HHS Public Access

Author manuscript

IFAC Pap OnLine. Author manuscript; available in PMC 2022 December 15.

Published in final edited form as:

IFAC Pap OnLine. 2021 ; 54(20): 340–345. doi:10.1016/j.ifacol.2021.11.197.

Model Segmentation in Single Particle Tracking

Boris I. Godoy[*], **Nicholas A. Vickers**[**], **Sean B. Andersson**[*,**]

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

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

Abstract

In this paper, we implement and compare two different change detection techniques applied to determining the time points in Single Particle Tracking (SPT) data where the particle changes the dynamic model of motion. The goal is to use this change detection to segment the data in order to estimate the relevant parameters of such models. We consider two well-known statistics commonly used for change detection: the likelihood ratio test (LRT) and the Kullback-Leibler divergence (KLD). We assume that our time-varying system is subject to step-like changes in the parameters that drive the process. The techniques are then applied to experimental data acquired on a microscope under controlled settings to validate our results.

Keywords

Modelling; Identification and Signal Processing; Stochastic Systems; Estimation

1. INTRODUCTION

Single Particle Tracking (SPT) is a class of experimental techniques and mathematical algorithms for following small (less than 100 nm) particles moving inside living cells, including viruses, proteins, and strands of RNA Shen et al. (2017).

There are many models in SPT that are used to describe biophysical motion inside the cell. Among the most common are free diffusion, confined diffusion, directed motion, and combinations of these, such as joint diffusion and directed motion Monnier et al. (2012). The most commonly applied technique to estimate the parameters defining these models is (nonlinear) least-squares fitting to the Mean Square Displacement (MSD) curve Manzo and Garcia-Parajo (2015) generated from the noisy estimated trajectory. However, the resulting estimates given by the MSD depend on choices such as the number of points to use in the fit, and the scheme does not account for many factors, including observation noise, motion blur arising from camera integration times, and other experimental realities Berglund (2010). As a result, estimates obtained on the same data can vary greatly from lab to lab. Despite these challenges, the MSD approach remains widely used, despite recent results in the literature demonstrating that they are outperformed by methods based on optimal estimation, such as maximum likelihood (ML) estimation; for further details see e.g. Lin and Andersson (2019).

In general, MSD, and even ML based schemes, assume that the model form is known and that the model parameters, while unknown, are fixed Vega et al. (2018). There have

been some efforts to extend these approaches. For example, in Monnier et al. (2012), they determine the most likely model among a set, but they also consider fixed parameters in the selected model. One of the authors of the present paper has considered time-varying parameters using a jump Markov model Ashley and Andersson (2014) but such models impose a probabilistic structure on the changing parameter values that may be non-physical. In addition, they require a priori knowledge of the number of states, something that in practice is unrealistic to know. In a different approach, the authors have considered the inclusion of time-varying parameters in SPT models using a windowed approach for a simple pure diffusion model that switches the diffusion coefficient at distinct, but unknown times, combined with a strategy of change detection to refine the estimation of the diffusion constants Godoy et al. (2019). Later on, this algorithm was extended to include more general models, and to include nonlinear dynamics, see e.g. Godoy et al. (2020a,b), respectively. This was further developed into a three-stage algorithm, which considered first a windowed approach to provide an initial guess on number of changes of the parameters, followed by a change detection (CD) algorithm to obtain the time points of these changes, and then a refinement of the estimates by using ML on the segmented trajectory Godoy et al. (2021).

This work differs from Godoy et al. (2021) in the following important aspect: here, we compare two different CD algorithms to segment time-varying models typically found in SPT, with the goal of establishing which one works best in the SPT application. This comparison is carried out by using comprehensive simulations and tested with real data.

The literature on CD is vast and these methods have found utility in a wide range of applications. For example, CD has been previously applied to speech processing Gray and Markel (1976); Gray et al. (1980), image processing Baseville (1981), automatic analysis of biomedical signals in electroencephalogram (EEG) Bodenstein and Praetorius (1977); Bohlin (1977); Isaksson et al. (1981) and electrocardiogram Gustafson et al. (1978a,b) data, and geophysics Baseville (1981). The SPT application is particularly challenging for change detection algorithms because the data sets are typically much shorter than commonly seen in CD. In addition, there are limited results for change detection in state-space models where the change occurs nonlinearly in the model (which is the natural setting for SPT motion). We circumvent this issue by calculating an approximation of the system that is amenable to CD and then using well-known statistics for change detection on these simplified models.

