

**Hidden Markov modeling of single-particle diffusion with stochastic tethering**Amit Federbush<sup>1,2</sup>, Amit Moscovich<sup>3</sup>, and Yohai Bar-Sinai<sup>1,2</sup><sup>1</sup>*Department of Condensed Matter Physics, Tel Aviv University, Tel Aviv 69978, Israel*<sup>2</sup>*Center for Physics and Chemistry of Living Systems, Tel Aviv University, Tel Aviv 69978, Israel*<sup>3</sup>*Department of Statistics and Operations Research, Tel Aviv University, Tel Aviv 69978, Israel*

(Received 3 August 2023; accepted 14 February 2024; published 26 March 2024)

The statistics of the diffusive motion of particles often serve as an experimental proxy for their interaction with the environment. However, inferring the physical properties from the observed trajectories is challenging. Inspired by a recent experiment, here we analyze the problem of particles undergoing two-dimensional Brownian motion with transient tethering to the surface. We model the problem as a hidden Markov model where the physical position is observed and the tethering state is hidden. We develop an alternating maximization algorithm to infer the hidden state of the particle and estimate the physical parameters of the system. The crux of our method is a saddle-point-like approximation, which involves finding the most likely sequence of hidden states and estimating the physical parameters from it. Extensive numerical tests demonstrate that our algorithm reliably finds the model parameters and is insensitive to the initial guess. We discuss the different regimes of physical parameters and the algorithm's performance in these regimes. We also provide a free software implementation of our algorithm.

DOI: [10.1103/PhysRevE.109.034129](https://doi.org/10.1103/PhysRevE.109.034129)**I. INTRODUCTION**

Since the early days of statistical mechanics, the statistics of the stochastic motion of mesoscopic particles have been an important experimental probe for their microscopic properties. Most prominently, the Gaussian statistics of Brownian motion provided experimental proof of the atomic nature of matter [1–3] and were used to measure Avogadro's number [4]. Currently, new models are actively being developed to explain deviations from purely Brownian statistics in biological and colloidal systems [5,6]. Of specific interest are systems undergoing anomalous diffusion which exhibit a Fickian behavior, i.e., a mean-square displacement that is linear with time where the displacement statistics are non-Gaussian [5–7]. These deviations from Gaussianity can serve as an accessible probe for various experiments [8–13].

In the analysis of such systems, two key sources for non-Gaussianity are considered. First, particles may undergo several different types of diffusion modes, stochastically switching between them [14–19]. Second, particles may be transiently confined to a small region [9,20–32]. In this work, inspired by the experimental system of Chakraborty *et al.* [9], we focus on the latter and specifically on two-dimensional (2D) diffusion with transient tethering to the underlying surface. Figure 1 depicts an example trajectory from such an experiment. In such experiments, colloids or nanoparticles are coated with molecules of interest (typically, peptides) and undergo 2D diffusion on a surface coated with a different molecule. The interaction between the molecules leads to stochastic transient tethering of the particles to the surface. The experiment aims to extract information regarding the interaction between the peptides from the frequency of these tethering and untethering events.

However, identifying these events may be challenging, since the tethering to the surface is not directly observed but rather needs to be inferred from the observed trajectories. Several computational methods have been suggested to tackle this in the past [22,24,31]. In this work we present a simple and computationally efficient algorithm to infer tethering and untethering events from observed trajectories and to estimate the tethering and untethering rates, the diffusion coefficient, and the effective confinement area of the interaction potential. The algorithm is developed under the assumption that the dynamics follows normal diffusion with Poissonian tethering and untethering events, but can be easily extended to other types of diffusion and tethering and untethering rates.

The structure of this paper is as follows. In Sec. II we describe the stochastic hidden Markov model which we use to model the problem, discuss the different timescales that the model introduces, and define the regimes of model parameters where our methods are valid. In Sec. III we describe our algorithm and its underlying approximations. In Sec. IV we present our numerical results on synthetic data. In Sec. V we discuss the results and possible generalizations. The full source code of the implemented algorithm is available on GitHub [33].

**II. STOCHASTIC MODEL**

Our Markov model describes a particle alternating between a freely diffusing state and a tethered state in two dimensions. The transition between the states is modeled as a standard two-state continuous-time Markov chain with characteristic times  $\tau_0$  and  $\tau_1$ , which are the inverses of the average rates of tethering and untethering events, respectively. We denote the state of the particle by  $S(t)$ , where  $S = 0$  corresponds to the free state and  $S = 1$  to the tethered state. In the free state,

the particle undergoes standard Brownian motion, while in the tethered state, it is also confined to a harmonic potential centered at the tether point  $X^*(t)$ . Explicitly, the position of the particle  $X(t)$  follows an Ornstein-Uhlenbeck process with a state-dependent potential

$$\dot{X}(t) = -\frac{k}{\gamma}[X(t) - X^*(t)]S(t) + \sqrt{2D}\xi(t), \quad (1)$$

where  $k$  is the spring constant,  $\gamma$  is the friction coefficient,  $S(t) \in \{0, 1\}$  indicates the free or tethered state,  $D$  is the diffusion coefficient, and  $\xi$  is a standard two-dimensional white noise  $\langle \xi_i(t)\xi_j(t') \rangle = \delta_{ij}\delta(t-t')$ . Observe that only the ratio  $k/\gamma$  plays a role in the model, rather than  $k$  and  $\gamma$  individually. We assume the particle is tethered to the point at which the transition  $S = 0 \rightarrow 1$  occurs. More specifically, if the particle becomes tethered at time  $t_1$  and remains tethered until untethering at  $t_2$ , then  $X^*(t)$  satisfies

$$X^*(t) = X(t_1) \forall t \in [t_1, t_2]. \quad (2)$$

To summarize, the particle dynamics is modeled by a continuous-time Markov process

$$F(t) := (X(t), S(t), X^*(t)). \quad (3)$$

$$P(X(t + \Delta t)) = \frac{1}{2\pi A'(\Delta t)} \exp\left(-\frac{\{X(t + \Delta t) - \phi(\Delta t)X(t) - [1 - \phi(\Delta t)]X^*(t)\}^2}{2A'(\Delta t)}\right), \quad (5)$$

where we defined two auxiliary variables

$$\phi(\Delta t) = \exp\left(-\frac{D\Delta t}{A}\right), \quad A'(\Delta t) = [1 - \phi^2(\Delta t)]A. \quad (6)$$

### A. Discretized dynamics

The model defined above describes the full continuous-time dynamics. In principle, we could use it to estimate the model parameters  $\Theta = (\tau_0, \tau_1, D, A)$ . However, the experimental setup poses two difficulties. First, we can only measure the particle positions  $X(t)$  and do not have access to the particle states  $S(t)$  and tether points  $X^*(t)$ . This is the core challenge of the problem which we address in Sec. III. Second, we only sample the process at discrete times  $t_1, t_2, \dots, t_N$ , separated by a finite time resolution  $\Delta t = t_i - t_{i-1}$ . To account for this, we define a discrete Markov process, analogous to the continuous process, in which the tether point is constrained to be one of the previously observed positions. This discrete Markov process defines a sequence of random states  $\{F_n\}_{n=1}^N$ , where

$$F_n := F(t_n) = (X(t_n), S(t_n), X^*(t_n)). \quad (7)$$

The transition probabilities of  $F_n$  are given by the product of the transition probabilities of  $S$  and  $X$ ,

$$P(F_{n+1}|F_n) = P(S_{n+1}|S_n; \Theta)P(X_{n+1}|F_n; \Theta), \quad (8)$$

where  $X_{n+1}^*$  is uniquely determined by the history of  $\{X_n\}$  and  $\{S_n\}$ , in accordance with Eq. (2):

$$X_{n+1}^* = \begin{cases} X_{n+1} & \text{for } S_{n+1} = 1, S_n = 0 \\ X_n^* & \text{otherwise.} \end{cases} \quad (9)$$

The model is specified by four parameters:  $\tau_0, \tau_1, D$ , and the ratio  $k/\gamma$ . It is more convenient to work with the variable  $A = D\gamma/k$ , which is the characteristic area that the particle explores in the tethered state. For notational purposes, we group the model parameters as  $\Theta := (\tau_0, \tau_1, D, A)$ . Finally, we note that  $\frac{A}{D} = \frac{\gamma}{k}$  has units of time. Its meaning is discussed below.

