. In general, however, the underlying models are nonlinear and non-Gaussian in nature, such as when the particle motion follows a confined diffusion or when the presence of Poisson-distributed shot noise is accounted for explicitly in the observation model. In this work, these difficulties are overcome by using sequential Monte Carlo methods to simplify (7) by approximating the posterior densities as discrete, weighted sums of δ functions.

B. A sequential Monte Carlo approximation

Sequential Monte Carlo (SMC) methods approximate the conditional posterior densities in (8b) and (8c) at each EM iteration e by assuming they may be approximated by a weighted sum of M randomly drawn point estimates such that

$$p_{\hat{\theta}_e}(x_k|Y_N) \approx \sum_{i=1}^{M} w^i_{k|N,e}\,\delta\big(x_k - x^i_{k|N,e}\big), \tag{9a}$$

$$p_{\hat{\theta}_e}(x_k,x_{k+1}|Y_N) \approx \sum_{i=1}^{M}\sum_{j=1}^{M} w^{ij}_{k|N,e} \times \delta\big(x_k - x^i_{k|N,e}, x_{k+1} - x^j_{k+1|N,e}\big), \tag{9b}$$

where $x^i_{k|N,e}$ is the ith estimate at time step k and EM iteration e, $\delta(\cdot)$ is a Dirac δ function, and $w^i_{k|N,e}$ and $w^{ij}_{k|N}$ are weights determined by how well the point estimates approximate the posterior distribution. The value of these weights are driven in part by determining how likely the ith measurement was given the point estimate $x^i_{k|N,e}$. In the context of SPT, this translates to setting the weight by how likely it was to acquire the actual image data assuming that the particle was at the position of the point estimate. A discussion of the calculation of these weights for the demonstration examples in Sec. IV is given in the Appendix. A general treatment for how these estimates and weights are calculated is beyond the scope of this work; the interested reader is referred to Refs. [25–27] for more information.

Upon application of the SMC approximation (9) to the calculation of $Q(\theta,\hat{\theta}_e)$ in (7), one obtains the approximation

$$Q(\theta,\hat{\theta}_e) \approx \hat{Q}(\theta,\hat{\theta}_e) = \hat{I}_1 + \hat{I}_2 + \hat{I}_3, \tag{10}$$

where

$$\hat{I}_1 \triangleq \sum_{i=1}^{M} w^i_{1|N,e} \log\left[p_\theta^{\mathrm{Init}}\big(x^i_{1|N,e}\big)\right], \tag{11a}$$

$$\hat{I}_2 \triangleq \sum_{k=1}^{N-1}\sum_{i=1}^{M}\sum_{j=1}^{M} w^{ij}_{k|N,e} \log\left[p_\theta^{\mathrm{Mot}}\big(x^j_{k+1|N,e}|x^i_{k|N,e}\big)\right], \tag{11b}$$

$$\hat{I}_3 \triangleq \sum_{k=1}^{N}\sum_{i=1}^{M} w^i_{k|N,e} \log\left[p_\theta^{\mathrm{Obs}}\big(y_k|x^i_{k|N,e}\big)\right]. \tag{11c}$$

Note the similarities between (8) and (11); the SMC approximation of the posterior densities transforms potentially intractable integrals into straightforward summations. The SMC-based EM algorithm proceeds by calculating $\hat{Q}(\theta,\hat{\theta}_e)$ for some $\hat{\theta}_e$ and then by updating the estimate according to

$$\hat{\theta}_{e+1} = \arg\max_{\theta} \quad \hat{Q}(\theta,\hat{\theta}_e). \tag{12}$$

These two steps, namely the calculation of $\hat{Q}(\theta,\hat{\theta}_e)$ via SMC and its subsequent maximization over θ, are repeated until some convergence criterion is satisfied. For the remainder of this work, we denote this process as the SMC-EM algorithm.

It is to be noted that the SMC-EM algorithm, unlike the original EM algorithm, provides no guarantee of convergence to a (local) maximum. However, as the number of estimates M becomes large, $\hat{Q}$ approaches Q, thereby implying greater likelihood of converging to an extremum. Analysis of this behavior is further detailed in Ref. [13].

III. AN APPLICATION: SINGLE-PARTICLE TRACKING

In the application of SPT for the study of dynamic biological phenomena, the goal is to infer the location of the particle as well as any relevant parameters characterizing its motion from a collection of image sequences of one or more fluorescent particles acquired over some time period. The data may be from a variety of imaging modalities, including wide-field, total internal reflection (TIRF), structured illumination (SIM), and laser-scanning confocal microscopy (LSCM). As was mentioned in Sec. I, the standard method for analyzing these image sequences involves first inferring the trajectory of the particle by estimating the position of the particle in each image (via, e.g., a nonlinear fit). From these trajectories, model parameters, such as diffusion coefficients, are inferred by fitting a curve to the empirical MSD of the estimated positions.

In contrast to the standard method, which is noniterative and produces only point estimates of the particle position, the SMC-EM technique developed in Sec. II takes into account the coupling between both parameter estimation and localization and iterates between the two to refine their values over each iteration. Specifically, at the eth iteration, the probability density describing the particle is first estimated in all the images via SMC methods with respect to known $\hat{\theta}_e$-parametrized motion and observation models. An improved value for $\hat{\theta}_e$, namely, $\hat{\theta}_{e+1}$, is calculated from the maximization of $\hat{Q}(\theta,\hat{\theta}_e)$ in (10). The method then uses this refined estimate to re-estimate the probability distributions of the particle position, again with SMC methods. This procedure repeats until some convergence criterion is satisfied.

To apply the SMC-EM technique to a specific experimental setting, one must define the parametrized observation and motion models. As an illustration, in the remainder of this section we show this process by developing these models for the general case of tracking a single moving particle imaged with a wide-field fluorescence microscope. The use of the models in the SMC-EM algorithm is detailed in Sec. IV and the Appendix.

A. A wide-field fluorescence observation model

We first describe how the observation model (3) may be formulated to represent the scenario where a subdiffraction-sized particle is imaged with a wide-field fluorescence microscope.

We assume a camera takes images of the particle at discrete, regularly spaced periods of length Δt and that the position of the particle at these instances k is (x_k, y_k, z_k). Since the particle is smaller than the diffraction limit of the optical system, the intensity pattern measured by the camera can be approximated by the point spread function [PSF (F_{PS} in the equations)] of the optical system. We note that the PSF may take any spatial shape; this includes the popular Gibson-Lanni wide-field model [28] as well as the double-helix [29] or other engineered PSFs [30,31]. In the model considered here, the detector consists of P rectangular pixels of known length Δx and width Δy. Let $(\bar{x}_{p,k}, \bar{y}_{p,k})$ denote the center of pixel p at time step k; the expected intensity measured by this pixel is

$$\lambda_{p,k} = \int_{x_{p,k}^{\min}}^{x_{p,k}^{\max}} \int_{y_{p,k}^{\min}}^{y_{p,k}^{\max}} G F_{\mathrm{PS}}(x_k - \xi', y_k - \xi'', z_k) d\xi' d\xi'', \quad (13)$$

where the integration limits are defined by

$$x_{p,k}^{\min} = \bar{x}_{p,k} - \frac{\Delta x}{2}, \quad (14a)$$

$$x_{p,k}^{\max} = \bar{x}_{p,k} + \frac{\Delta x}{2}, \quad (14b)$$

$$y_{p,k}^{\min} = \bar{y}_{p,k} - \frac{\Delta y}{2}, \quad (14c)$$

$$y_{p,k}^{\max} = \bar{y}_{p,k} + \frac{\Delta y}{2}. \quad (14d)$$

Note that the pixel center may depend on the discrete time index k thereby allowing the center of the image to be adjusted during the experiment. For the remainder of this work, we assume the PSF is normalized so $F_{\mathrm{PS}}(0,0,0) = (\Delta x \Delta y)^{-1}$ and that the multiplicative gain G separately scales the intensity of the whole image.

Given the quantized nature of light, shot noise is a prevalent noise source in most situations involving CCD cameras. In addition, light from surrounding objects, usually due to autofluorescence, may contribute as noise during the imaging process; here we consider such a background that is distributed uniformly in space and that contributes additively to the light emitted by the fluorophore. The measured number of counts at pixel p at time-instance k is therefore the Poisson random variable

$$I_{p,k} \sim X_{\mathrm{Poi}(\lambda_{p,k} + N_{\mathrm{bgd}})}, \quad (15)$$

where N_{bgd} is the number of background counts per pixel per update period. We note that there are many factors that contribute to the overall noise floor of these imaging devices, including readout noise, pixel nonuniformity, and dark current. It has been shown, for example, that electron-multiplying CCDs exhibit non-Poissonian noise [32]; Ref. [33] presents one potential probabilistic model that could be incorporated. All of these factors could be taken into account by the SMC-EM algorithm; for simplicity of presentation we focus on the model in (15).

B. Relevant motion models

Like the observation model, which is specific to the imaging modality (e.g., wide field, confocal, etc.), the motion model (2) is also specific to the experimental setting (e.g., diffusion, directed, etc.). In general, the motion model represents the probability that the particle, given its current location x_k, transitions to the next potential location x_{k+1} over the time period Δt. An important assumption here is that the next location depends *only* on the current location; in other words, the particle moves according to a Markov process. Although the position of the particle realistically fluctuates continuously in both time and space, it is often the case that its position can be sampled at discrete time instances k without affecting its Markovian nature. This is true for, for example, a particle undergoing isotropic diffusion since the solution to its Fokker-Planck equation is known to be a normal distribution with mean x_k and variance $2D\Delta t$. It is also true for more complicated motion, such as tethered or confined diffusions. We now develop three particular motion models for use in the demonstrations.

1. Diffusion in two dimensions

The first motion model considered is two-dimensional isotropic diffusion with negligible axial motion. This model is applicable to several situations related to biology, including those involving membrane dynamics and viral trafficking. The transition density for this motion model is given by the solution to the diffusion equation [34] and takes the form of a normal distribution; when discretized with a time period of $\Delta t'$, the motion model in the form of (2) for the x axis is

$$p(x_{k+1}|x_k) = \frac{1}{\sqrt{4\pi D \Delta t'}} \exp\left[-\frac{(x_{k+1} - x_k)^2}{4D\Delta t'} \right], \quad (16)$$

where D is the diffusion coefficient. The transition density for y_k is identical to that of (16). Under the assumption that both axes are independent, the motion model for the joint position (x_k, y_k) is simply the product of the individual normal distributions for each axis. The position of the particle in z, however, is assumed zero for all time and does not have a transition density; to extend this model to three dimensions one simply includes another term of the form (16) for the z direction.

2. Three-dimensional diffusion with axial confinement

Although the diffusive model (16) is often used in practice, it makes the assumption that the particle is unconstrained in all directions. There are, however, many situations where this does not hold. As an example, we next consider a three-dimensional isotropic diffusion with non-negligible axial motion confined within the interval $[-L/2, L/2]$ with L defining the length of the confinement channel. Since the motion in the x and y axes are independent diffusions, this model reuses the transition densities (16) for these directions. The transition density for the z position, however, is given by the solution to the diffusion equation with reflecting boundaries at $z = -L/2$ and $z = +L/2$, given by

$$p(z_{k+1}|z_k) = \frac{1}{L} + \frac{2}{L}\sum_{n=1}^{\infty} \exp\left[-D\Delta t'\left(\frac{n\pi}{L}\right)^2\right] \times \cos\left[\frac{n\pi}{L}\left(z_{k+1}+\frac{L}{2}\right)\right]\cos\left[\frac{n\pi}{L}\left(z_k+\frac{L}{2}\right)\right]; \tag{17}$$

see Ref. [35] for a derivation. Since the motion in each axis is independent, the joint transition density in the form of (2) is given by the product of the z transition density (17) with the x and y transition densities (16).