The remainder of this paper is organized as follows: in Sec. 2 we provide background on SPT; in Sec. 3, we detail the CD strategies; in Sec. 4 we demonstrate the scheme on experimental data; finally, Sec. 5 gives conclusions and future work.

2. PROBLEM FORMULATION

In SPT, the standard paradigm is to collect a series of images from a CCD camera, localize the particle in each of the images, link them into a trajectory, and then analyze the trajectories for motion model parameters using, typically, a curve fit to an MSD curve. Localization can be done using a variety of different algorithms, ranging from simple centroid calculations through full ML estimation Cheezum et al. (2001); Andersson (2008). The most common remains a fit of the intensity in the image to a simple Gaussian profile,

yielding an accuracy on the order of 10 [nm], well below the diffraction limit of light. An example of one frame of a particular trajectory is given in Fig. 1.

Given a trajectory, analysis typically proceeds by selecting a parameterized model and using the trajectory to identify those parameters. In this work we consider the following quite general model (expressed here in the x-axis only with similar models holding in the other axes):

$$\begin{aligned} x_{k+1} &= a_t x_k + b_t + w_k,\ w_k \sim \mathcal{N}(0, q_t) \\ y_k &= x_k + v_k,\ v_k \sim \mathcal{N}(0, r_t), \end{aligned} \tag{1}$$

where $x_k, y_k, w_k, v_k \in \mathbb{R}$, $q_t = 2D_t\Delta t$ is the variance of the process noise defined by the diffusion coefficient D_t and the sampling time Δt, and r_t is the variance of the measurement noise as generated by a variety of processes, including shot noise due to the physics of photon generation in fluorescence and read-out noise in the camera. System (1) is a generalization that can be used to represent several models in the SPT application. For example, setting $a_t = 1$, $b_t = 0$ describes pure diffusion; $a_t < 1$, $b_t = 0$ yields the Ornstein-Uhlenbeck model that can capture tethered motion of a biomolecule of be used to approximate confined diffusion Calderon (2016). For later reference, we define the unknown parameter in (1) as $\beta = [a_t\ b_t\ q_t\ r_t]$.

The model in (1) also allows us to have time-varying parameters. For example, the process noise q_t can vary as the particles move into different environments of the cell with different diffusion coefficients, interact with different species in their surroundings, or go through biochemical changes due to the natural activity in the cell. These time-varying parameters will allow us to represent different motion modes that might be happening during a whole trajectory. For notation purposes, we add subindex t to a parameter to indicate that it is time-varying, and remove the subindex t when it is presumed to be time-invariant.

Despite the complex reality of measurement noise, it is often modeled as a simple Gaussian white noise process. The Gaussian approximation works well at high signal-to-noise levels but does break down at low intensities, necessitating a more complex description of the data Ashley and Andersson (2015). In general, the measurement noise r_t has reasonably static statistics as it is produced by the experimental equipment. However, r_t may also arise from non-uniform background or changes in illumination intensity during the measurements.

Our goal, then, is to compare and choose a change detection algorithm to split a trajectory into segments, each of which may be described by a different set of model parameters. We note that the experiment is blind, that is, no prior information is known about the particle besides the assumption that it can be described by a linear model such as in (1). Once the segmentation has been carried out, it is possible to estimate the model associated with each portion (or segment) of the data.

3. CHANGE DETECTION

Change detection (CD) is a widespread topic in the systems and control community. Much research has been done, mainly in the 70s and 80s, see e.g. Basseville and Benveniste (1986); Basseville and Nikiforov (1993) and the references therein. CD can be carried out online or offline, and used to detect additive and non-additive changes. The problem of change detection can be defined as follows:

Definition 1. Given a sequence of random variables y_n from a probability density function (pdf) $p_\theta(y_n)$, *change detection* aims to find the unknown time t_c such that the parameter $\theta = \theta_o$ for $t < t_c$, and $\theta = \theta_1 (\neq \theta_o)$ for $t >= t_c$.

The above question (finding t_c) can be answered online or offline. Either way, one needs to know the value of the parameter before the change occurs. If this is not possible, then estimates of these parameters can be used. To do offline CD, we also need to know the value of the parameter after the change has occurred. For online estimation, the value of θ_1 (after change) is not necessary to be known a priori, because it can also be estimated.

The general approach to CD begins with the generation of a *residual* or change (indicating) signal. This signal is ideally close to zero when there is no change, and starts growing when a change happens. Secondly, CD requires the creation of a *decision rule*, that is, a detector which monitors the residual signal to look for changes. There are a few qualities that are expected from a CD algorithm, in particular, (i) few false alarms, (ii) quick detection, and (iii) symmetry in the detection, that is, the algorithm can detect a change from model A to B and from B to A. Note that in many cases one is willing to accept some amount of bias in the detection to achieve these qualities.