The transition probabilities of  $S$  are Poissonian. In the untethered state  $S(t) = 0$ , the particle undergoes classic Brownian motion. Hence its position at  $t + \Delta t$  is normally distributed around  $X(t)$ ,

$$P(X(t + \Delta t)) = \frac{1}{4\pi D\Delta t} \exp\left(-\frac{[X(t + \Delta t) - X(t)]^2}{4D\Delta t}\right). \quad (4)$$

In the tethered state  $S(t) = 1$ , the probability density function of  $X(t + \Delta t)$  is given by the solution of the Fokker-Planck equation with strong friction in a harmonic potential around the anchor point  $X^*(t)$  [34],

The state transitions are Poissonian and to leading order in  $\Delta t$  read

$$P(S_{n+1}|S_n) = \begin{cases} 1 - \frac{\Delta t}{\tau_n}, & S_{n+1} = S_n \\ \frac{\Delta t}{\tau_n}, & S_{n+1} \neq S_n, \end{cases} \quad (10)$$

where  $\tau_n = \tau_0$  if  $S_n = 0$  and  $\tau_n = \tau_1$  if  $S_n = 1$ . The distribution of the particle position in the next step follows Eqs. (4) and (5),

$$P(X_{n+1}|F_n; \Theta) = \begin{cases} \frac{1}{4\pi D\Delta t} \exp[-\frac{1}{4D\Delta t}(X_{n+1} - X_n)^2], & S_n = 0 \\ \frac{1}{2\pi A'} \exp\{-\frac{1}{2A'}[X_{n+1} - \phi X_n - (1 - \phi)X_n^*]^2\}, & S_n = 1, \end{cases} \quad (11)$$

where now  $\phi$  and  $A'$  are constants that depend on  $A$  and  $D$  and the time step  $\Delta t$  as in Eq. (6). Furthermore, we assume that at the beginning of the measurement the particle is either free ( $S_1 = 0$ ) or tethered at  $X_1$  ( $S_1 = 1$  and  $X_1^* = X_1$ ). Importantly,  $X_n^*$  can only attain values among the previously visited positions  $X_1, \dots, X_n$ .

As usual for Markov processes, the probability of a trajectory  $\{F_n\}_{n=1}^N$  is the product of the transition probabilities in each step; thus the log-probability is additive:

$$\begin{aligned} \log P(\{F_n\}_{n=1}^N; \Theta) \\ = \log P(F_1) + \sum_{n=1}^{N-1} \log P(F_{n+1}|F_n; \Theta). \end{aligned} \quad (12)$$

An example trajectory generated from the discrete model is depicted in Fig. 2.

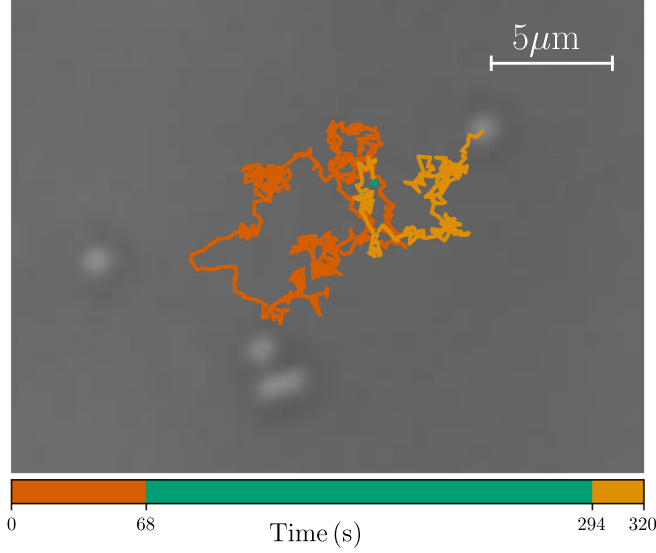


FIG. 1. Snapshot of an experimental microscopy video of peptide-coated microparticles, in a system similar to that of Chakraborty *et al.* [9]. The colored curve represents the top right particle's recorded trajectory in the 320 s preceding the snapshot. The trajectory is colored based on time, as indicated by the timeline at the bottom. During the time segment  $t \in [68, 294]$ , the particle is confined to a small region of space (shown in green) due to tethering. (Image is courtesy of Amandeep Sekhon, Roy Beck, and Yael Roichman).

### B. Timescales

As seen above, the problem involves multiple timescales. Three of them are given by the model parameters and represent the underlying physics:  $\tau_0$ ,  $\tau_1$ , and  $\frac{A}{D}$ , which is the equilibration time of the Brownian particle with the harmonic potential. The other two timescales,  $\Delta t$  and the total

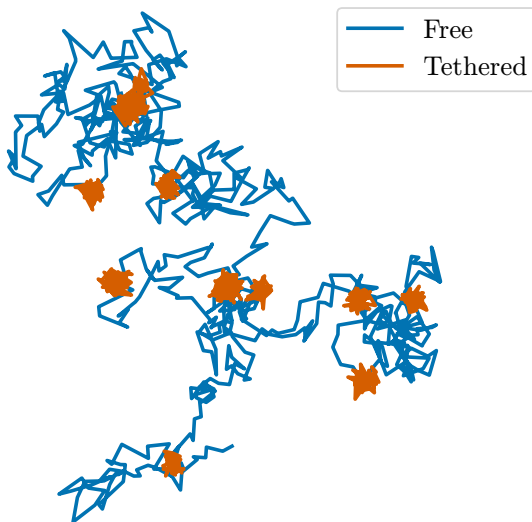


FIG. 2. Example trajectory of  $N = 2000$  steps generated from the discrete model with  $\tau_0 = \tau_1 = 100$ ,  $D = 1$ , and  $A = 0.5$ . The sampling time step is  $\Delta t = 1$ . The trajectory is color coded according to the particle's state, where blue and orange depict the free and tethered states, respectively.