3. Elastic tethering in three dimensions

Last, we consider the case where a particle is elastically tethered to a fixed anchor. This model is common in the case of tethered particle microscopy where one end of a biopolymer is fixed to a coverslip and a fluorescent particle is attached the other (free) end; the position of the bead over time yields information regarding the structure of the biopolymer (see, e.g., Ref. [36]). We assume the location of the tethering point is known and fixed to the origin and that the tether is sufficiently stiff and isotropically obeys Hook's law. Inspired by the model presented in Ref. [37], the position of the particle can be modeled by the Ornstein-Uhlenbeck process with transition density

$$p(x_{k+1}|x_k) = \sqrt{\frac{A}{2\pi D(1-e^{-2A\Delta t'})}} \times \exp\left\{-\frac{A}{2D}\left[\frac{(x_{k+1}-x_k e^{-A\Delta t'})^2}{1-e^{-2A\Delta t'}}\right]\right\}, \tag{18}$$

in the x axis; the densities for y and z are identical in form. The stiffness coefficient $A > 0$ and the diffusion coefficient D both determine the behavior of the motion about the tethered point. Due to independence among axes, the joint transition density in the form of (2) is once again given by the product of the individual densities (18) for each axis.

IV. DEMONSTRATION OF METHOD

In this section, we discuss four numerical demonstrations that showcase the capabilities of the SMC-EM algorithm. The four demonstrations rely on three sets of data, each consisting of multiple trials with each trial an image sequence of a single particle undergoing motion according to one of the models introduced in Sec. III B. The first data set is a two-dimensional isotropic diffusion, the second is a three-dimensional isotropic diffusion with axial confinement, and the third is a three-dimensional tether. The simulated experimental setup is a wide-field imaging scenario using a Debye diffraction integral satisfying the sine condition in all three data sets.

The three data sets are used throughout four distinct demonstrations. The first data set considers the two-dimensional diffusion and attempts to estimate, under the assumption of anisotropicity, the two diffusion coefficients. To ground the results from the SMC-EM algorithm, we compare them to a conventional localize-then-estimate approach using Gaussian fit and the ML estimator developed in Ref. [8]. The second data set considers the three-dimensional axially confined isotropic diffusion and estimates each diffusion coefficient in addition to the confinement length. The third data set considers the three-dimensional isotropic tether and estimates the stiffness coefficient in addition to the diffusion coefficient. The three aforementioned demonstrations assume precise knowledge of the PSF. This is, however, often not the case in most settings and thus in the fourth demonstration we show that the SMC-EM algorithm is capable of estimating PSF parameters as well. We revisit the three-dimensional tether and estimate the peak intensity of the PSF in addition to the same parameters as before. The specific parameter values for each of these demonstrations, such as diffusion coefficients, are defined in Table I.

TABLE I. Parameter values used in the generation of ground-truth data and for each of the four demonstrations.

Symbol	Parameter	Value
A	Stiffness coefficient	1.0 s^{-1}
D	Diffusion coefficient	0.01 $\mu\text{m}^2/\text{s}$
E	Number of SMC-EM iterations	10 iterations
G	Peak intensity	100 counts
K	Number of sequences	40 sequences
L	Length of channel	500 nm
M	Number of SMC point estimates	125
n	Refraction index	1.33
N	Sequence length	100 images
N_{bgd}	Background noise	10 counts
N_{sub}	Subsamples per image	100 subsamples
NA	Numerical aperture	1.2
P	Number of pixels	25 pixels
$\Delta x, \Delta y$	Effective pixel length	100 nm
δt	Shutter period	10 ms
Δt	Imaging period	100 ms
λ	Emission wavelength	540 nm

A. Generation of ground-truth data

For each of the three models, we generated 40 data sets, each composed of 100 images. Specifically, the first set of images was generated from the two-dimensional isotropic diffusion model described in Sec. III B 1, the second set of images was generated from the axially confined diffusion model described in Sec. III B 2, and the third set of images was generated from the elastic tether model described in Sec. III B 3. The observation models were identical for all image sequences. The number of images N was chosen with regard to the imaging time $\Delta t = 100$ ms (i.e., 10 frames per second), yielding a total duration of 10 s for each sequence. In practice, $N\Delta t$ is selected according to the hardware's imaging capabilities, the fluorophore's emission capacity, and the time scale of the particle's motion; the values chosen in this work were inspired by a recent competition [18] as well as a brief survey of the literature.

To generate each sequence of images, independent trajectories consisting of $N \times N_{\text{sub}}$ positions were generated from one of the three aforementioned motion models; here $N_{\text{sub}} = 100$ represents a subsampling factor with each position in the trajectory spaced $\delta t = \Delta t / N_{\text{sub}}$ seconds apart, where $\Delta t = 100$ ms is the imaging period. In this work, N_{sub} was chosen large enough so the motion between subimages was negligible; specifically, the value was selected to ensure that the probability that a particle following an isotropic three-dimensional diffusion deviated more than 25 nm between each subimage was less than 0.2%.

From these trajectories, images were generated using the wide-field fluorescence observation model described in Sec. III A with

$$F_{\text{PS}}(x,y,z) = \left| C \int_0^{\alpha} \sqrt{\cos\theta}\, J_0(\kappa \sin\theta \sqrt{x^2+y^2}) \times \exp(-i\kappa z \cos\theta) \sin\theta\, d\theta \right|^2, \tag{19}$$

where C is a complex constant chosen so $F_{\text{PS}}(0,0,0) = (\Delta x \Delta y)^{-1}$, $J_0(\cdot)$ is a zeroth-order Bessel function of the first kind, $\kappa \triangleq 2\pi n/\lambda$ is the wave number of the emitted light, and $\alpha \triangleq \sin^{-1}(\text{NA}/n)$ is the maximum semiangle of the objective lens. This PSF model, which is based on the Debye diffraction integral, is well suited for approximating the near-focus nonparaxial distribution of light and is valid for high-NA objective lenses [38]. The parameters describing the PSF were set to $\text{NA} = 1.2$, $\lambda = 540$ nm and $n = 1.33$, which are common values that one may find in an experimental setting. In addition, we assumed the effective pixel width (after magnification) was $\Delta x = \Delta y = 100$ nm.

Realistically, an image of the particle is formed by accumulating photons during an exposure period. To replicate this effect, we assumed the camera accumulated photons continuously during the first $\delta t = 10$ ms of each imaging period $\Delta t = 100$ ms by averaging the first 10 consecutive images in the period and ignoring the rest. As noted above, δt was chosen sufficiently small so the motion of particle during photon accumulation was negligible while remaining large enough to ensure an adequate number of photons was collected.

The particle was set to have a peak fluorescence of $G = 100$ counts over the entire exposure time δt. The background noise was assumed to be uniform throughout the image with a fixed number of $N_{\text{bgd}} = 10$ counts counts during the exposure period. For simplicity we assumed there was no blinking or bleaching during the accumulation period.

In practice, when one acquires images in real experimental situations, many particles may be present and segmentation and linking among frames must be performed. The choice of method for this step affects the quality of the final localization and parameter estimation. To use the algorithm presented in this work, however, *any* segmentation and linking technique may be applied to generate a sequence of pixel arrays (see, e.g., Ref. [18]). To abstract the impact of segmentation from the behavior of the SMC-EM algorithm, "perfect segmentation" was assumed in the trials presented here. Thus, each image was approximately centered in each frame by an oracle, and only one particle was present throughout the image generation process. As will be discussed later, the computational complexity of the proposed algorithm increases with the number of pixels observed per frame; consequently, the user must be judicious as to how large the image should be to ensure that enough information regarding the particle is present without including too many "empty" pixels. For the image sequences described in this work, each image contained $P = 25$ pixels arranged into a square $\sqrt{P} \times \sqrt{P}$ array; the effective length of each segmented image was 0.5 μm, which is approximately twice the full width at half-maximum value when the particle is in focus.

Since the localization resolution depends on the total number of informative photons acquired during the imaging process, one possible metric for comparing the three data sets is the average number of photons acquired per (segmented) frame. The average number of photons was 756 per frame for the first data set, 737 per frame for the second data set, and 721 per frame; recall that each segmented frame consisted of $P = 25$ pixels. Thus, the average number of photons per frame is nearly equivalent for each of the demonstrations and one may expect similar localization resolution.

B. Algorithm implementation

Recall from Sec. II that the SMC-EM algorithm iterates between two distinct steps. On the first iteration of the algorithm, the *expectation* step yields an evaluation of (10) through application of SMC-based filtering and smoothing techniques to the acquired data (i.e. images) with respect to the motion (2) and observation (3) models and an *a priori* parameter estimate $\hat{\theta}_0$. The *maximization* step yields an improved estimate $\hat{\theta}_1$ through the evaluation of (12). The expectation step is then executed again with respect to the new estimate $\hat{\theta}_1$, and the maximization step follows afterward yielding $\hat{\theta}_2$. This process continues until a termination criterion is satisfied.

For each of the demonstrations presented in this work, the SMC-EM algorithm was terminated after a fixed number $E = 10$ of iterations. The number of iterations was chosen large enough so the parameters approximately converged to a fixed value. Although more sophisticated termination criteria may be used, such as those employing optimal change point detection [39], such methods were deemed unnecessary for the demonstrations in this work.

The number of Monte Carlo estimates M was chosen to be 125 for each of the four demonstrations. In general, proper selection of this parameter requires some degree of experimentation. Ideally, to minimize the Monte Carlo variability, the parameter M should be large; however, as further discussed in the Appendix, the SMC-EM algorithm implemented in this work has a computational complexity that scales quadratically with M. Thus, using a large number of Monte Carlo estimates may significantly impede throughput. One way of determining an appropriate value for M is to process the data using an initial value and then process the data again using double the initial value. If the statistics of the resulting parameter estimates did not improve between M_0 and M_1, then M_0 may be deemed sufficient. Otherwise, process the data again using quadruple the value and repeat the comparison. In this work, $M = 125$ was chosen because $M = 250$ did not yield an improvement in the variance of the parameter estimates in any of the demonstrations.

The expectation step, as defined by $\hat{Q}$ in (10), requires that the posterior densities be represented in the form of a sum of weighted estimates of the particle's position (9). To calculate these weighted estimates, we employed the sampling importance resampling (SIR) algorithm [40] in conjunction with the forward-filtering backward-smoothing (FFBS) algorithm [41]; implementation details for these algorithms are given in Appendixes 1a and 1b, respectively.

For all the demonstrations, the initial parameter estimates were randomly generated within one order of magnitude of their true value (aside from the confinement length L for which an upper bound was used instead). It is important to note that the SMC-EM algorithm may fail to converge if the initial parameter estimates are too far from their true values. Since the expectation step relies on a finite number of Monte Carlo estimates to generate an approximate posterior density, a severe mismatch between the data and assumed model may result in most of the Monte Carlo estimates having low (or even zero) likelihood. In the case that all the estimates have zero likelihood, the approximate posterior density becomes degenerate and the algorithm fails. The probability of this failure occurring can be reduced by using a larger number of Monte Carlo estimates or by using more efficient SMC algorithms. Alternatively, if initial parameter estimates are not known to a degree so the SMC-EM algorithm converges, methods other than SMC-EM may be used to provide an initial estimate that can be used in the subsequent application of SMC-EM.

C. Measures of algorithm performance

Recall from Sec. II that the output of the SMC-EM algorithm consists of two parts: the parameter estimates and the joint posterior density estimates on the position of the particle. Although assessing the quality of the estimated parameters by comparing them to their true values is straightforward, determining the performance of localization is not. This is mostly due to the fact that the SMC-EM algorithm yields a Monte Carlo approximation of the full joint posterior density of the particle's position over time rather than just a point estimate. This provides significantly more information than a point estimate. To assess the performance of localization in this work, we assume the most representative estimate of the particle's position at a given time is the weighted arithmetic mean over all estimates. In other words, the position estimate $(\hat{x}_{k,e},\hat{y}_{k,e},\hat{z}_{k,e})$ is used as a surrogate for the joint posterior density at time step k, where

$$\hat{x}_{k,e} \triangleq \sum_{i=1}^{M} w^{i}_{k|N,e} x^{i}_{k|N,e}, \tag{20}$$

with $\hat{y}_{k,e}$ and $\hat{z}_{k,e}$ defined similarly. Given these surrogates, the root-mean-squared (RMS) localization error can be calculated for each image sequence.

The variability in the results presented in this section—specifically, the bias and the standard deviations in both the parameter estimates and localization errors—are primarily driven by two sources of random error. The first, experimental variability, comes from several sources, including randomness in the motion of the particle, shot and background noise in the measurement, motion blur in the data, and the finite pixel size of the detector. This variability is inherent to the experimental setting itself and is present regardless of the estimation scheme.