There are applications where it is necessary to monitor a system online. However, in our problem, and since we deal with batch-mode data, we can focus on the offline case. Also, for our system, we aim to detect changes that are nonadditive (spectral changes), for instance, a change in the diffusion constant D (consequently in the variance q in model (1)), as well as a change in the dynamics of the system, that is, a. A sufficient statistic to detect nonadditive changes is the CUSUM algorithm, based on the likelihood ratio test (LRT) or on the Kullback-Leibler divergence (KLD) see e.g. Basseville and Benveniste (1983).

3.1 Application of the change detection algorithm

To apply the CUSUM algorithm, we define a signal based on the LRT:

$$s_k = \log \frac{p_{\theta_1}(y_k \mid Y_{k-1})}{p_{\theta_o}(y_k \mid Y_{k-1},)},$$
$$g_k = (g_{k-1} + s_k)^+, \tag{2}$$

Where Y_{k-1}, is the vector containing data up to $k-1$, $(a)^+ = \max\{a, 0\}$, and θ_o and θ_1 are the parameter values before and after the change, respectively. In reality, we do not know the exact values of θ_o and θ_1, hence, we need to use estimates of their values in generating the

signals in (2). These estimates are generated using a sliding window approach as described in Sec. 3.4 below.

The CD algorithm is then defined by a threshold λ defined by the user such that

$$\hat{t}_c = \min\{k : g_k \geq \lambda\}. \tag{3}$$

The choice of λ determines the sensitivity of the algorithm and is a crucial design parameter.

An alternative to the LRT is the KLD measure. This measure takes into consideration the self-entropy of the pdfs before and after the change. In this case, the CUSUM test is based on the following expression:

$$\begin{aligned} \bar{s}_k &= \int p_{\theta_0}(y \mid Y_{k-1}) \log \frac{p_{\theta_1}(y_k \mid Y_{k-1})}{p_{\theta_0}(y_k \mid Y_{k-1})} dy - \log \frac{p_{\theta_1}(y_k \mid Y_{k-1})}{p_{\theta_0}(y_k \mid Y_{k-1})} \\ \bar{g}_k &= (\bar{g}_{k-1} + \bar{s}_k)^+. \end{aligned} \tag{4}$$

This test has certain advantages over the CUSUM algorithm based on the LRT. For example, it is less sensitive to noise, and in general, it delivers cleaner (less noisy) residuals. However, the major disadvantage is that it requires a good model to fit the data.

We note that there are also other methods not based on the LRT, collectively referred to as non-likelihood-based algorithms. The advantage of these methods is that they are simpler than those based on the likelihoods, and they are efficient from the statistical inference point of view, see e.g. Basseville and Nikiforov (1993). However, their consideration is left for future work.

3.2 Approximation using an ARX model

The CD approach relies on a model that defines the pdf relating the parameters to the data. While (1) is our fundamental description, such a model does not lend itself well to CUSUM methods. However, autoregressive (ARX) and Autoregressive-moving average (ARMAX) models are useful for detection of spectral changes. This has been demonstrated for various digital signals, see e.g. Markel and Gray (1976); Isaksson et al. (1981). The idea is to look for changes in the residual innovation signal (e_k), which is defined as $e_k = y_k - \hat{y}_{k \mid k-1}$, and $\hat{y}_{k \mid k-1}$ is the 1-step ahead predictor. The ARX approximation for the measurements y_k will have the following form

$$y_k \approx \sum_{i=1}^{p} a_i^j y_{k-i} + \epsilon_k, \ \mathrm{var}\{\epsilon_k\} = \sigma_\epsilon^2. \tag{5}$$

We define $a^j = (a_1^j, \ldots, a_p^j)$. When there is no change, data will be approximated with $j = 0$, and when there is a change, then data is approximated with $j = 1$. For details, see e.g. Basseville and Benveniste (1983).

For data produced by model (1), we have found that an ARX model provides a sufficiently good approximation of such data for CD purposes. To select the order of the model, we turn to the Bayesian information criterion (BIC) as described next.