experiment duration  $T = t_N - t_1$ , are properties of the experiment. We employ several realistic working assumptions about these timescales that greatly reduce computational complexity. First, we assume that  $\tau_0, \tau_1 \gg \frac{A}{D}$ , which physically means that the rate of tethering and untethering events is much smaller than the inverse of the harmonic equilibration time of the particle. Violation of this condition means that the particle can untether before the tethering potential has a significant effect and thus tethering events will not be experimentally discernible. This regime corresponds to standard Brownian motion with an effective diffusion constant smaller than  $D$ . Second, we require that  $\Delta t \ll \tau_0, \tau_1$  to avoid the possibility of multiple transitions of tethering or untethering events within a single sampling interval. This assumption is experimentally realistic since modern cameras can easily achieve frame rates higher than  $10^3$  Hz, and in many experiments  $\tau_0$  and  $\tau_1$  are of orders of at least seconds [9,20,25]. In any case, if  $\Delta t \approx \tau_0, \tau_1$ , this would again correspond to an essentially pure diffusive behavior in the discretized data. Third, if  $\Delta t < \frac{A}{D}$ , the time discretization resolves the equilibration of a tethered particle with its confining potential. Since our goal is only to extract the physical parameters of the system, such resolution does not add relevant information and only increases the computational cost. Therefore, undersampling the discrete dynamics to increase  $\Delta t$  should not lead to a significant loss of accuracy in the estimation of  $\tau_0$  and  $\tau_1$  but would greatly reduce the search space. This is explicitly demonstrated in Sec. IV. Even if the trajectories are experimentally measured with small  $\Delta t$ , we can safely downsample them such that  $\frac{A}{D} \lesssim \Delta t$ , or in other words  $\phi \lesssim e^{-1}$  [cf. Eq. (6)]. For concreteness, we mention the experimental parameters of the diffusing nanoparticles system of Chakraborty *et al.* [9]. In this system,  $\tau_0, \tau_1 \sim 1$  s,  $D \sim 10 \mu\text{m}^2/\text{s}$ , and  $A \sim 1 \mu\text{m}^2$ , which means a sampling interval of  $\Delta t = \frac{A}{D} = 0.1$  s is sufficient and is easily achievable experimentally. To summarize this, we assume the following separation of timescales:

$$\frac{A}{D} \lesssim \Delta t \ll \tau_0, \tau_1. \quad (13)$$

The leading-order expansion of Eq. (11), which is first order in  $\Delta t$  and zeroth order in  $\phi$ , now reads

$$P(S_{n+1}|S_n) = \begin{cases} 1 - \frac{\Delta t}{\tau_n}, & S_{n+1} = S_n \\ \frac{\Delta t}{\tau_n}, & S_{n+1} \neq S_n, \end{cases}$$

$$P(X_{n+1}|F_n) = \begin{cases} \frac{1}{4\pi D\Delta t} \exp[-\frac{1}{4D\Delta t}(X_{n+1} - X_n)^2], & S_n = 0 \\ \frac{1}{2\pi A} \exp[-\frac{1}{2A}(X_{n+1} - X_n^*)^2], & S_n = 1, \end{cases} \quad (14)$$

where  $\tau_n = \tau_0$  if  $S_n = 0$  and  $\tau_n = \tau_1$  if  $S_n = 1$ .

To conclude the discussion regarding timescales, Fig. 3 illustrates the section of phase space where our assumptions hold (in time units of  $\frac{A}{D}$ ). For a point in the yellow region in the figure, the farther it is from the two boundary lines, the better the separation of scales is and the better our assumptions hold. The marked points correspond to regimes that will be analyzed in Sec. IV.

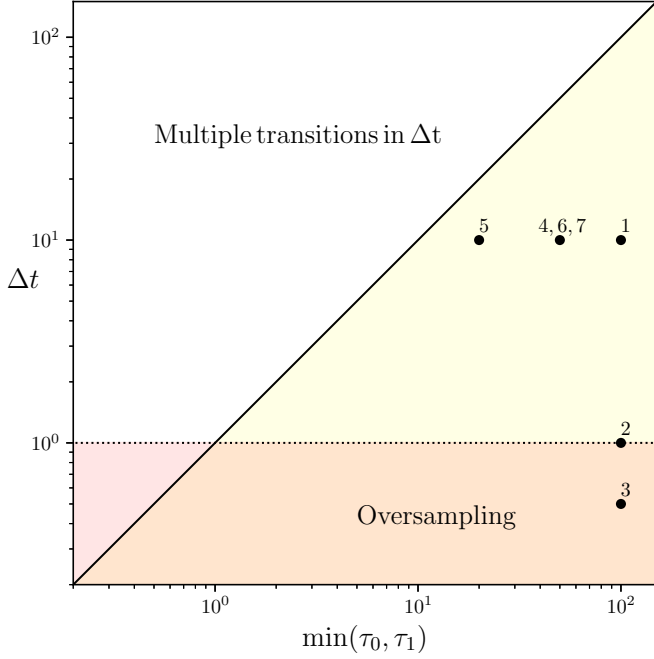


FIG. 3. Section of the phase space of  $\Delta t$ ,  $\tau_0$ , and  $\tau_1$  in time units of  $\frac{A}{D}$ . Note the logarithmic scale. The yellow region is where our assumptions hold. The bottom horizontal boundary of the yellow region corresponds to the inequality  $\frac{A}{D} \lesssim \Delta t$  and the diagonal boundary corresponds to the inequality  $\Delta t \ll \tau_0, \tau_1$ . The bottom and left regions are annotated according to the discussion in this section. The markers correspond to the seven regimes analyzed in Sec. IV.

### III. OUR METHOD

Our problem is as follows: In an experiment, we can measure the observed states  $\{X_n\}$ , but we do not have access to the hidden states  $\{S_n, X_n^*\}$  or to the model parameters  $\Theta = (\tau_0, \tau_1, D, A)$ . We use the term *hidden path* to denote the sequence of the hidden states  $\{S_n\}_{n=1}^N$  from which the sequence  $\{X_n^*\}_{n=1}^N$  can be determined. The goal is to infer the model parameters  $\Theta$  from a series of measurements. To this end, we developed an alternating maximization algorithm [35], similar to the classical expectation-maximization algorithm [36], to estimate both the hidden states and the parameters of our hidden Markov model.

If the hidden path is known, the problem of optimal parameter estimation is fairly standard and is typically solved by maximizing the likelihood of the model parameters, which according to Bayes' rule is proportional to the exponential of Eq. (12). Maximizing  $P(\Theta|\{F_n\})$  can be done numerically or using analytical approximations, as described below. However, when the hidden path is not known, the likelihood function to consider is

$$\begin{aligned} \mathcal{L}(\Theta|\{X_n\}) &= \frac{P(\Theta)}{P(\{X_n\})} P(\{X_n\}|\Theta) \quad (\text{Bayes}) \\ &= \frac{P(\Theta)}{P(\{X_n\})} \sum_{\{S_n\}} P(\{F_n\}|\Theta), \end{aligned} \quad (15)$$

where the sum is over all  $2^N$  possibilities for the hidden paths,  $P(\Theta)$  is the prior probability distribution for the model parameters, and the evidence  $P(\{X_n\})$  is a constant we may

ignore [37]. Recall that  $\{F_n\} = \{X_n, S_n, X_n^*\}$  is the trajectory of both the observed and hidden states of the particle, and the log-probability of such a trajectory is the sum of the log-probabilities of all the steps, as in Eq. (12). Computing the sum in Eq. (15) is intractable for typical values of  $N$ . However, numerical evidence shows that most hidden paths are very unlikely and thus have a negligible contribution to this sum. Taking a uniform prior  $P(\Theta) = \text{const}$ , we posit that

$$\log \mathcal{L}(\Theta|\{X_n\}) \sim \max_{\{S_n\}} \log P(\{F_n\}|\Theta). \quad (16)$$

A key idea of our method is to use the right-hand side of Eq. (16) as a proxy for the computationally intractable left-hand side. It is analogous to the saddle-point approximation from statistical mechanics, where the integral is replaced with the maximum of the integrand. To compute the right-hand side of Eq. (16), we only need to find the most likely hidden path  $\{\hat{S}_n\}_{n=1}^N$ , given the model parameters  $\Theta$ . This discrete optimization problem can be efficiently solved using the Viterbi algorithm from dynamic programming [37,38]. Below we briefly describe the parameter estimation method and how the Viterbi algorithm is implemented for our model. Then we present our alternating maximization approach for estimating the maximum-likelihood model parameters.