The second source of variability arises from the Monte Carlo nature of the SMC element of the estimation scheme. This variability scales as $\sqrt{M}$ and thus vanishes in the limit of a large number of point estimates in the discrete representation of the densities. However, as discussed in Sec. IV B, the computation time of the algorithm is driven in large part by the size of M and thus there is a trade-off between error introduced by small M and the computation time needed for large M. As noted previously, the value of 125 used in these demonstrations was selected based on the observation that the standard deviations of the estimates did not diminish when doubling M to 250.

An additional source of variability includes experimental variability. For each of the following demonstrations we report sample means and sample standard deviations for both the parameter estimates and the RMS errors. As was mentioned previously, 40 independent experiments were performed for each demonstration; this value was experimentally determined to be large enough so the experimental variability was small relative to the other sources.

We note that the fundamental limit of estimation variability for each of the fixed parameters (e.g., A, L, D, and G) is given by the Cramér-Rao lower bound (CRLB). In other words, the CRLB defines the absolute minimum variability one could obtain by estimating a set of fixed parameters using a finite number of images N, given a choice of motion and observation models but independent of the estimation algorithm used. For the case of an isotropic diffusion with diffusion coefficient D and additive Gaussian localization uncertainty with variance σ^2, the corresponding CRLB was derived analytically in Ref. [8]. In general, however, the calculation of CRLB for models that take the form of (2) and (3) are much more difficult since the exact likelihood is unknown. One approach is to condition on a latent state, as was done in the derivation of the SMC-EM algorithm in Sec. II, but this becomes numerically intractable for a large number of images. Recognizing this difficulty, some attempts have been made to estimate the CRLB via Monte Carlo simulations, e.g., Ref. [42], but there currently exists no analytical approach for the models we consider in this work. Thus, given this difficulty, we leave the discussion of the SMC-EM algorithm's performance relative to CRLB for future work.

Since most conventional approaches assume the localization uncertainty is zero-mean, additive, white, and Gaussian, a common approach is to treat the statistics of this uncertainty as a fixed parameter and, as with the motion parameters, compare localization uncertainty to the CRLB. As was indicated in Sec. II, however, the localized position of the particle is described in the SMC-EM framework by the time-dependent posterior probability density function which takes into account the observed images. Since the CRLB is limited to *fixed parameters*, i.e., quantities that are not dependent on time and do not vary during the experiment, it is not an applicable measure for localization error in this setting. One approach that is similar to the CRLB and can handle time-varying parameters is the *posterior* Cramér-Rao lower bound (PCRLB). The PCRLB was first presented in Ref. [43] and later extended to the context of nonlinear filtering in Ref. [44]. Much like the CRLB, the calculation of the PCRLB for general models of the form (2) and (3) is computationally intractable, and, although approaches have been made to numerically approximate it, e.g., Ref. [45], there is no currently analytically tractable method for its calculation. As such, we leave discussion of the SMC-EM algorithm's localization performance relative to PCRLB for future work.

D. Results for demonstration 1: Two-dimensional diffusion

The goal of the first demonstration was to use the SMC-EM algorithm to estimate the diffusion coefficients and trajectory for a particle undergoing two-dimensional isotropic diffusion. For this demonstration, we analyzed the K image sequences from the first set of data that was described in Sec. IV A. In addition, we compared the SMC-EM results to a conventional method of first localizing the particle in each image with a nonlinear fit to a Gaussian function and then calculating the ML estimate of the diffusion coefficients using the method proposed in Ref. [8]; for brevity, we refer to this method by *GF-ML*.

The motion model was assumed to be of the form (16) in both x and y axes; for identification purposes, however, we assumed the diffusion was anisotropic and that the diffusion coefficients for each axis, namely D_x and D_y, needed to be identified. Moreover, the position of the particle in the z direction was assumed to be zero, its true value.

To initialize the SMC-EM algorithm for each of the image sequences, the initial diffusion coefficient estimates, $\hat{D}_{x,0}$ and $\hat{D}_{y,0}$, were randomly generated within an order of magnitude of their true value $D = 0.01\ \mu\text{m}^2/\text{s}$.

As described in the Appendix, the maximization step consisted of evaluating the parameter update equation (12) with respect to the motion model (16) to yield the parameter update equations (A8) for both x and y axes.

For only this demonstration, we assumed the PSF could be approximately modeled by the Gaussian function

$$F_{\text{PS}}(x,y,z) = \frac{1}{\Delta x \Delta y} \exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right); \tag{21}$$

it was shown in Ref. [46] that the parameters

$$\sigma_x = \frac{\sqrt{2}\lambda}{2\pi \text{NA}}, \quad \sigma_y = \frac{\sqrt{2}\lambda}{2\pi \text{NA}}, \tag{22}$$

yield an optimal approximation of the Debye model (19) in an appropriate sense. The parameters λ and NA were assumed known with certainty, as was the background noise N_{bgd} and the z axis position of the particle (the latter of which was assumed to be zero). Note that as described in Sec. IV A, the true data were generated with the Debye model (19).

The approximation (21) is not a requirement for use of the SMC-EM algorithm. In fact, it will be shown in the three subsequent demonstrations that the SMC-EM algorithm is not restricted to any particular PSF model and may incorporate those from other imaging modalities. The primary benefit of using (21) is that it is simple to calculate.

Recall that the true diffusion coefficient for both x and y axes was $0.01\ \mu\text{m}^2/\text{s}$. After 10 iterations of SMC-EM, the resulting diffusion coefficient estimates were $0.009 \pm 0.002\ \mu\text{m}^2/\text{s}$ in both the x and y axes; the GF-ML algorithm yielded identical results. Additionally, after 10 iterations of SMC-EM, the RMS localization errors were $0.013 \pm 0.001\ \mu\text{m}$ and $0.012 \pm 0.001\ \mu\text{m}$ in x and y, respectively; the GF-ML algorithm yielded $0.009 \pm 0.001\ \mu\text{m}$ in both the x and y axes. The resulting diffusion coefficient estimates as a function of EM iteration number are shown in Fig. 1 while the RMS localization errors are shown in Fig. 2. We note that convergence for the SMC-EM algorithm was relatively fast in this case and only needed two to three iterations to converge.

An illustration comparing the performance of the two algorithms (GF-ML and SMC-EM) on estimating the particle trajectory is given in Fig. 3. Although the two yielded similar parameter and localization estimates, the two methods make very different assumptions on the underlying motion and observation models. In particular, the GF-ML algorithm assumes there is no information to be gained by considering the time history of the location of the particle during the localization step. In addition, the SMC-EM algorithm is not necessarily restricted to any motion or imaging modality so long as it can be represented in the form (2) and (3); the GF-ML algorithm is only applicable to a diffusing particle with a PSF that is approximately Gaussian and localization uncertainty that is normally distributed. Moreover, the SMC-EM algorithm yields a full, albeit approximate, probability distribution of the particle position given the acquired images, whereas the GF-ML provides Gaussian statistics.

It is important to note that blur during the exposure period may significantly affect the accuracy of both localization and estimation, especially if there is no attempt to compensate for it. To demonstrate its effect, we repeated Demonstration 1 with $\delta t = \Delta t = 100$ ms (i.e., full exposure during the imaging period). The SMC-EM method resulted in diffusion coefficient estimates of $0.007 \pm 0.001\ \mu\text{m/s}$ and RMS errors of $0.026 \pm 0.002\ \mu\text{m}$ for both the x and y axes. The GF-ML method, however, resulted in $0.010 \pm 0.002\ \mu\text{m/s}$ and RMS errors of $0.025 \pm 0.002\ \mu\text{m}$ for both the x and y axes. We note that although the localization errors were nearly equivalent for both methods, the GF-ML method resulted in a more accurate diffusion coefficient estimate. This is because the parameter estimation algorithm used in the GF-ML method

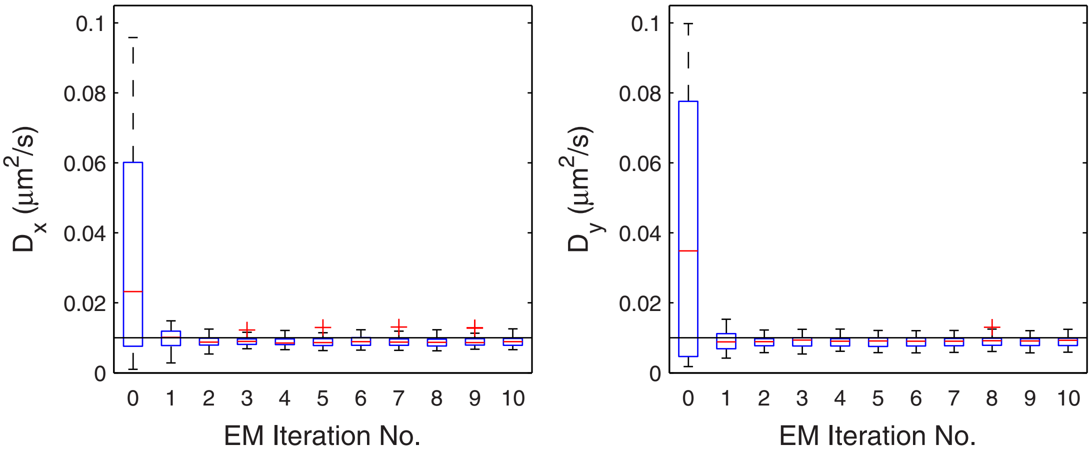

FIG. 1. (Color online) Box plots of the estimated diffusion coefficients as a function of the number of iterations of the SMC-EM algorithm for the first demonstration (2D diffusion, Sec. IV D). The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. The true values of the diffusion parameters, indicated by the solid black lines, were $D_x = D_y = 0.01\ \ \mu\text{m}^2/\text{s}$. At the 10th iteration of the SMC-EM algorithm, the estimated parameters were $\hat{D}_x = \hat{D}_y = 0.009 \pm 0.002\ \mu\text{m}^2/\text{s}$.

can compensate for the expected blur that occurs during the imaging process and that corrupts the position estimate (details on how this is done can be found in Ref. [8]).

Both the GF-ML and SMC-EM algorithms possess unique features and have their time and place for proper use. For example, when long exposure times are required (and blur contributes significantly to the localization uncertainty), the GF-ML algorithm may yield more accurate estimates. In contrast, when the experimentalist observes significant deviation from a standard diffusion and requires a more complicated motion and/or observation models, the SMC-EM algorithm may yield superior results.

E. Results for demonstration 2: Axially confined diffusion

The second demonstration was an extension of the first; specifically, we considered an isotropic diffusion where the axial motion was confined to the interval $[-L/2, L/2]$. For both the expectation and maximization steps, the motion models in the x and y axes were assumed to be of the form of (16), whereas the motion model for the z axis was assumed to be of the form of (17). For the purposes of estimation, the diffusion was assumed to be independent and anisotropic in all three axes. The SMC-EM algorithm was used to estimate the four motion parameters, namely the confinement length L and the three diffusion coefficients D_x, D_y, and D_z.

The expectation step for this demonstration was performed similarly to the first. That is, we used the SIR and FFBS algorithms (described in the Appendix) to calculate SMC approximations to the joint posterior density. The observation model was assumed to be identical to the one presented in Sec. III A. However, unlike the previous demonstration which used a Gaussian approximation (21) for its PSF model, we chose to implement the full Debye model (19) instead.

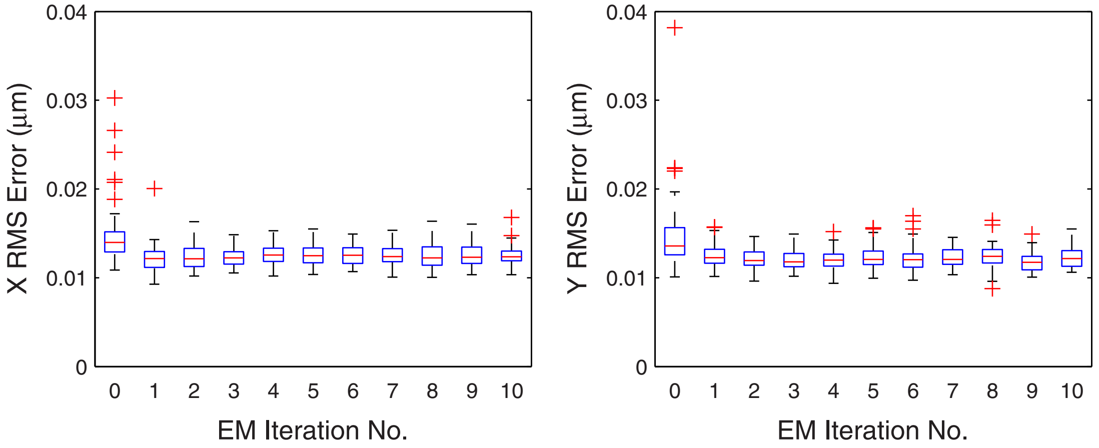

FIG. 2. (Color online) Box plots of the root mean square (RMS) localization error as a function of the number of iterations of the SMC-EM algorithm for the first demonstration (2D diffusion, Sec. IV D). The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. At the 10th iteration of the SMC-EM algorithm, the RMS errors of the means of the estimated posterior distributions were $0.013 \pm 0.001\ \mu$m and $0.012 \pm 0.001\ \mu$m in x and y, respectively.