3.3 Using BIC for model selection

Once a type of model has been chosen, then we need to select the order of such model. To that end, we can apply either the Akaike or the Bayesian information criterion (AIC) or (BIC). In general, the BIC favours low order models compared to AIC. It is defined as:

$$\mathrm{BIC} = -2l(\theta) + d \log N, \tag{6}$$

where N is the number of data points, d is the order of the model, and $l(\theta)$ is the log-likelihood associated with the estimation problem. For the ARX case, the log-likelihood is defined as:

$$-2l(\theta) = N \log(2\pi) + N \log\left(\sigma_\epsilon^2\right) + \sum_{k=1}^{N} \left(\frac{\left(y_k - \sum_{i=1}^{p} a_i^{(k)} y_{k-i}\right)^2}{\sigma_\epsilon^2} \right) \tag{7}$$

where $\theta = \left[a_1 \ a_2 \ldots a_p \ \sigma_\epsilon^2\right]$. The estimation of θ is then calculated as:

$$\hat{\theta} = \arg\max_{\theta} l(\theta). \tag{8}$$

(Note that θ in (6) refers to the parameters defining the ARX approximation (5) (1), and not the unknown parameter β of the original model (1).) To explore the model order needed, we generated 50 sample paths using the model in (1). We then applied the BIC to determine the optimal ARX model order for different data lengths (by simply truncating the sample paths to the desired length). The results of this model selection is shown in Fig. 2. These results indicate that an approximation with a low order ARX model works well, typically with only 1–2 parameters. Finally, we note that while it is common in CD to have much longer data sets than considered here, in the SPT setting typical trajectories are often on the order of 100–1000 data points.

3.4 Putting it all together

Let us consider a stochastic proces y_n with conditional distribution $p_\theta(y_t|Y_{t-1})$, where $Y_{t-1} = \{y_{t-1}, \ldots, y_1\}$. Also consider a set of data (measurements) $\{Y_t\}_{0\leq t\leq n}$, the problem is to decide between two hypothesis:

$$\begin{aligned} &\mathscr{H}_o: \theta = \theta_o, \\ &\mathscr{H}_1: \theta = \theta_o, \quad \text{for } 0 \leq t \leq r-1, \\ &\theta = \theta_1, \quad \text{for } r \leq t \leq n. \end{aligned} \tag{9}$$

To answer this question, the idea is to have two ARX models that we can compare against each other. The first one is a long-term model, M_{θ_o}, that adds a sample at each iteration. The second one is a short-term model, M_{θ_1}, which gets updated by adding new data in a sliding window approach. Then, both models are compared using a change detection algorithm. Once a change is detected, then the long-term model is reset to a new M_{θ_o} and the oldest section of the data is discarded.

Change detection is done on the residuals e_k^0 and e_k^1 from the models M_{θ_o} and M_{θ_1}, respectively. The form of the CUSUM test based on the LRT in (2) for an ARX model is

$$s_k = \frac{1}{2}\log\frac{\sigma_0^2}{\sigma_1^2} + \frac{(e_k^0)^2}{2\sigma_0^2} - \frac{(e_k^1)^2}{2\sigma_1^2}, \quad \bar{g}_k = \sum_{i=1}^{n} s_k, \tag{10}$$

where $\epsilon^o \sim \mathcal{N}(0, \sigma_0^2)$ and $\epsilon^1 \sim \mathcal{N}(0, \sigma_1^2)$, respectively. The test is determined by choosing a value for λ and implementing (3) for g_k.

Similarly, the CUSUM test based on the KLD in (11) for an ARX model is

$$\bar{s}_k = \frac{1}{2}\left(2\frac{e_k^0 e_k^1}{\sigma_1^2} - \left(1 + \frac{\sigma_0^2}{\sigma_1^2}\right)\frac{(e_k^0)^2}{\sigma_0^2} + \left(1 - \frac{\sigma_0^2}{\sigma_1^2}\right)\right), \quad \bar{g}_k = \sum_{i=1}^{n} \bar{s}_k. \tag{11}$$

The test is determined by choosing a value for λ and implementing (3) for $\bar{g}_k$.

One methodological novelty considered in this work is to apply a *backward* change detection. The definition of this concept is as follows:

Definition 2. For data in batch mode Y_N, a *backward* change detection problem is defined as usual over data Y_N, but inverting the original data to $\bar{Y}_N$ where:

$$\bar{Y}_N = \{y_N, y_{N-1}, \ldots, y_1\}, \tag{12}$$

That is, now the last original measurement Y_N corresponds to the first new one $\bar{y}_1$, and the first original measurement y_1 to the last new one $\bar{Y}_N$. The above definition allows us to run the CD algorithm in two ways, forward and backward. Thus, for every change, there will be two different estimates of time of change t_c, each with their own bias. These can then be averaged to reduce the net bias. Note that since the bias depends on the direction of the change (that is increasing versus decreasing parameter values), this approach is not expected to completely eliminate bias.