#### A. Parameter estimation

Given the most likely hidden path  $\{\hat{S}_n\}$ , maximizing the log-likelihood in Eq. (16) is maximizing a sum of the logarithmic terms from Eq. (14). We use the following maximum-likelihood estimators (MLEs) to approximate the most likely model parameters:

$$\begin{aligned} \hat{\tau}_0 &:= \frac{N_{00} + N_{01}}{N_{01}} \Delta t, & \hat{\tau}_1 &:= \frac{N_{11} + N_{10}}{N_{10}} \Delta t, \\ \hat{D} &:= \frac{1}{4(N_{00} + N_{01}) \Delta t} \sum_{n=1}^{N-1} (1 - \hat{S}_n) (X_{n+1} - X_n)^2, \\ \hat{A} &:= \frac{1}{2(N_{10} + N_{11})} \sum_{n=1}^{N-1} \hat{S}_n (X_{n+1} - X_n^*)^2. \end{aligned} \quad (17)$$

Here  $N_{ij}$  is the number of  $i \rightarrow j$  transitions of  $\hat{S}_n$  that occur along the trajectory. The derivation of these estimators is trivial by taking the derivative of Eq. (16) with respect to each model parameter and equating to zero.

#### B. Finding the most likely sequence of states

We now describe how to find the maximum-likelihood path  $\{\hat{S}_n\}$ , conditioned on the model parameters  $\Theta$ . Recall that the tether point  $X_n^*$  can only assume one of the previously visited positions  $X_1, X_2, \dots, X_n$  [see Eq. (9)]. We represent the set of all possible paths as a directed layer graph (Fig. 4), known as a trellis in the Viterbi literature. In this graph, columns correspond to the discrete time  $n$  and rows to the tether point  $X_n^*$ , which can be either one of  $X_1, \dots, X_n$  or free (untethered). Each vertex represents the state of the particle at time  $n$  and each path from left to right corresponds to a specific sequence of tethered and free states.

We set the edge weights to be the logarithm of the transition probabilities between states according to Eq. (8). With

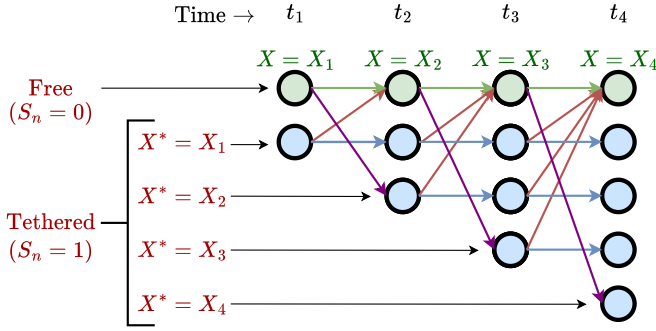


FIG. 4. Trellis graph of the observed and hidden states, given the observed particle positions  $X_1, \dots, X_4$ . Each node represents a different state  $F_n = (X_n, S_n, X_n^*)$ , with the row corresponding to the hidden state and the column to the time  $n$  and the observed state  $X_n$ . The edges represent allowed state transitions and are weighted according to the logarithm of Eq. (8). Each trajectory  $\{F_n\}$  is given as a path on the graph that advances from left to right.

this choice of edge weights, the log-likelihood of a path [Eq. (12)] is just  $\log P(F_1)$  plus the sum of edge weights along the corresponding path in the graph. Thus, finding the maximum-likelihood sequence of hidden states is reduced to the problem of finding the path of maximum weight in a directed layer graph. The latter problem is efficiently solved using the Viterbi algorithm [38]. This algorithm scans the trellis column by column from left to right and computes, for each vertex, the maximum-weight path that ends at that vertex.

### C. Alternating maximization algorithm

Given an estimate of the hidden path  $\{S_n\}$ , we can apply Eq. (17) to obtain the maximum-likelihood estimate of the model parameters  $\Theta = (\tau_0, \tau_1, D, A)$ . Conversely, given an estimate of the model parameters  $\Theta$ , it is easy to find the maximum-likelihood hidden path using the Viterbi algorithm as explained in the preceding section. Combining these two observations naturally leads to an alternating maximization procedure for estimating both the most likely model parameters  $\hat{\Theta}$  and the most likely hidden path.

(1) *Initial guess.* Guess an initial value  $\hat{\Theta}^{(0)}$  for the four model parameters.

(2) *Path maximization step.* Conditioned on the current parameter estimate  $\hat{\Theta}^{(m)}$ , apply the Viterbi algorithm to find  $\{S_n^{(m)}\}_{n=1}^N$ , the most likely hidden path given  $\hat{\Theta}^{(m)}$ ,

$$S^{(m)} := \arg \max_{\{S_n\}} \mathcal{L}(\{S_n\} | \hat{\Theta}^{(m)}, \{X_n\}). \quad (18)$$

(3) *Parameter maximization step.* Use Eq. (17) to obtain the maximum-likelihood estimate of the model parameters conditioned on the current estimate of the hidden path,

$$\hat{\Theta}^{(m+1)} := \hat{\Theta}(\{S^{(m)}\}, \{X_n\}). \quad (19)$$

(4) *Convergence.* Alternate between steps 2 and 3 until  $|\hat{\theta}^{(m+1)} - \hat{\theta}^{(m)}| / \hat{\theta}^{(m)} \leq \epsilon$  for each  $\hat{\theta} \in \Theta$ .

Since the sample space is discrete, convergence typically occurs exactly, i.e.,  $\hat{\Theta}^{(n+1)} = \hat{\Theta}^{(n)}$ . However, in our experiments we used  $\epsilon = 10^{-3}$  to stop the iterations when the relative change is small. The specific threshold  $10^{-3}$  is inconsequential. Furthermore, we set the maximum number of

iterations to 20, to prevent the possibility of infinite loops that alternate between several discrete hidden paths without satisfying the convergence criterion (in practice, less than ten iterations typically suffice for convergence). We also define a criterion for divergence: If after the parameter maximization step the estimators  $\hat{\tau}_0$  or  $\hat{\tau}_1$  exceed the arbitrary threshold of  $0.9T$ , we say the algorithm diverged and stop the iterations. This is done to avoid cases where the particle is estimated to stay tethered or untethered for the entire sampling time due to an unphysical divergence of the MLEs. In practice, we test the regimes where  $T \gg \tau_0, \tau_1$  and most runs do not diverge. We say the algorithm has converged if the convergence criterion was satisfied without first triggering the divergence criterion.

The Viterbi algorithm scans each edge on the graph once, leading to an  $O(N^2)$  complexity. However, most paths are not likely, e.g., a path where at some point  $S_n = 1$  and  $(X_n - X_n^*)^2 \gg A$ , meaning the particle roams far from its tether point. To improve the running time, we implemented edge pruning of the trellis graph. Specifically, we discard the paths of all but the  $q$  most likely tethered nodes at each time step (column) during the algorithm's execution. This reduces the complexity to  $O(qN)$ . In our tests, we used  $q = 10$ , which showed no significant impact on the results compared to no pruning.

In the Appendix we discuss an extension of our algorithm that outputs the  $K$  most likely hidden paths rather than the single best path. We generally found that the 100 most likely paths yield similar parameter estimates, so the results in the rest of this paper use  $K = 1$ .

### D. Bootstrap bias correction

For reasons that will be discussed in Sec. IV, the MLE predictions for the model parameters give very good estimates of  $D$  and  $A$ , but consistently overestimate the waiting times  $\tau_0$  and  $\tau_1$ . This overestimation can be largely corrected by using a parametric bootstrap procedure [39]. The idea is simple: Define the bias  $\Delta$  as the expected difference between the MLE and the true parameters:  $\Delta = \langle \hat{\Theta} \rangle - \Theta$ . Assuming the bias does not depend sensitively on  $\Theta$  itself, one can estimate  $\Delta$  by simulating trajectories with parameters  $\hat{\Theta}$  and rerunning the procedure on this simulated data (for which we know the true parameters). This is done as follows.