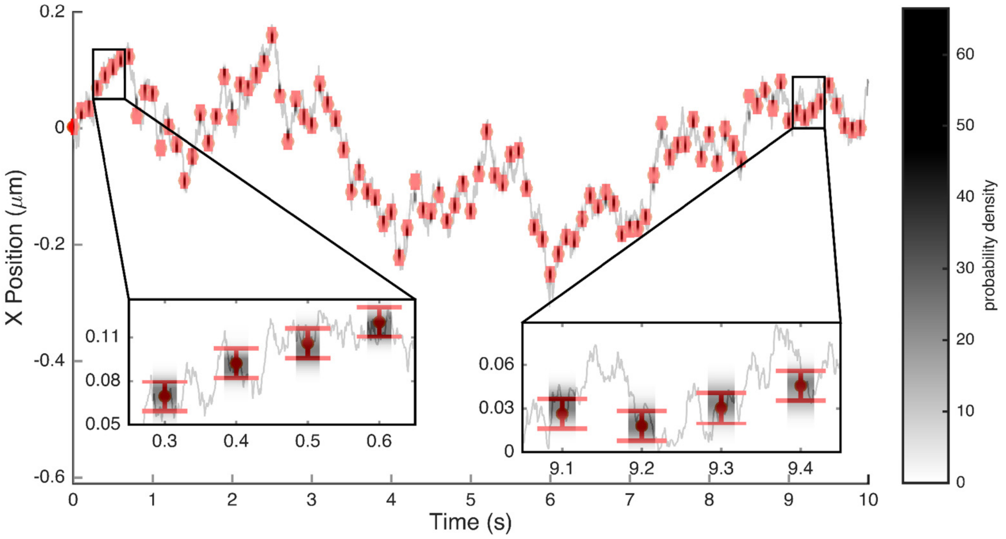

FIG. 3. (Color online) This figure illustrates a typical trajectory from the two-dimensional diffusion in the first demonstration (2D diffusion, Sec. IV D); only the x axis results are shown as the y axis results were similar in appearance. The random walk shown in gray represents the true position of the particle over time. The red dots indicate the Gaussian fit (GF) estimates and are positioned at the center of the red error bars which represent the 3σ uncertainties as determined via maximum likelihood estimation [8]. The shaded regions represent the approximate posterior probability densities for the particle positions after 10 iterations of the SMC-EM algorithm. Note that both the SMC-EM and GF estimates occur only every 100 ms while the true trajectory is defined (essentially) continuously. Although the SMC-EM algorithm provides a discrete Monte Carlo approximation of the posterior density, this figure shows a smoothed, continuous approximation calculated via Gaussian kernel density estimation.

This choice was made for two reasons; first, because no accurate Gaussian approximation exists for three-dimensional wide-field PSFs [46] and, second, to highlight the fact that complicated PSF models can be incorporated into the SMC-EM algorithm. We assumed all PSF parameters, as well as the gain parameter G and the background noise N_{bgd}, were known with certainty. We note that SMC-EM is capable of also estimating parameters describing the observation model; this is highlighted in the fourth demonstration.

The maximization step was also performed in a similar manner to the first demonstration. In particular, both the x and y diffusion coefficient estimates are of the form (A8). In contrast, the z-axis diffusion coefficient and confinement length estimates take a different form; see Appendix A 2 b for details.

Similarly to the first demonstration, the three initial diffusion coefficient estimates were randomly selected from within an order of magnitude of their true values $D = 0.01\ \mu\text{m}^2/\text{s}$. The initial length estimate $\hat{L}$ was set to 1.0 μm for every image sequence. The true value was 500 nm.

After 10 iterations of SMC-EM, the resulting diffusion coefficient estimates were $0.010 \pm 0.001\ \mu\text{m}^2/\text{s}$, $0.009 \pm 0.001 \mu\text{m}^2/\text{s}$, and $0.009 \pm 0.004\ \mu\text{m}^2/\text{s}$ in the x, y, and z axes, respectively, and the confinement length estimate was $0.53 \pm 0.05\ \mu$m. The RMS localization errors were $0.013 \pm 0.001\ \mu$m in both in x and y and $0.14 \pm 0.05\ \mu$m in z. The resulting parameter estimates and RMS localization errors as a function of EM iteration number are shown in Fig. 4 and 5.

Note that the error in localization along the z axis was an order of magnitude worse than in x and y; further, as seen in Fig. 5, this localization error did not show significant improvement over the initial condition. This reflects the fact that there is a large amount of localization uncertainty in z since the two-dimensional image of the symmetric three-dimensional PSF provides only limited information of the particle's axial position. This is due in part to the fact that the implemented PSF (19) has an effective axial slope that is much smaller than the longitudinal slope. Perhaps even more important, however, is that the axial symmetry of the PSF causes "crossing" errors about the particle center. Due to the symmetry of the PSF, the measurements cannot distinguish between positions above the focal plane and positions below the focal plane and as a result, trajectories that are mirror reflected through the focal plane are equally likely. In general, in the limit of very large M, the SMC approximation to the posterior density can maintain a bimodal distribution reflecting these two possibilities. Given a finite (and small) M, however, the estimator tends to lock the distribution onto one side of focal plane, chosen essentially at random. (This effect is essentially the same as what is known in the SMC literature as *sample impoverishment*. See, e.g., Ref. [40].) Unlike point estimates based on a single frame, however, the SMC-EM approach uses the entire set of frames in conjunction with the motion model to produce position estimates. As a result the expected location of the particle will not hop from one side of the plane to the other. This is illustrated in Fig. 6, which shows a typical z trajectory and the posterior density estimates; the figure shows that the posterior sometimes follows the true trajectory and sometimes is reflected through the focal plane at zero. In either case, however, the dynamics are well captured and as a result the estimates of the z diffusion coefficient and confinement length were still both quite accurate. The uncertainty did affect the convergence rate, however. As seen in Fig. 4, D_z took nearly all 10 iterations of SMC-EM to converge while D_x and D_y both converged within two iterations.

This localization problem can be remedied in the experimental setting by using an asymmetric PSF, such as those induced by astigmatism; since the SMC-EM algorithm makes

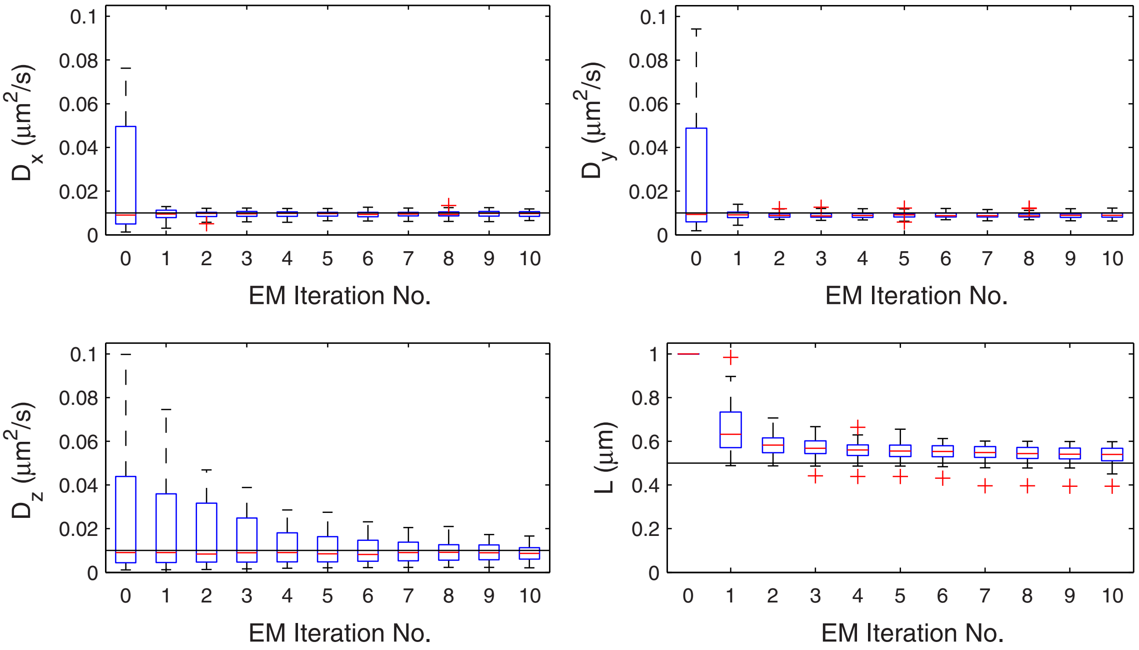

FIG. 4. (Color online) Box plots of the estimated parameters as a function of the number of iterations of the SMC-EM algorithm for the second demonstration (confined diffusion, Sec. IV E). In this case, the z position of the diffusing particle was confined to the interval $[-L/2,L/2]$; the diffusion coefficients D_x,D_y,D_z and the confinement length L were assumed unknown. The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. The true values, indicated by the solid black lines, were $D_x = D_y = D_z = 0.01\ \mu\text{m}^2/\text{s}$ and $L = 0.5\ \mu$m. After the 10th iteration, the estimates were $\hat{D}_x = 0.010 \pm 0.001\ \mu\text{m}^2/\text{s}$, $\hat{D}_y = 0.009 \pm 0.001\ \mu\text{m}^2/\text{s}$, $\hat{D}_z = 0.009 \pm 0.004\ \mu\text{m}^2/\text{s}$, and $\hat{L} = 0.53 \pm 0.04\ \mu$m.

no assumption on the form of the PSF, such experimental settings can still be analyzed with the SMC-EM method. Alternatively, one could recognize that the PSF does not yield information about the sign of the z position relative to the focal plane and use an estimator that gives only the distance from that plane. For example, one could replace the arithmetic mean (20) with the absolute value of the arithmetic mean. Doing so yields an RMS error (relative to the absolute value of the true z position) of $0.048 \pm 0.007\ \mu$m, a significant improvement over the 0.14-μm error for the signed case; these values are indicated in Fig. 5 in the rightmost figure by boxes in blue.

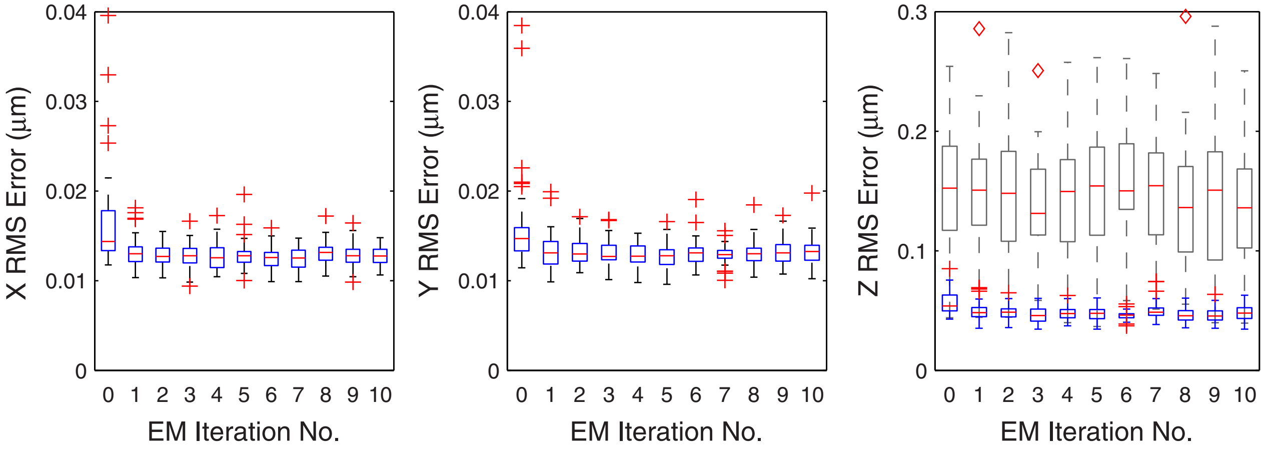

FIG. 5. (Color online) Box plots of the root mean square (RMS) localization error as a function of the number of iterations of the EM algorithm for the second demonstration (confined diffusion, Sec. IV E). The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. After the 10th iteration, the RMS errors of the means of the estimated posterior distributions were $0.013 \pm 0.001\ \mu$m in both x and y. The raw RMS errors for z are indicated in the rightmost figure as gray boxes with outliers denoted by red diamonds, whereas the RMS errors for the absolute value of z are indicated by blue boxes with outliers denoted by red + symbols. Note that the RMS errors for the raw z values are significantly larger than those for the absolute value of z. After the 10th iteration, the raw RMS z errors were $0.14 \pm 0.05\ \mu$m and the RMS $|z|$ errors were $0.048 \pm 0.007\ \mu$m.

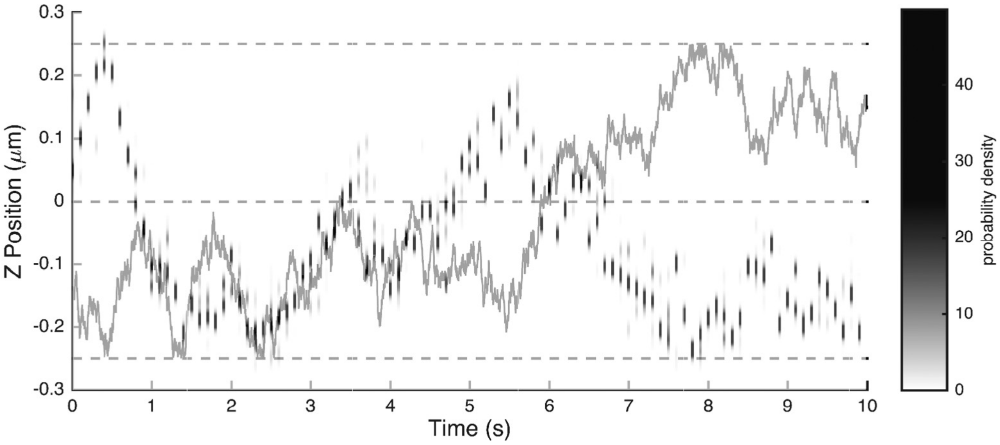

FIG. 6. A typical z trajectory from the second demonstration (confined diffusion, Sec. IV E). Here the true trajectory is shown in gray and the posterior density estimates after 10 iterations of the SMC-EM algorithm are shown with values according to the color bar. Note that estimates occur only every 100 ms while the true trajectory is defined (essentially) continuously. The dashed lines indicate the focal plane (at $z = 0$) and the bounds of the channel ($z = \pm 0.25$ μm). From this figure, it is clear that the PSF's symmetry about the focal point at $z = 0$ creates potential localization error in which the trajectory is essentially reflected through the focal plane (for example, from time 0 s to approximately 1 s and again from approximately 4.5 s through the end at 10 s). We note that this phenomenon is solely due to the PSFs symmetry about the focal point and is exhibited by all localization algorithms; PSFs that are asymmetric will not produce this behavior. Use of the motion model and the entire set of data prevent the estimates from randomly changing their sign relative to the focal plane and thus the motion model parameters can still be effectively estimated.

F. Results for demonstration 3: Three-dimensional tether

For the third demonstration, we considered the case where the fluorescent particle was elastically tethered to a fixed point. We assumed that, in all three axes, the motion was derived from the Ornstein-Uhlenbeck model (18). Although the motion in each axis was assumed independent, each axes' motion depended on the two to-be-estimated parameters, A and D.

The implementation of the SMC-EM algorithm for this demonstration was nearly identical to that of the second demonstration. Once again, the expectation step implemented both the SIR and FFBS algorithms (detailed in the Appendix) to estimate the joint posterior density. The observation model was identical to the second demonstration; that is, the full Debye model (19) was used and all parameters were assumed known with certainty. The maximization step used the formulas (A11) and (A14) to update the parameter estimates A and D, respectively. The initial parameter estimates $\hat{A}_0$ and $\hat{D}_0$ were randomly generated within an order of magnitude of their true values, 1.0 s^{-1} and 0.01 μm^2/s, respectively.

After 10 iterations of SMC-EM, the resulting stiffness coefficient was 1.0 ± 0.3 s^{-1} and the resulting diffusion coefficient was 0.009 ± 0.00 μm^2/s. The resulting RMS localization errors were 0.012 ± 0.001 μm in both x and y and 0.12 ± 0.05 μm in z. The resulting parameter estimates and RMS localization errors as a function of EM iteration number are shown in Figs. 7 and 8.

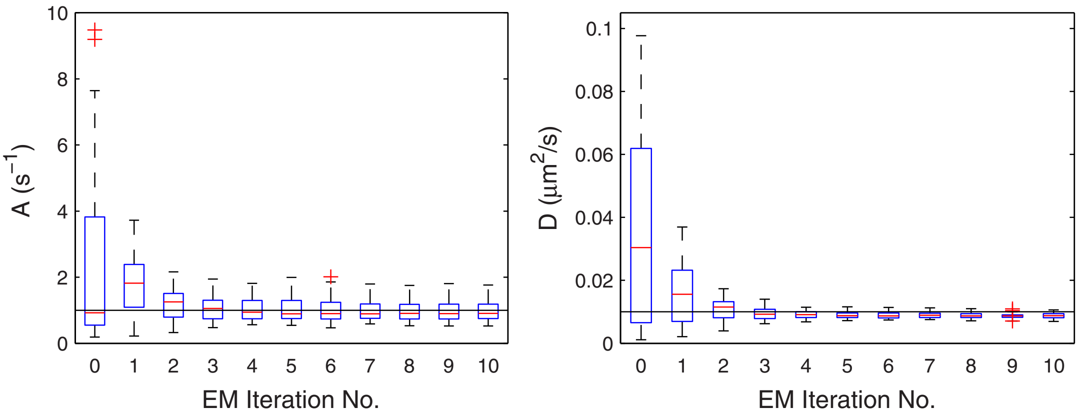

FIG. 7. (Color online) Box plots of the estimated parameters as a function of the number of iterations of the SMC-EM algorithm for the third demonstration (3D tether, Sec. IV F). In this case, the particle was assumed to be elastically tethered to a known location; the stiffness coefficient A and the diffusion coefficient D were assumed unknown. The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. The true values, indicated by the solid black lines, were $A = 1.0$ s^{-1} and $D = 0.01$ μm^2/s. After the 10th iteration, the estimated parameters were $\hat{A} = 1.0 \pm 0.3$ s^{-1} and $\hat{D} = 0.009 \pm 0.001$ μm/s.

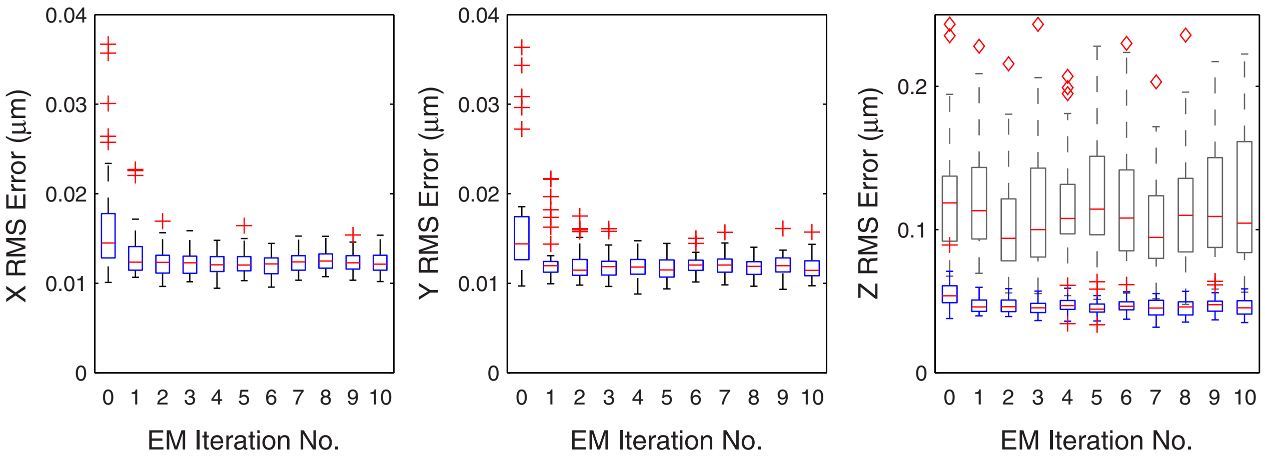

FIG. 8. (Color online) Box plots of the root mean square (RMS) localization error as a function of the number of iterations of the SMC-EM algorithm for the third demonstration (3D tether, Sec. IV F). The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. After the 10th iteration, the RMS errors of the means of the estimated posterior distributions were 0.012 ± 0.001 μm in both x and y. The raw RMS errors for z are indicated in the rightmost figure as gray boxes with outliers denoted by red diamonds, whereas the RMS errors for the absolute value of z are indicated by blue boxes with outliers denoted by red + symbols. Note that the RMS errors for the raw z values are significantly larger than those for the absolute value of z. After the tenth iteration, the raw RMS z errors were 0.12 ± 0.05 μm and the RMS $|z|$ errors were 0.046 ± 0.006 μm.

The results from the third demonstration were similar in character to that of the second. In particular, we once again see significant axial localization uncertainty due to properties of the PSF, though using the absolute value as before yielded a much better error of 0.046 ± 0.006 μm. In addition, the estimated parameters require four or five iterations to converge instead of the expected two or three. This slower convergence rate is due to the parameters' dependence on the z position; had we decoupled x and y from z and used four parameters instead, i.e., A_{xy}, A_z, D_{xy}, D_z, the convergence rates would have been faster in x and y.

G. Results for demonstration 4: Three-dimensional tether with unknown G

Last, the fourth demonstration used exactly the same data as in the third demonstration but estimated one additional parameter. Unlike the third demonstration, which assumed complete knowledge of the observation model, the fourth demonstration assumed that the gain G must be estimated. Relative to the third demonstration, the only major algorithmic change this requires is the additional calculation of the estimate of G during the maximization step; the parameter estimate

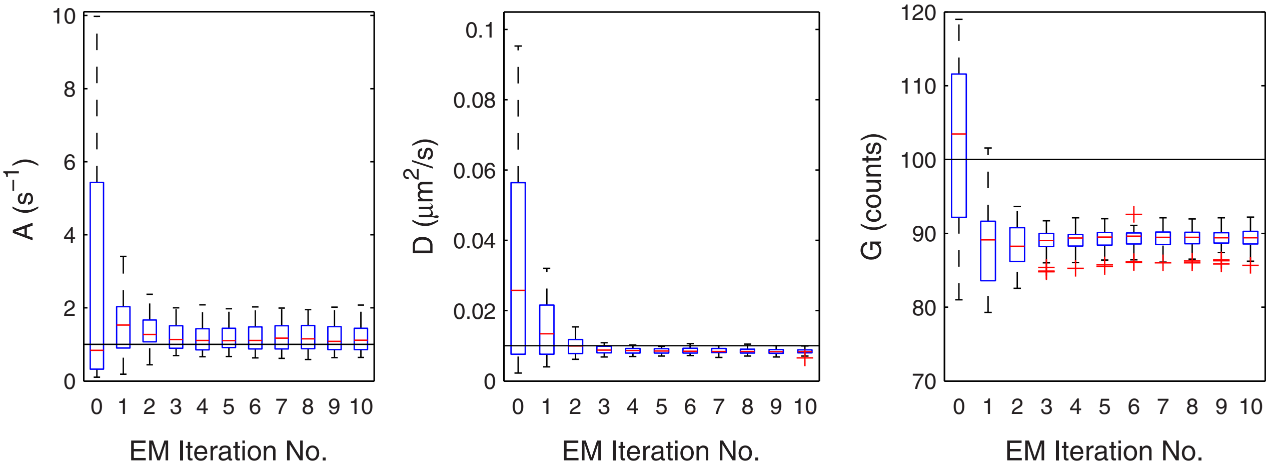

FIG. 9. (Color online) Box plots of the estimated parameters as a function of the number of iterations of the SMC-EM algorithm for the fourth demonstration (3D tether with unknown G, Sec. IV G). In this case, the particle was assumed to be elastically tethered to a known location; the stiffness coefficient A, the diffusion coefficient D, and the peak fluorescence intensity G were assumed unknown. The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. The true values, indicated by the solid black lines, were $A = 1.0$ s^{-1}, $D = 0.01$ μm^2/s, and $G = 100$ counts. The estimates after the 10th iteration were $\hat{A} = 1.2 \pm 0.4$ s^{-1}, $\hat{D} = 0.009 \pm 0.001$ μm^2/s, and $\hat{G} = 89 \pm 1$cts.