(i) Apply the alternating maximization algorithm described in Sec. III C to obtain the MLE  $\hat{\Theta}$ .

(ii) Generate  $M$  random trajectories using  $\hat{\Theta}$  as the “true” simulated model parameters.

(iii) Apply the alternating maximization algorithm to each simulated trajectory, yielding the estimated model parameters  $\hat{\Theta}_i^j$  for  $1 \leq i \leq M$ .

(iv) The bias is estimated by the median value of the observed biases,

$$\hat{\Delta} := \text{median}_{1 \leq i \leq M} (\hat{\Theta}_i^j - \hat{\Theta}).$$

(v) Finally, the bias is used to correct the original estimate, that is, the bias-corrected estimate for the true model parameters is given by

$$\hat{\Theta}^B = \hat{\Theta} - \hat{\Delta}. \quad (20)$$

TABLE I. True model parameters, accuracy, and estimated parameters (mean  $\pm$  standard deviation) for the seven regimes analyzed using the alternating maximization algorithm (Sec. III C) without bias correction. The accuracy is defined as the fraction of time steps at which the algorithm correctly predicted both the tethered state  $S$  and the tether point  $X^*$ .

Regime	$D, A$	$\Delta t$	$\tau_0$	$\tau_1$	Accuracy (%)	$\hat{\tau}_0$	$\hat{\tau}_1$	$\hat{D}$	$\hat{A}$
1	1	10	100	100	$96 \pm 2$	$131 \pm 24$	$130 \pm 19$	$1.00 \pm 0.05$	$0.99 \pm 0.05$
2	1	1	100	100	$94 \pm 2$	$122 \pm 21$	$122 \pm 16$	$1.00 \pm 0.01$	$0.99 \pm 0.02$
3	1	0.5	100	100	$88 \pm 4$	$125 \pm 22$	$123 \pm 17$	$1.00 \pm 0.01$	$0.99 \pm 0.02$
4	1	10	50	50	$93 \pm 2$	$77 \pm 11$	$75 \pm 8$	$0.99 \pm 0.05$	$0.98 \pm 0.05$
5	1	10	20	20	$87 \pm 2$	$47 \pm 9$	$43 \pm 5$	$0.97 \pm 0.06$	$0.94 \pm 0.06$
6	1	10	200	50	$96 \pm 1$	$356 \pm 95$	$79 \pm 12$	$0.99 \pm 0.04$	$0.97 \pm 0.09$
7	1	10	50	200	$97 \pm 2$	$60 \pm 11$	$248 \pm 43$	$1.01 \pm 0.08$	$1.00 \pm 0.04$

Note that the median is calculated for each parameter  $\theta \in \Theta$  independently. We chose to estimate  $\hat{A}$  using the median since it is robust to outliers and we found that it gave better results in mean-square error.

#### IV. RESULTS

We test the algorithm's performance on synthetically generated trajectories, whose model parameters  $\Theta$  are known. The implementation of the algorithm in PYTHON, as well as the code to generate all figures in this paper, is available at [33].

In all trajectories, we use  $D = 1$  and  $A = 1$  and a total experiment duration of  $T = 10\,000$ . Note that both  $D$  and  $A$  can always be set to unity by properly choosing length and time units, so this choice is without loss of generality. Thus, each experiment is specified by three scalars, the sampling time  $\Delta t$  and the model parameters  $\tau_0$  and  $\tau_1$ , all expressed in time units of  $\frac{A}{D} = 1$ . We focus on seven different regimes of  $\Theta$  and  $\Delta t$ , detailed in Table I. We mention that the conditions in Eq. (13) are barely satisfied in regimes 3 and 5. This is intentional, as we wish to test the algorithm slightly beyond its bounds.

##### A. Stability

As described above, the algorithm requires an initial guess of the model parameters  $\hat{\Theta}^{(0)}$ . We found empirically that the algorithm converges to a single fixed point almost

regardless of the initial parameters. To demonstrate this, we show in Fig. 5 the results of the algorithm when applied to a single observed trajectory corresponding to regime 1, with 1000 different  $\hat{\Theta}^{(0)}$ 's, randomly drawn from a log-uniform distribution spanning two decades around  $\Theta$ . We mention this experiment's wide distribution of initializations  $\hat{\Theta}^{(0)}$  is intended to illustrate the algorithm's stability, and in a practical situation, better priors can often be used, especially for  $D$  and  $A$ . The convergence for other regimes is qualitatively similar and is not shown here. Out of the 1000 algorithm runs, 960 converged and only these runs were taken into account in the figure. Since the outcome of the algorithm is largely independent of the initial guess, in what follows we use the true model parameters  $\Theta$  as the initial guess  $\hat{\Theta}^{(0)}$ .

##### B. Recovering the parameters and hidden states

For each of the regimes in Table I, we generated 1000 synthetic trajectories and ran the algorithm with initial parameters that are equal to the true model parameters, as discussed in the preceding section regarding stability. Over 98% of the runs in each regime have converged (the other runs entered a loop and stopped after the maximum number of iterations). Runs that converged typically did so within three to eight iterations. We define the accuracy as the fraction of time steps for which the algorithm predicted both the correct state  $S_n$  and tether point  $X_n^*$  (if  $S_n = 1$ ). The mean and standard deviation of the accuracy over all converged runs are detailed in Table I. It can be seen that the accuracy is fairly high in all regimes and that

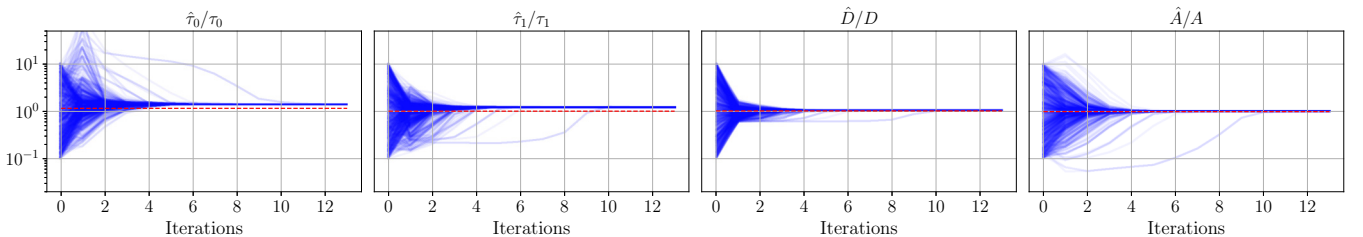


FIG. 5. The algorithm is insensitive to the initialization  $\hat{\Theta}^{(0)}$ . Here we show the evolution of the estimators  $\hat{\tau}_0$ ,  $\hat{\tau}_1$ ,  $\hat{D}$ , and  $\hat{A}$  of the four model parameters along the algorithm iterations over a single synthetic trajectory. The values are normalized by the true parameter values so that a perfect prediction would be unity in all panels. The first iteration is the initial parameter guess. We drew 1000 initial guesses at random from a log-uniform distribution around the true value and ran the algorithm, keeping only the 960 runs that converged. The evolution of the estimators of each (converged) run is shown, exhibiting a strong convergence towards a single fixed point. Note that the vertical axis is logarithmic. The red dashed line represents the most likely parameter value, given the true hidden path. It does not coincide, in general, with the true model parameters.