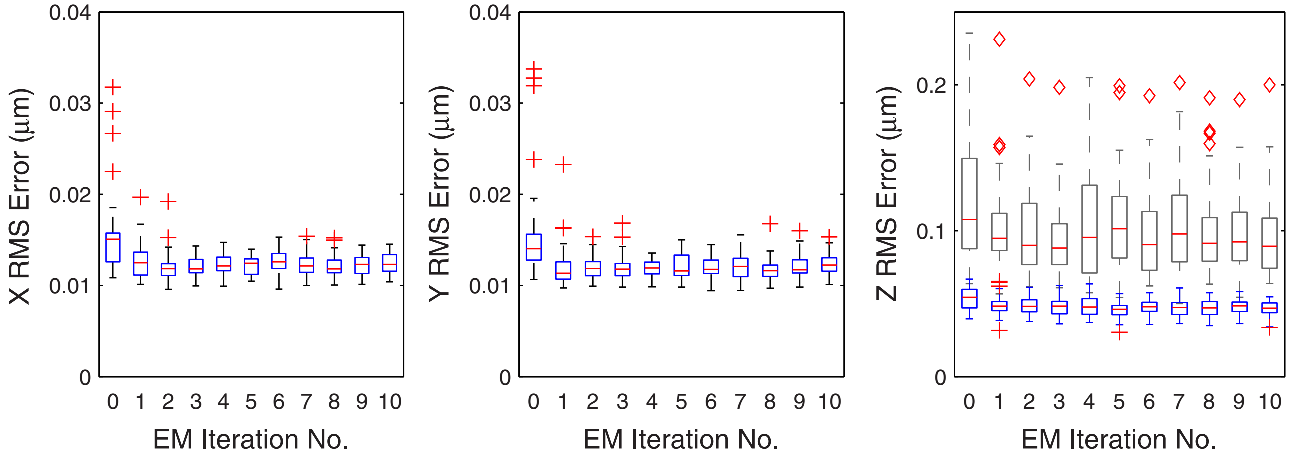

FIG. 10. (Color online) Box plots of the root mean square (RMS) localization error as a function of the number of iterations of the SMC-EM algorithm for the fourth demonstration (3D tether with unknown G, Sec. IV G). The edges of the box represent the first and third quartiles, the red line inside the box is the median, the vertical dashed line indicates the bounds for data within 1.5 times the interquartile range, and the red + symbols are data points outside this range. After the 10th iteration, the RMS errors of the means of the estimated posterior distributions were 0.012 ± 0.001 μm in both x and y. The raw RMS errors for z are indicated in the rightmost figure as gray boxes with outliers denoted by red diamonds, whereas the RMS errors for the absolute value of z are indicated by blue boxes with outliers denoted by red + symbols. Note that the RMS errors for the raw z values are significantly larger than those for the absolute value of z. After the 10th iteration, the raw RMS z errors were 0.10 ± 0.03 μm and the RMS $|z|$ errors were 0.047 ± 0.006 μm.

equation for G is shown in (A17); further details are given in the Appendix. The initial estimate for G was randomly generated within 20% of the true value. In practice the initial estimate would be guided by the peak values in the data.

The true diffusion coefficient was 0.01 $\mu\text{m}^2/\text{s}$, the true stiffness coefficient was 1.0 s^{-1}, and the true gain was 100 counts. After 10 iterations of SMC-EM, the resulting estimated diffusion coefficient was 0.009 ± 0.001 $\mu\text{m}^2/\text{s}$, the resulting stiffness coefficient was 1.2 ± 0.3 s^{-1}, and the resulting gain was 89 ± 1 counts. The resulting RMS localization errors were 0.012 ± 0.001 μm in both x and y and 0.10 ± 0.03 μm in z. The error for the absolute value of z was 0.047 ± 0.006 μm. The parameter estimates and RMS localization errors as a function of EM iteration number are shown in Figs. 9 and 10.

Convergence for the fourth demonstration was very similar to the third. In particular, we see that the A, D, and G parameters all required four or five iterations to converge. As in the third demonstration, the estimate of D is very close to the true value. The estimate of A shows a larger error than previously and perhaps some bias (though the true parameter value is within one standard deviation) while there is a clear bias in the estimate of G. These errors likely arise from two sources. The first is again the axial uncertainty in the PSF. Since the intensity gain G is affected by the axial position of the particle, errors in that localization propagate into the gain estimate. The second is the fact that the same amount of data is being used to estimate a larger number of parameters; because the ML estimators are only *asymptotically* efficient, a larger number of images would reduce variance and bias.

V. CONCLUSION

In this work, we describe an inference method that accomplishes two tasks: First, given a set of noisy measurements, it constructs a discrete, Monte Carlo estimate of the posterior density of an unknown, time-varying state; second, it calculates maximum likelihood estimates of unknown parameters which characterize the the unknown state as it transitions over time. In addition, we develop a novel paradigm for which this method may be applied to the analysis of single-particle tracking data. To demonstrate the effectiveness of this method and paradigm, we applied it to several scenarios involving a fluorescent particle being observed by a wide-field fluorescence microscope. In particular, we showed that the method can be applied to a variety of motions of differing complexity; this includes simple two-dimensional isotropic diffusion as well as confined and tethered diffusions in three dimensions. In the 2D case, the performance of the algorithm was similar to the more standard approach of fitting the data in each image to a Gaussian point spread function to localize the particle, followed by the application of a maximum likelihood estimator to determine the diffusion coefficients. The three-dimensional examples demonstrated the ability of the technique to model a variety of motion and observation models and showed good performance.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation through Grants No. CMMI-0845742 and No. DBI-1352729. The first author was supported in part by an NSF GK-12 fellowship.

APPENDIX: IMPLEMENTATION OF SMC-EM

The SMC-EM algorithm described in Sec. II alternates between two distinct steps. The result of the first step, i.e., the *expectation step*, is an approximation for the joint posterior density of the state (i.e., particle location at each time step) given the observations (i.e., images). The outcome of the second step, i.e., the *maximization step*, is an approximation to the maximum likelihood (ML) estimate of an unknown parameter θ describing the motion and observation models (2) and (3). By alternating between these two steps, the outcomes from each of them, specifically the approximate posterior densities and ML estimates, stochastically converge to a local optima.

In this Appendix, we describe how the SMC-EM algorithm was implemented for each of the four demonstrations described in Sec. IV.

1. The expectation step

The expectation step requires the evaluation of $\hat{Q}(\theta,\hat{\theta}_e)$ in (10). Recall from Sec. II that (10) is actually a Monte Carlo approximation of $Q(\theta,\hat{\theta}_e)$ in (7) given the SMC approximation (9) of the posterior densities. Thus, the evaluation of the three quantities $\hat{I}_1,\hat{I}_2,\hat{I}_3$ in (11) requires M particle location estimates $(x^i_{k|N,e},y^i_{k|N,e},z^i_{k|N,e})$ for each of the N time steps. In addition, these estimates must be weighted according to their ability to approximate both the individual and sequentially pairwise joint posterior densities; the weights $w^i_{k|N}$ and $w^{ij}_{k|N}$ respectively accomplish this.

The notation used throughout this work is commonly used in the literature for state estimation of dynamical systems. To be explicit, the object $x^i_{k|m}$ defines the ith Monte Carlo estimate for the x position at time step k taking into account all observations (i.e., images) up until the mth. To denote a value that depends on the eth EM iteration, the subscript e is appended. In addition, when the Monte Carlo estimates are coupled across two points in the sequence, such as in the weight $w^{ij}_{k|N}$, the superscript includes two indices, in this case i and j. Last, to simplify notation, we use a capital X to denote the position of the particle in three dimensions, for example, $\tilde{X}^i_{k|N,e} \triangleq (\tilde{x}^i_{k|N,e},\tilde{y}^i_{k|N,e},\tilde{z}^i_{k|N,e})$.

In this subsection, we describe how these estimates and weights were calculated for the demonstrations described in Sec. IV. Specifically, we use the sampling importance resampling algorithm in conjunction with the FFBS algorithm to calculate the position estimates and their weights.

a. Sampling importance resampling

To calculate the Monte Carlo position estimates and their weights, we first employ the SIR algorithm. SIR is just one example of an SMC filter and was chosen because it is relatively easy to implement. For further information on other algorithms and their relative merits, see, e.g., Ref. [40].

A standard implementation of the SIR algorithm outputs a Monte Carlo approximation to the posterior density $p_{\hat{\theta}_e}(X_k|I_{1,1},\ldots,I_{P,1},I_{1,2},\ldots,I_{P,2},\ldots,I_{P,k})$. Note that this density is with respect to all images up until the *current* time step k; in the SMC and particle filtering literature, this is also known as the *filtered* posterior density. Similarly to the SMC approximation in (9), the SIR algorithm approximates the density by a sum of M-weighted δ functions. However, unlike the posterior density in (9) which weighted the ith estimate at time step k by $w^i_{k|N}$, here we denote the ith weight at time step k by $w^i_{k|k}$. In other words, the former weight takes into account all the images at every time step, whereas the latter only takes into account the images up until its respective time step.

To calculate the aforementioned posterior density, the SIR algorithm applies both importance sampling and statistical bootstrapping (i.e., resampling) at each time step. In contrast with a batch algorithm, SIR is recursive, meaning that the SMC approximation at time step k need only be calculated from the image acquired at time step k and the output of SIR at the $(k-1)$-th iteration. This presents a considerable improvement in computational complexity relative to a batch implementation; the recursive implementation of SIR has a computational complexity that scales as NM, that is, as the product of the number of time steps (images) and the number of discrete estimates.

At a given time index k, the SIR algorithm first uses importance sampling to generate a set of weighted position estimates. For a standard implementation of the SIR algorithm, this requires a draw of M independent and identically distributed candidate position estimates according to one of two probability densities. At the initial time index $k=1$, the candidate positions are drawn according to

$$\left(\tilde{X}^i_{1|1}\right) \sim p^{\text{Init}}_{\hat{\theta}_e}(X_1). \tag{A1}$$

where the probability density is given by the initial distribution (1). If, instead, $k>1$, then the candidates are drawn according to

$$\left(\tilde{X}^i_{k|k}\right) \sim p^{\text{Mot}}_{\hat{\theta}_e}\left(\tilde{X}_{k|k}\middle|X^i_{k-1|k-1}\right), \tag{A2}$$

where the probability density is given by the *motion* model (2). In both cases, these candidates, denoted by the tilde, are then weighted according to

$$w^i_{k|k} \propto p^{\text{Obs}}_{\hat{\theta}_e}\left(I_{1,k},\ldots,I_{P,k}\middle|\tilde{X}^i_{k|k}\right), \tag{A3}$$

where the probability density is given by the *observation* model (3). In the scenario presented in Sec. III, the observation model is given by a Poisson distribution (due to the presence of shot noise). In addition, the intensity measured at each pixel is independent when conditioned on the particle position, so (A3) simplifies to

$$w^i_{k|k} \propto \prod_{p=1}^{P} p^{\text{Obs}}_{\hat{\theta}_e}\left(I_{p,k}\middle|X^i_{k|k}\right). \tag{A4}$$

The weights are normalized so their sum with respect to i is unity. This completes the importance sampling process at time step k.

Directly after the importance sampling process at time step k, the SIR algorithm generates a new collection of position estimates from the estimates that were formed during the importance sampling process. Specifically, this resampling step draws a new collection of M realizations of $p_{\hat{\theta}_e}(X_k|I_{1,1},\ldots,I_{P,k})$ via bootstrapping from the previously generated collection of candidates. There are many methods for resampling from a discrete collection; again, see Ref. [40]. Due to its simplicity in implementation, we chose multinomial resampling. Thus, we draw with replacement $j=1,\ldots,M$ new samples X^j_k according to

$$\mathbb{P}\left(X^j_{k|k}=\tilde{X}^i_{k|k}\right)=w^i_{k|k}. \tag{A5}$$

The resulting collection of positions forms a set of independent and identically distributed realizations of

$p_{\hat{\theta}_e}(X_k|I_{1,1},\ldots,I_{P,k})$, and, consequently, the weights are set to $1/M$. This completes the resampling step for time step k.