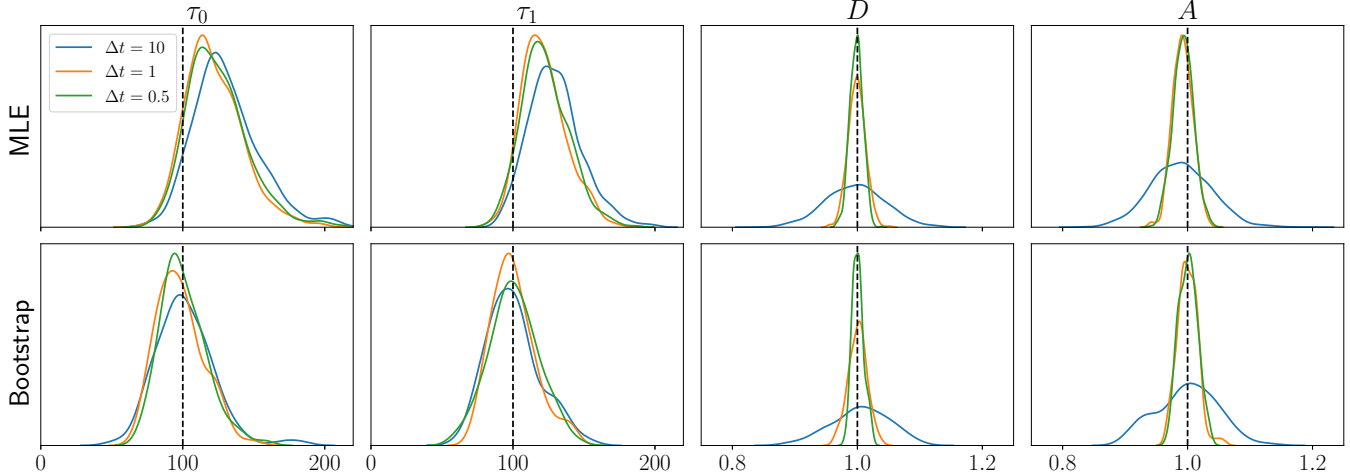


FIG. 6. Distribution of the algorithm's estimates of physical model parameters for regimes 1, 2, and 3 with  $\tau_0 = \tau_1 = 100$  and  $\Delta t = 10, 1, 0.5$ , respectively. The dashed lines represent the true parameter value. The top row shows the maximum-likelihood estimates of the alternating maximization algorithm. The bottom row shows the estimators after the bootstrap bias correction procedure.

it decreases as  $\Delta t$  approaches  $\frac{A}{D}$  from above or  $\tau_0$  and  $\tau_1$  from below, consistently with the constraints of Eq. (13).

Next we examine how well the algorithm recovers the model parameters. For each regime, Table I depicts the mean and the standard deviation of the algorithm's estimates  $\hat{\Theta} = (\hat{\tau}_0, \hat{\tau}_1, \hat{D}, \hat{A})$  over all converged runs. These results exhibit two clear trends: The temporal parameters  $\tau_0$  and  $\tau_1$  are consistently overestimated, while the spatial ones  $D$  and  $A$  are correctly estimated (although there is a very slight yet consistent underestimate of  $A$  in most regimes). The overestimate of the temporal parameters ranges from 20% to 120% and will be discussed in Sec. IV C.

To investigate the effect of the sampling time  $\Delta t$ , we focus on the three regimes 1, 2, and 3, which differ only by  $\Delta t$  (see Table I). The distributions of the estimated model parameters are depicted in the top row of Fig. 6. Note that the centers of the distributions hardly change with  $\Delta t$ . This supports the claim described in Sec. II that the sampling time can be increased without losing information, as long as Eq. (13) holds. We mention that the width of the distribution for the spatial parameters  $D$  and  $A$  decreases with  $\Delta t$ . That is to be expected since the accuracy of the MLEs increases with  $N$ , but for a fixed  $T$ , the number of samples  $N$  is inversely proportional to  $\Delta t$ .

By focusing on regimes 1, 4, and 5, the effect of  $\tau_0$  and  $\tau_1$ , when they are equal, can be isolated. As  $\tau_0$  and  $\tau_1$  decrease, the accuracy decreases and the relative overestimate of  $\tau_0$  and  $\tau_1$  increases. This is consistent with Eq. (13), since decreasing  $\tau_0$  and  $\tau_1$  challenges the assumption that  $\Delta t \ll \tau_0, \tau_1$ . Finally, regimes 6 and 7 explore the case of  $\tau_0 \neq \tau_1$ . Note that when  $\tau_0 < \tau_1$ , i.e., when the particle is more likely to be tethered than free, the parameter estimates are better, with less overestimation for the temporal parameters, compared to the opposite case.

### C. Overestimation and bias correction

As seen in Table I and Fig. 6, while the spatial parameters  $D$  and  $A$  are correctly estimated to high accuracy, the temporal

parameters  $\tau_0$  and  $\tau_1$  are consistently overestimated, with an increasing overestimation as  $\tau_0$  and  $\tau_1$  decrease compared to  $\Delta t$ . This bias is due to a systematic misidentification of brief tethered or untethered intervals. More specifically, if the particle tethers and then untethers (or vice versa) over a short time window, the likelihood is dominated by the temporal terms (regarding  $S_n$ ) in Eq. (14) over the spatial terms. As a result, the likelihood of a trajectory that stays in the same state is higher than that of a trajectory that switches tethering states twice within this short interval, even if the true dynamics involved such a switch.

To see this, take a short time interval of  $l$  steps, in which the particle does not move much. Compare two sequences of hidden states, one in which the particle is tethered in the first step and untethered in the last one, and one in which it is free throughout the time window (a similar argument holds for the opposite scenario). Taking the difference between the log-likelihoods of the two hidden trajectories using Eq. (14) and expanding to leading order in  $l$ , we obtain

$$\Delta \mathcal{L} = 2 \log \left( \frac{1 - \Delta t / \tau}{\Delta t / \tau} \right) + O(l, \Delta t^2), \quad (21)$$

where for simplicity we took  $\tau_0 = \tau_1 = \tau$ . For short time windows, this tends to be positive since  $\Delta t \ll \tau_0, \tau_1$ , with the first term slowly diverging as  $\Delta t \rightarrow 0$ .

In addition, since the distribution of the free and tethered interval durations times is exponential, it is dominated by such short intervals, which are systematically misclassified by the MLE estimates. An example of this phenomenon is illustrated in Fig. 7. Since the MLE estimates for  $\tau_0$  and  $\tau_1$  [Eq. (17)] are inversely proportional to the number of switching events, missing brief tethering and untethering intervals leads to overestimation of the waiting times. We correct this bias using a parametric bootstrap procedure, as described in Sec. III D. The bottom row of Fig. 6 shows the distribution of the bias-corrected estimates  $\hat{\Theta}^B$ , which are concentrated within a small region around the true value. Note that although the mode of the distributions for  $\tau_0$  and  $\tau_1$  is slightly biased, the means of the distributions are close to the true values, as evident in

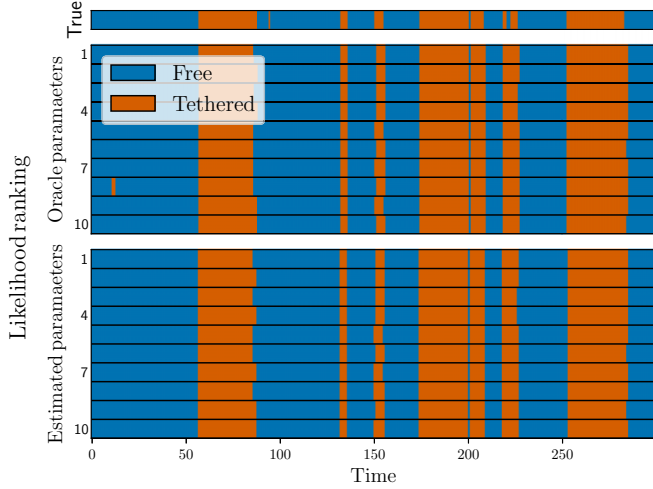


FIG. 7. Hidden paths of a single trajectory realization with  $N = 300$  steps,  $\tau_0 = \tau_1 = 100$ ,  $D = 1$ ,  $A = 1$ , and  $\Delta t = 10$ . The true hidden path is at the top. In the middle are the ten most likely hidden paths conditioned on the true parameters, with the upper row being the most likely. At the bottom are the ten most likely hidden paths given the algorithm’s estimated model parameters, again with the upper row being the most likely. Blue intervals correspond to the free state and orange intervals correspond to the tethered state.