This whole process, namely the generation of candidates in (A2), the calculation of the weights in (A3), and the resampling of the candidates in (A5), is repeated for each time step k until $k=N$. After the algorithm calculates the $k=N$ time step, the SIR algorithm terminates.

b. Forward-filtering backward smoothing

The SIR algorithm produces a collection of M position estimates $X^i_{k|k}$ each weighted by $w^i_{k|k}$, a measure that describes how well they approximate the posterior density that takes into account the observations up until the kth time step (i.e., the filtered posterior density). The SMC-EM algorithm, however, cannot directly use the filtered posterior density in (11) as it instead requires a Monte Carlo estimate for the posterior density that takes into account *all* N observations. In the state estimation literature, this is also known as the *smoothed* posterior density. To calculate the Monte Carlo approximation to the smoothed posterior density, we leverage the filtered posterior density from the implementation of the SIR algorithm and apply the FFBS algorithm.

To calculate the smoothed posterior density, we implemented the FFBS algorithm as described in Ref. [41]. An implementation of this algorithm uses the results from the SIR algorithm and operates backward in time to calculate the smoothed posterior density.

At the initial time step, here $k=N$, the smoothed posterior density is equivalent to the filtered posterior density. Thus, the position estimates and weights at $k=N$ are equivalent.

At the remaining time steps, $k=(N-1),\ldots,1$, the formulas

$$w^i_{k|N,e} = w^i_{k|k,e}\sum_{m=1}^{M} w^m_{k+1|N,e}\frac{p^{\text{Mot}}_{\hat{\theta}_e}\left(X^m_{k+1|k+1,e}\middle|X^i_{k|k,e}\right)}{v^m_k}, \tag{A6a}$$

$$v^m_k \triangleq \sum_{j=1}^{M} w^j_{k|k,e}\,p^{\text{Mot}}_{\hat{\theta}_e}\left(X^m_{k+1|k+1,e}\middle|X^j_{k|k,e}\right), \tag{A6b}$$

are used to calculate the smoothed weights. Note that the quantity (A6a) only depends on the motion model and not the observation model. Resampling was not performed during the smoothing process; thus, $X^i_{k|N,e}=X^i_{k|k,e}$ for all $k=1,\ldots,N$ and $i=1,\ldots,M$.

In addition, the SMC representation of the sequentially pairwise joint density was originally presented in Lemma 6.1 in Ref. [13] and is given by the formula

$$w^{ij}_{k|N} = \frac{w^i_{k|k}w^j_{k+1|N}\,p^{\text{Mot}}_{\hat{\theta}_e}\left(X^j_{k+1|k+1}\middle|X^i_{k|k}\right)}{\sum_{l=1}^{M} w^l_{k|k}\,p^{\text{Mot}}_{\hat{\theta}_e}\left(X^j_{k+1|k+1}\middle|X^l_{k|k}\right)}. \tag{A7}$$

for all $k=1,\ldots,N$, $i=1,\ldots,M$, and $j=1,\ldots,M$.

In contrast to the SIR algorithm, the computational complexity of the FFBS algorithm scales as NM^2. Since the complexity grows with the square of the number of estimates M, the user of the algorithm must choose M so the posterior density is adequately approximated while also taking into account the computational load. Note that in the context of the SMC-EM algorithm, the combination of SIR and FFBS must be performed for each iteration, and the overall complexity of the SMC-EM algorithm therefore scales as ENM^2, where E is the maximum number of iterations. In this work, the value for M was chosen by running the SMC-EM algorithm with a small initial value for M and increasing it until the output did not significantly differ. Here $M=250$ yielded negligible benefit over $M=125$ in terms of the bias and variance of the parameter estimates. If a better approximation for the posterior density were desired, one could implement the SMC-EM algorithm with a sufficiently small number of estimates and then rerun the SIR and FFBS algorithms (or any other SMC method) with a much larger value for M using the (fixed) values of the motion and observation parameters.

2. The maximization step

The maximization step for the SMC-EM algorithm requires the evaluation of (12); that is, the updated parameters are given by the argument θ which maximizes $\hat{Q}(\theta,\hat{\theta}_e)$. In this subsection, we list the formulas that were used for each of the four demonstrations describe in Sec. IV.

a. Demonstration 1: Two-dimensional diffusion

In the first demonstration, the motion was assumed to be derived from an anisotropic two-dimensional diffusion (16) with unknown diffusion coefficients D_x and D_y. We also assumed that the observation model did not contain any parameters that needed to be estimated. Inserting the motion model (16) into the update equation for $\hat{Q}(\theta,\hat{\theta}_e)$, (12), together with the SMC approximation to $Q(\theta,\hat{\theta}_e)$ given by (10) and (11), and then setting the derivative with respect to θ to zero yields the parameter update equations

$$\hat{D}_{x,e+1} = \frac{1}{2N\Delta t}\left(D^a_{x,e}+D^b_{x,e}\right), \tag{A8a}$$

$$D^a_{x,e} \triangleq \sum_{i=1}^{M} w^i_{1|N,e}\left(x^i_{1|N,e}\right)^2, \tag{A8b}$$

$$D^b_{x,e} \triangleq \sum_{k=1}^{N-1}\sum_{i=1}^{M}\sum_{j=1}^{M} w^{ij}_{k|N,e}\left(x^j_{k+1|N,e}-x^i_{k|N,e}\right)^2, \tag{A8c}$$

for the x axis; the equations for the y axis are exactly the same with y replacing x. We note that the first term, denoted by the superscript a, represents the initial condition and the second term, denoted by the superscript b, represents the first-order difference between sequential estimates. If the diffusion were assumed to be *isotropic*, the resulting diffusion coefficient estimate would be the average of the x and y diffusion coefficient estimates.

Not surprisingly, the diffusion coefficient estimates essentially take the form of a weighted linear regression to the mean-square displacement, with the difference here being that weights and position estimates are formed via the SMC methods described in the previous subsection. In addition, the diffusion coefficients, position estimates, and weights are iteratively improved via the SMC-EM algorithm, unlike the standard method which does not iterate.

b. Demonstration 2: Axially confined diffusion

In the second demonstration, we assumed the motion was derived from a three-dimensional anisotropic diffusion with reflecting axial confinement $z_k \in [-L/2, L/2]$. The transition densities for the position of the particle in the x and y axes are given by (16), whereas for the z axis it is (17). Since the diffusion was assumed anisotropic, the update equations for the x and y diffusion coefficient estimates are the same as in the first demonstration (A8).

The update equation for the length of the confinement channel is found by inserting the motion model (17) into (12), using the SMC approximation (11), and then maximizing. This gives

$$\hat{L}_{e+1} = \max_{k,i} 2\left|z^i_{k|N,e}\right|. \tag{A9}$$

This implies that the best estimate for the length of the channel is the absolute maximum position estimate in the z axis. We further note that if, instead, the position were confined within $z_k \in [L_1, L_2]$ with $L_1 < L_2$, the same reasoning would hold; that is, L_1 would be the minimum position estimate and L_2 would be the maximum.

We note that the initial value for the length parameter estimate *must* be greater than its corresponding true value to guarantee convergence. To see this, consider the length parameter estimate $\hat{L}_e$. During the expectation step at iteration $e+1$, z position estimates generated according to (17) will be constrained to the interval $[-\hat{L}_e/2, \hat{L}_e/2]$. During the subsequent maximization step, $\hat{L}_{e+1}$ is set to be the absolute maximum of every z position estimate via (A9). From this, it follows that $\hat{L}_{e+1} \leqslant \hat{L}_e$. As such, in practice, the user must always set the initial length parameter estimate to a value that overestimates the true length.

Unfortunately, there is no analytical solution for the z-axis diffusion coefficient. Thus, (12) must be maximized numerically with respect to D_z. In this demonstration, however, D_z is the only parameter for which this is required and can therefore be calculated by solving a one-dimensional unconstrained optimization problem via, for example, the bisection method. For completeness, we describe how the bisection method may be used to calculate D_z.

The bisection method is a commonly used numerical algorithm for finding a root of a mathematical expression contained within a known upper and lower bound. The method can therefore be used to find local extrema through its application to the derivative of the expression. To find the maximizing value D_z at EM iteration e, this expression is

$$J(D_z) = \frac{\partial}{\partial D_z} \sum_{k=1}^{N-1} \sum_{i=1}^{M} \sum_{j=1}^{M} w^{ij}_{k|N,e} \log\left[p_{D_z}\left(z^{ij}_{k+1|N,e} \middle| z^{ij}_{k|N,e}\right)\right], \tag{A10}$$

where p_{D_z} is the confined probability density (17) as a function of the unknown diffusion coefficient D_z evaluated on the Monte Carlo estimate positions $z^{ij}_{k|N,e}$. Note that (17) is also a function of the confinement length L; since the optimal estimate for L may be calculated via (A9) without any knowledge of D_z, this value should be used in the evaluation of (A10). The bisection method iteratively calculates an optimal value for D_z by methodically searching for a candidate such that the expression (A10) is approximately zero. It begins by considering the closed interval $[D_{z,\text{Low}}, D_{z,\text{High}}]$ in which contains the maximizing value; in this work, we considered the interval $[0.1D_z, 10D_z]$ with D_z denoting the true parameter value. The method evaluates $J(D_z)$ at $D_{z,\text{Low}}$ and the midpoint, $D_{z,\text{Mid}} \triangleq (D_{z,\text{High}} + D_{z,\text{Low}})/2$. If the signs of these evaluations are equivalent, then $D_{z,\text{Low}}$ is replaced with $D_{z,\text{Mid}}$; otherwise, $D_{z,\text{High}}$ is replaced with $D_{z,\text{Mid}}$. This process is repeated until either $J(D_{z,\text{Mid}})$ yields zero or until $(D_{z,\text{High}} - D_{z,\text{Low}})/2$ is below some threshold (here, chosen to be 10^{-6}). The estimated value for D_z is equivalent to $D_{z,\text{Mid}}$.

c. Demonstration 3: Three-dimensional elastic tethering

In the third demonstration, we assumed the motion was derived from the three-dimensional Ornstein-Uhlenbeck process with the transition density given by (18). This motion model can represent, for example, a particle of negligible mass attached to a stiff, elastic tether. The motion was assumed to be isotropic so only two parameters, A and D, needed to be estimated.

The update equations for the two parameters A and D can be derived analytically by solving (12) with respect to the motion model (18). The parameter update equation for the stiffness coefficient A is given by

$$\hat{A}_{e+1} = -\frac{1}{\Delta t} \log\left(\frac{\alpha_e}{\beta_e}\right) \tag{A11}$$

with α_e and β_e defined by

$$\alpha_e = \sum_{k=1}^{N-1} \sum_{i=1}^{M} \sum_{j=1}^{M} w^{ij}_{k|N,e}\left(\alpha^{ij,a}_{k,e} + \alpha^{ij,b}_{k,e} + \alpha^{ij,c}_{k,e}\right), \tag{A12a}$$

$$\alpha^{ij,a}_{k,e} = x^j_{k+1|N,e} x^i_{k|N,e}, \tag{A12b}$$

$$\alpha^{ij,b}_{k,e} = y^j_{k+1|N,e} y^i_{k|N,e}, \tag{A12c}$$

$$\alpha^{ij,c}_{k,e} = z^j_{k+1|N,e} z^i_{k|N,e}, \tag{A12d}$$

$$\beta_e = \sum_{k=1}^{N-1} \sum_{i=1}^{M} \sum_{j=1}^{M} w^{i}_{k|N,e}\left(\beta^{i,a}_{k,e} + \beta^{i,b}_{k,e} + \beta^{ij,c}_{k,e}\right), \tag{A13a}$$

$$\beta^{i,a}_{k,e} = \left(x^i_{k|N,e}\right)^2, \tag{A13b}$$

$$\beta^{i,b}_{k,e} = \left(y^i_{k|N,e}\right)^2, \tag{A13c}$$

$$\beta^{i,c}_{k,e} = \left(z^i_{k|N,e}\right)^2. \tag{A13d}$$

The parameter update equation for the diffusion coefficient D is given by

$$\hat{D}_{e+1} = \frac{(\gamma_e + \delta_e)\hat{A}_{e+1}}{3N[1 - \exp(-2\hat{A}_{e+1}\Delta t)]} \tag{A14}$$

with γ_e and δ_e defined by

$$\gamma_e = \sum_{k=1}^{N-1} \sum_{i=1}^{M} \sum_{j=1}^{M} w^{ij}_{k|N,e}\left(\gamma^{ij,a}_{k,e} + \gamma^{ij,b}_{k,e} + \gamma^{ij,c}_{k,e}\right), \tag{A15a}$$

$$\gamma^{ij,a}_{k,e} = \left(x^j_{k+1|N,e} - x^i_{k|N,e}, e^{-2\Delta t \hat{A}_e}\right)^2, \tag{A15b}$$

$$\gamma_{k,e}^{ij,b} = \left(y_{k+1|N,e}^{j} - y_{k|N,e}^{i} e^{-2\Delta t \hat{A}_e}\right)^2, \tag{A15c}$$

$$\gamma_{k,e}^{ij,c} = \left(z_{k+1|N,e}^{j} - z_{k|N,e}^{i} e^{-2\Delta t \hat{A}_e}\right)^2, \tag{A15d}$$

$$\delta_e = \sum_{i=1}^{M} w_{1|N}^{i}\left(\delta_e^{i,a} + \delta_e^{i,b} + \delta_e^{i,c}\right), \tag{A16a}$$

$$\delta_e^{i,a} = \left(x_{1|N}^{i}\right)^2, \tag{A16b}$$

$$\delta_e^{i,b} = \left(y_{1|N}^{i}\right)^2, \tag{A16c}$$

$$\delta_e^{i,c} = \left(z_{1|N}^{i}\right)^2. \tag{A16d}$$

d. Demonstration 4: Three-dimensional tether with unknown PSF

In the fourth demonstration, we assume once again that the motion is derived from the three-dimensional Ornstein-Uhlenbeck process with transition density given by (18). However, unlike the third demonstration, in this demonstration we further assume that the gain G in (13) is unknown. Both the parameter update equations for the stiffness and diffusion coefficients A and D remain unchanged and are found in (A11) and (A14); the parameter estimate for the unknown gain G, however, must be calculated numerically by

$$\hat{G}_{e+1} = \arg\max_{G} \epsilon_e(G) \tag{A17}$$

with ϵ_e defined by

$$\epsilon_e(G) = \sum_{k=1}^{N}\sum_{i=1}^{M}\sum_{p=1}^{P} w_{k|N,e}^{i}\lambda_{p,k,e}^{i}(G)\left[\frac{I_{p,k}}{\lambda_{p,k,e}^{i}(G) + N_{\text{bgd}}} - 1\right]. \tag{A18}$$

The expected intensity for each pixel p is given by

$$\lambda_{p,k,e}^{i}(G) = \int_{x_{p,k}^{\min}}^{x_{p,k}^{\max}}\int_{y_{p,k}^{\min}}^{y_{p,k}^{\max}} G\, F_{\text{PS}}\left(x_{k|N,e}^{i} - \xi', y_{k|N,e}^{i} - \xi'', z_{k|N,e}^{i}\right) \times d\xi' d\xi'', \tag{A19}$$

which is analogous to how it was defined in (13) with the only difference being the dependence on both the EM iteration e and the particle index i. Since (A17) is twice differentiable, we used a Newton-like method (the tangent-hyperbolas method; see Ref. [47]) to find the resulting maximum value at each iteration.

[1] H. Geerts, M. De Brabander, R. Nuydens, S. Geuens, M. Moeremans, J. De Mey, and P. Hollenbeck, Nanovid tracking: a new automatic method for the study of mobility in living cells based on colloidal gold and video microscopy, Biophys. J. **52**, 775 (1987).

[2] J. Gelles, B. J. Schnapp, and M. P. Sheetz, Tracking kinesin-driven movements with nanometre-scale precision, Nature **331**, 450 (1988).

[3] F. Zhang, Z. Zheng, S.-L. Liu, W. Lu, Z. Zhang, C. Zhang, P. Zhou, Y. Zhang, G. Long, Z. He, D.-W. Pang, Q. Hu, and H. Wang, Self-biotinylation and site-specific double labeling of baculovirus using quantum dots for single-virus in-situ tracking, Biomaterials **34**, 7506 (2013).

[4] A. Yildiz, H. Park, D. Safer, Z. Yang, L. Q. Chen, P. R. Selvin, and H. L. Sweeney, Myosin vi steps via a hand-over-hand mechanism with its lever arm undergoing fluctuations when attached to actin, J. Biol. Chem. **279**, 37223 (2004).

[5] L. Cognet, C. Leduc, and B. Lounis, Advances in live-cell single-particle tracking and dynamic super-resolution imaging, Curr. Opin. Chem. Biol. **20**, 78 (2014).

[6] A. Kusumi, T. A. Tsunoyama, K. M. Hirosawa, R. S. Kasai, and T. K. Fujiwara, Tracking single molecules at work in living cells, Nat. Chem. Biol. **10**, 524 (2014).

[7] B. S. Schuster, L. M. Ensign, D. B. Allan, J. S. Suk, and J. Hanes, Particle tracking in drug and gene delivery research: State-of-the-art applications and methods, Adv. Drug Deliver. Rev. **91**, 70 (2015).

[8] X. Michalet and A. J. Berglund, Optimal diffusion coefficient estimation in single-particle tracking, Phys. Rev. E **85**, 061916 (2012).

[9] B. Shuang, C. Byers, L. Kisley, L. Wang, J. Zhao, H. Morimura, S. Link, and C. Landes, Improved analysis for determining diffusion coefficients from short, single-molecule trajectories with photoblinking, Langmuir **29**, 228 (2013).

[10] C. L. Vestergaard, P. C. Blainey, and H. Flyvbjerg, Optimal estimation of diffusion coefficients from single-particle trajectories, Phys. Rev. E **89**, 022726 (2014).

[11] M. J. Saxton and K. Jacobson, Single-particle tracking: applications to membrane dynamics, Annu. Rev. Bioph. Biom. **26**, 373 (1997).

[12] T. T. Ashley and S. B. Andersson, A sequential Monte Carlo framework for the system identification of jump Markov state space models, in *Proceedings of the 2014 American Control Conference (ACC)* (IEEE, New York, 2014), pp. 1144–1149.

[13] T. B. Schön, A. Wills, and B. Ninness, System identification of nonlinear state-space models, Automatica **47**, 39 (2011).

[14] R. E. Thompson, D. R. Larson, and W. W. Webb, Precise nanometer localization analysis for individual fluorescent probes, Biophys. J. **82**, 2775 (2002).

[15] M. K. Cheezum, W. F. Walker, and W. H. Guilford, Quantitative comparison of algorithms for tracking single fluorescent particles, Biophys. J. **81**, 2378 (2001).

[16] R. Parthasarathy, Rapid, accurate particle tracking by calculation of radial symmetry centers, Nat. Methods **9**, 724 (2012).

[17] S. B. Andersson, Localization of a fluorescent source without numerical fitting, Opt. Express **16**, 18714 (2008).

[18] N. Chenouard, I. Smal, F. De Chaumont, M. Maška, I. F. Sbalzarini, Y. Gong, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter *et al.*, Objective comparison of particle tracking methods, Nat. Methods **11**, 281 (2014).

[19] I. Smal, K. Draegestein, N. Galjart, W. Niessen, and E. Meijering, Particle filtering for multiple object tracking in dynamic fluorescence microscopy images: Application to microtubule growth analysis, IEEE T. Med. Imaging **27**, 789 (2008).

[20] M. J. Saxton, Single-particle tracking: The distribution of diffusion coefficients, Biophys. J. **72**, 1744 (1997).

[21] M. J. Saxton, Single-particle tracking: Effects of corrals, Biophys. J. **69**, 389 (1995).

[22] Y. Pawitan, *In All Likelihood: Statistical Modelling and Inference Using Likelihood* (Oxford University Press, Oxford, 2001).

[23] A. P. Dempster, N. M. Laird, and D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Roy. Stat. Soc. B Met. **39**, 1 (1977).

[24] H. E. Rauch, C. T. Striebel, and F. Tung, Maximum likelihood estimates of linear dynamic systems, AIAA J. **3**, 1445 (1965).

[25] A. Doucet and A. M. Johansen, A tutorial on particle filtering and smoothing: Fifteen years later, *The Oxford Handbook of Nonlinear Filtering*, edited by D. Crisan and B. Rozovskiĭ (Oxford University Press, New York, 2009), pp. 656–704.

[26] A. Doucet, N. de Freitas, and N. Gordon (eds.), *Sequential Monte Carlo Methods in Practice* (Springer, Berlin, 2001).

[27] M. Sanjeev Arulampalam, S. Maskell, N. Gordon, and T. Clapp, A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking, IEEE T. Signal Process. **50**, 174 (2002).

[28] S. Frisken-Gibson and F. Lanni, Diffraction by a circular aperture as a model for three-dimensional optical microscopy, J. Opt. Soc. Am. A **6**, 1357 (1989).

[29] 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, Proc. Natl. Acad. Sci. U.S.A. **106**, 2995 (2009).

[30] S. Jia, J. C. Vaughan, and X. Zhuang, Isotropic three-dimensional super-resolution imaging with a self-bending point spread function, Nat. Photon. **8**, 302 (2014).

[31] Y. Shechtman, S. J. Sahl, A. S. Backer, and W. E. Moerner, Optimal Point Spread Function Design for 3D Imaging, Phys. Rev. Lett. **113**, 133902 (2014).

[32] S. E. Moran, B. L. Ulich, W. P. Elkins, R. L. Strittmatter, and M. J. Deweert, Intensified ccd (iccd) dynamic range and noise performance, Proc. SPIE **3173**, 430 (1997).

[33] J. Chao, E. S. Ward, and R. J. Ober, Fisher information matrix for branching processes with application to electron-multiplying charge-coupled devices, Multidim. Syst. Sign. P. **23**, 349 (2012).

[34] H. Risken, *The Fokker-Planck Equation: Methods of Solution and Applications* (Springer, Berlin, 1996).

[35] H. S. Carslaw and J. C. Jaeger, *Conduction of Heat in Solids*, 2nd ed. (Clarendon Press, Oxford, 1959).

[36] P. C. Nelson, C. Zurla, D. Brogioli, J. F. Beausang, L. Finzi, and D. Dunlap, Tethered particle motion as a diagnostic of dna tether length, J. Phys. Chem. B **110**, 17260 (2006).

[37] J. F. Beausang, C. Zurla, L. Finzi, L. Sullivan, and P. C. Nelson, Elementary simulation of tethered brownian motion, Am. J. Phys. **75**, 520 (2007).

[38] M. Gu, *Advanced Optical Imaging Theory* (Springer, Berlin, 2000).

[39] H. V. Poor and O. Hadjiliadis, *Quickest Detection* (Cambridge University Press, Cambridge, 2008).

[40] O. Cappe, S. J. Godsill, and E. Moulines, An overview of existing methods and recent advances in sequential Monte Carlo, Proc. IEEE **95**, 899 (2007).

[41] F. Lindsten and T. B. Schön, Backward simulation methods for Monte Carlo statistical inference, Found. Trends Mach. Learn. **6**, 1 (2013).

[42] J. C. Spall, J. Comput. Graph. Stat. **14**, 889 (2005).

[43] H. Trees, *Detection, Estimation, and Modulation Theory* (Wiley, New York, 1968).

[44] P. Tichavský, C. H. Muravchik, and A. Nehorai, Posterior cramér-rao bounds for discrete-time nonlinear filtering, IEEE T. Signal Process. **46**, 1386 (1998).

[45] A. Tulsyan, B. Huang, R. B. Gopaluni, and J. F. Forbes, A particle filter approach to approximate posterior cramér-rao lower bound: The case of hidden states, IEEE T. Aero. Elec. Sys. **49**, 2478 (2013).

[46] B. Zhang, J. Zerubia, and J.-C. Olivo-Marin, Gaussian approximations of fluorescence microscope point-spread functions, Appl. Opt. **46**, 1819 (2007).

[47] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Classics in Applied Mathematics, Vol. 30 (SIAM, Philadelphia, 2000).