Table II. Importantly, the bootstrapping successfully mitigates the bias in the rate estimation without affecting the already accurate estimates of  $D$  and  $A$ . In our simulations we generated  $M = 100$  bootstrapping trajectories (cf. Sec. III D) for each analyzed trajectory.

## V. DISCUSSION AND OUTLOOK

We developed and tested an efficient and accurate algorithm to analyze trajectories of diffusing particles with transient tethering, to identify tethering and untethering events, and to estimate the physical parameters of the system. The crux of our method is using the Viterbi algorithm to find the most likely sequence of hidden states, estimate the most likely model parameters for that sequence, and then bias correct the estimates using a parametric bootstrap procedure. Our algorithm successfully recovers the model parameters and is largely insensitive to the parameter initialization. It is applicable when the time interval between frames  $\Delta t$  is significantly shorter than the typical tethering and untethering times and longer than, or comparable to, the equilibration time

of the particle with the tethering potential. When the latter condition is not met, downsampling of the time series can be performed without significant loss of accuracy in estimating the physical parameters.

Our method as presented above only applies, of course, to normal 2D diffusion with Poissonian transient tethering. However, it can be readily generalized to other situations. For example, it can easily be generalized to a different spatial dimension ( $d \neq 2$ ) by adjusting the prefactors in Eq. (11). In higher dimensions, the aforementioned misidentification of brief tethered intervals should be less prominent since a sequence of short steps becomes less likely (no recurrence). Other diffusive statistics (anomalous diffusion) or non-Poissonian tethering and untethering can also be modeled by adjusting the transition probabilities [Eq. (14)]. For example, it could incorporate a drift term or an observational error term for position samples, as in Bernstein and Fricks’s work [22]. However, one must assume a specific functional form for the diffusion and tethering dynamics; in the regime in which we are working, only a few dozen tethering and untethering events per trajectory are observed. Hence, due to the small sample size, statistical goodness-of-fit tests would lack sufficient power to distinguish, e.g., between Poissonian and non-Poissonian tethering.

## ACKNOWLEDGMENTS

We thank Roy Beck, Yael Roichman, Indrani Chakraborty, and Amandeep Sekhon for useful discussions and for providing experimental data. Y.B.-S. was supported by ISF Grant No. 1907/22 and by Google Gift grant. A.M. was supported by ISF Grant No. 1662/22 and NSF-BSF Grant No. 2022778.

## APPENDIX: THE $K$ MOST LIKELY PATHS

Our saddle-point approximation in Eq. (16) replaces the sum over all hidden paths with the likelihood value of the single most likely path. An immediate generalization is to sum over the top  $K$  likeliest paths, which would improve the estimation accuracy of  $\mathcal{L}$ . Given the top  $K$  likeliest paths, we can estimate the model parameters as a weighted sum of the MLEs of each path [cf. Eq. (17)] as

$$\hat{\Theta}^{[K]} = \frac{\sum_{i=1}^K P(\{F_n\}^{[i]}|\Theta)\hat{\Theta}(\{F_n\}^{[i]})}{\sum_{i=1}^K P(\{F_n\}^{[i]}|\Theta)}, \quad (\text{A1})$$

TABLE II. True model parameters vs their bootstrap bias-corrected estimates for the seven regimes analyzed. The estimates display the mean and a 95% confidence interval. We used  $M = 100$  synthetic bootstrap trajectories per analyzed trajectory.

Regime	$D, A$	$\Delta t$	$\tau_0$	$\tau_1$	$\hat{\tau}_0^B$	$\hat{\tau}_1^B$	$\hat{D}^B$	$\hat{A}^B$
1	1	10	100	100	102 (71–141)	100 (73–139)	1.00 (0.91–1.08)	1.00 (0.91–1.08)
2	1	1	100	100	98 (73–129)	100 (76–138)	1.00 (0.97–1.03)	1.00 (0.97–1.03)
3	1	0.5	100	100	101 (75–137)	101 (69–133)	1.00 (0.99–1.02)	1.00 (0.98–1.03)
4	1	10	50	50	49 (38–64)	49 (36–61)	1.00 (0.92–1.08)	0.99 (0.87–1.07)
5	1	10	20	20	18 (9–28)	19 (11–26)	0.98 (0.88–1.11)	0.98 (0.87–1.10)
6	1	10	200	50	190 (100–316)	51 (32–77)	1.00 (0.92–1.06)	0.99 (0.85–1.16)
7	1	10	50	200	51 (36–76)	198 (131–295)	1.00 (0.87–1.20)	1.00 (0.94–1.08)



where  $\{F_n\}^{[i]}$  is the trajectory corresponding to the  $i$ th most likely hidden path. Finding the  $K$  highest likelihood paths can be easily done by a slight modification of the Viterbi algorithm, also known as list Viterbi [40,41]. The modification is discarding all but the  $K$  most likely paths ending at each node rather than all but the most likely path. Doing this increases the computational complexity by  $K^2$ .

In our numerical tests, we found no significant improvement in the results when increasing  $K$  up to  $K = 100$ . This is because the  $K$  most likely paths are very similar, differing from each other in just a few steps, leading to very similar MLEs of  $\Theta$ . Typically, the top paths only differ by slight perturbations of the tethering and untethering times and do not display qualitative differences.

To demonstrate this, Fig. 7 depicts the true hidden path of a single trajectory along with the top  $K = 10$  most likely hidden paths. The likelihood of these paths can be computed with respect to either the most likely parameters given the true hidden path (“oracle”) or the estimated  $\hat{\Theta}$  (without bootstrapping). It is seen that this choice does not significantly change the result, and the most likely hidden paths are all qualitatively very similar, differing by just a few steps from each other, and are similar to the true hidden path. They miss the brief tethered interval at time  $t = 93$  and the brief free interval at time  $t = 220$ . Because the results were similar for  $K$  up to 100, in Sec. IV we presented only results pertaining to  $K = 1$ . However, using  $K > 1$  top paths may be beneficial in other regimes.

- 
- [1] A. Einstein, Über die von der molekularkinetischen theorie der wärme geforderte bewegung von in ruhenden flüssigkeiten suspendierten teilchen, *Ann. Phys. (Leipzig)* **322**, 549 (1905).
- [2] M. von Smoluchowski, Zur kinetischen theorie der Brownschen molekularbewegung und der suspensionen, *Ann. Phys. (Leipzig)* **326**, 756 (1906).
- [3] J. Perrin, Mouvement Brownien et réalité moléculaire, *Ann. Chim. Phys.* **18**, 1 (1909), English translation by F. Soddy in [https://web.mit.edu/swangroup/footer/perrin\\_bm.pdf](https://web.mit.edu/swangroup/footer/perrin_bm.pdf).
- [4] E. Kappler, Versuche zur messung der Avogadro-Loschmidtschen zahl aus der Brownschen bewegung einer drehwaage, *Ann. Phys. (Leipzig)* **403**, 233 (1931).
- [5] B. Wang, S. M. Anthony, S. C. Bae, and S. Granick, Anomalous yet Brownian, *Proc. Natl. Acad. Sci. USA* **106**, 15160 (2009).
- [6] B. Wang, J. Kuo, S. C. Bae, and S. Granick, When Brownian diffusion is not Gaussian, *Nat. Mater.* **11**, 481 (2012).
- [7] A. V. Chechkin, F. Seno, R. Metzler, and I. M. Sokolov, Brownian yet non-Gaussian diffusion: From superstatistics to subordination of diffusing diffusivities, *Phys. Rev. X* **7**, 021002 (2017).
- [8] J. Guan, B. Wang, and S. Granick, Even hard-sphere colloidal suspensions display Fickian yet non-Gaussian diffusion, *ACS Nano* **8**, 3331 (2014).
- [9] I. Chakraborty, G. Rahamim, R. Avinery, Y. Roichman, and R. Beck, Nanoparticle mobility over a surface as a probe for weak transient disordered peptide-peptide interactions, *Nano Lett.* **19**, 6524 (2019).
- [10] I. Chakraborty and Y. Roichman, Disorder-induced Fickian, yet non-Gaussian diffusion in heterogeneous media, *Phys. Rev. Res.* **2**, 022020(R) (2020).
- [11] R. Pastore, A. Ciarlo, G. Pesce, A. Sasso, and F. Greco, A model-system of Fickian yet non-Gaussian diffusion: Light patterns in place of complex matter, *Soft Matter* **18**, 351 (2022).
- [12] F. Rusciano, R. Pastore, and F. Greco, Fickian non-Gaussian diffusion in glass-forming liquids, *Phys. Rev. Lett.* **128**, 168001 (2022).
- [13] A. Ciarlo, R. Pastore, F. Greco, A. Sasso, and G. Pesce, Fickian yet non-Gaussian diffusion of a quasi-2D colloidal system in an optical speckle field: Experiment and simulations, *Sci. Rep.* **13**, 7408 (2023).
- [14] C. Yoshina-Ishii, Y.-H. M. Chan, J. M. Johnson, L. A. Kung, P. Lenz, and S. G. Boxer, Diffusive dynamics of vesicles tethered to a fluid supported bilayer by single-particle tracking, *Langmuir* **22**, 5682 (2006).
- [15] R. Das, C. W. Cairo, and D. Coombs, A hidden Markov model for single particle tracks quantifies dynamic interactions between LFA-1 and the actin cytoskeleton, *PLoS Comput. Biol.* **5**, e1000556 (2009).
- [16] M. Ott, Y. Shai, and G. Haran, Single-particle tracking reveals switching of the HIV fusion peptide between two diffusive modes in membranes, *J. Phys. Chem. B* **117**, 13308 (2013).
- [17] S. Thapa, M. A. Lomholt, J. Krog, A. G. Cherstvy, and R. Metzler, Bayesian analysis of single-particle tracking data using the nested-sampling algorithm: Maximum-likelihood model selection applied to stochastic-diffusivity data, *Phys. Chem. Chem. Phys.* **20**, 29018 (2018).
- [18] N. Granik, L. E. Weiss, E. Nehme, M. Levin, M. Chein, E. Perlson, Y. Roichman, and Y. Shechtman, Single-particle diffusion characterization by deep learning, *Biophys. J.* **117**, 185 (2019).
- [19] R. C. Falcao and D. Coombs, Diffusion analysis of single particle trajectories in a Bayesian nonparametrics framework, *Phys. Biol.* **17**, 025001 (2020).
- [20] Q. Xu, L. Feng, R. Sha, N. C. Seeman, and P. M. Chaikin, Subdiffusion of a sticky particle on a surface, *Phys. Rev. Lett.* **106**, 228102 (2011).
- [21] M. J. Skaug, J. Mabry, and D. K. Schwartz, Intermittent molecular hopping at the solid-liquid interface, *Phys. Rev. Lett.* **110**, 256101 (2013).
- [22] J. Bernstein and J. Fricks, Analysis of single particle diffusion with transient binding using particle filtering, *J. Theor. Biol.* **401**, 109 (2016).
- [23] I.-H. Wang, C. J. Burckhardt, A. Yakimovich, and U. F. Greber, Imaging, tracking and computational analyses of virus entry and egress with the cytoskeleton, *Viruses* **10**, 166 (2018).
- [24] P. J. Slator and N. J. Burroughs, A hidden Markov model for detecting confinement in single-particle tracking trajectories, *Biophys. J.* **115**, 1741 (2018).
- [25] A. Callegari, C. Sieben, A. Benke, D. M. Suter, B. Fierz, D. Mazza, and S. Manley, Single-molecule dynamics and genome-wide transcriptomics reveal that NF- $\kappa$ B (p65)-DNA binding times can be decoupled from transcriptional activation, *PLoS Genet.* **15**, e1007891 (2019).

- [26] P. Kowalek, H. Loch-Olszewska, and J. Szwabinski, Classification of diffusion modes in single-particle tracking data: Feature-based versus deep-learning approach, *Phys. Rev. E* **100**, 032410 (2019).
- [27] M. Jensen, Y.-Y. Wang., S. K. Lai, M. G. Forest, and S. A. McKinley, Antibody-mediated immobilization of virions in mucus, *Bull. Math. Biol.* **81**, 4069 (2019).
- [28] M. K. Huseyin and R. J. Klose, Live-cell single particle tracking of PRC1 reveals a highly dynamic system with low target site occupancy, *Nat. Commun.* **12**, 887 (2021).
- [29] T. J. Doerries, A. V. Chechkin, R. Schumer, and R. Metzler, Rate equations, spatial moments, and concentration profiles for mobile-immobile models with power-law and mixed waiting time distributions, *Phys. Rev. E* **105**, 014105 (2022).
- [30] T. J. Doerries, A. V. Chechkin, and R. Metzler, Apparent anomalous diffusion and non-Gaussian distributions in a simple mobile-immobile transport model with Poissonian switching, *J. R. Soc. Interface* **19**, 20220233 (2022).
- [31] F. Simon, J.-Y. Tinevez, and S. van Teeffelen, ExTrack characterizes transition kinetics and diffusion in noisy single-particle tracks, *J. Cell Biol.* **222**, e202208059 (2023).
- [32] T. J. Doerries, R. Metzler, and A. V. Chechkin, Emergent anomalous transport and non-Gaussianity in a simple mobile-immobile model: The role of advection, *New J. Phys.* **25**, 063009 (2023).
- [33] A. Federbush, A. Moscovich, and Y. Bar-Sinai, <https://github.com/ybs-lab/diffusion-with-tethering>.
- [34] L. E. Reichl, *A Modern Course in Statistical Physics*, 4th ed. (Wiley-VCH, Weinheim, 2016), Chap. 7.3.
- [35] Q. Li, Z. Zhu, and G. Tang, in *Proceedings of the 36th International Conference on Machine Learning, Long Beach, 2019*, edited by K. Chaudhuri and R. Salakhutdinov (PMLR, Cambridge, 2019), Vol. 97, pp. 3935–3943.
- [36] A. P. Dempster, N. M. Laird, and D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *J. R. Stat. Soc. B* **39**, 1 (1977).
- [37] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, Cambridge, 2006).
- [38] A. Viterbi, Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, *IEEE Trans. Inf. Theory* **13**, 260 (1967).
- [39] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap* (Chapman & Hall/CRC, Philadelphia, 1994).
- [40] N. Seshadri and C.-E. Sundberg, List Viterbi decoding algorithms with applications, *IEEE Trans. Commun.* **42**, 313 (1994).
- [41] M. Roder and R. Hamzaoui, Fast tree-trellis list Viterbi decoding, *IEEE Trans. Commun.* **54**, 453 (2006).