So far, we have proposed two possible ways to use CD techniques, namely by using the LRT and the KLD for the calculation of the decision rule. To test the efficacy of the two methods, we ran simulation trials by generating data according to the model in (1) using the model parameter values in Table 1, with $r_O = 0.01$. The time step was set to 0.1 s, the window size $h = 150$, and the time length to 75 s (750 steps). Notice that in this case, we have set up the simulated data with a strong SNR (about 10x), which would be a typical scenario in SNR experiments.

The first step is to tune the thresholds for CD. To do this, we generated 30 data runs with no change in the model (using the parameters in the first column of Table 1). The two CD tests were applied to these and thresholds of 5 for the LRT decision rule and 2.5 for the KLD one were selected to avoid false positive detections.

We then generated a second set of 30 data runs, now with a parameter change in D at 30 [s] (time step 300) as shown in Table 1. To quantify performance we used the metrics off Precision (P) and Recall (R) given by

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN},$$

where TP stands for true positives, FP for false positives, and FN for false negatives. The standard Confusion Matrix is shown in Table 2) This yields the metrics

$$\begin{aligned} P_{LRT} &= 0.85, \quad R_{LRT} = 0.97, \\ P_{KLD} &= 0.67, \quad R_{KLD} = 0.20 \end{aligned} \tag{13}$$

These results clearly indicate that the LRT is both more sensitive and more reliable as a detection signal than the KLD for our application. Note that while in general the KLD yields a smoother response Basseville and Benveniste (1983), this is under the assumption that the ARX model is an accurate representation of the data. In our SPT scenario, the ARX is a simplification for CD. In addition, the length of the data sets are relatively small.

4. DEMONSTRATION OF THE ALGORITHM

This section is divided into three parts, namely: (A) data generation; (B) detecting changes in the diffusion constant; (C) detecting changes in the dynamics of the model.

4.1 Data Generation

Synthetic motion is a method of experimentally testing the performance of single particle tracking microscopes and estimation algorithms with known ground truth. To do this, we first use the desired motion specification which includes the motion model parameters and the time point that they change. From this specification, a dictionary of trajectories are generated using the stochastic motion model. These trajectories are then used to control the position of a fixed fluorescent microsphere using a piezo actuated nano-positioning microscope stage. The stage is moved and allowed to settle during the read out of the camera image to ensure that the microsphere is still during image acquisition. Using this method allows control over the resulting motion, with repeatable and known motion model parameters, and control over the time point at which the parameters change, making it an ideal test for change detection algorithms.

For this experiment we generated data according to (1) with a total of 750 positions at a frequency of 20 Hz. The trajectories were converted into a control voltage by a National Instruments compact reconfigurable input output (NI cRIO 9076) device programmed in Labview, and sent to a Mad City Labs Nano-Drive stage amplifier which controls a Nano-PDQ350HS high speed piezo actuated microscope stage. Green fluorescent microspheres were fixed onto a microscope slide coverslip by spin coating, then the coverslip was fixed to the microscope stage. Imaging was done using a Zeiss Axiovert 200 equipped with a 63× 1.2 NA water immersion objective and a photometrics 95B sCMOS digital camera. Images were taken at a frame rate of 20 Hz and an integration time of 25 ms. The location of the particle in each frame was determined by fitting the intensity profile to a Gaussian function, a standard approach in SPT.

In these experiments, model changes were made at time 12.5 [s] (250 steps) and 25 [s] (500 steps). At the change point, only one parameter was modified, either the diffusion coefficient or the dynamic parameter a. The amount of data collected is $N = 750$ frames.

4.2 Change in the diffusion constant (D)

True values of a_t and b_t for both cases were fixed to $a_t = 1$ and $b_t = 0$. The diffusion coefficients began at 0.1 [$\mu m^2/s$], then switched to 0.01 [$\mu m^2/s$] at time 12.5 [s], and then again back to 0.1 [$\mu m^2/s$] at time 25 [s]. 10 sets of data were acquired.

Boxplots of the detected change points under LRT and KLD are shown in Fig. 4 and Fig. 5 respectively. Each of the figures also shows the detected time of change based on the forward pass (FP), the backward pass (BP), and the average of both (AP) for both statistics. From these results we observe that, at least in these runs, the AP exhibits less bias under both statistics, as expected. In addition, the LRT outperformed the KLD in terms of variance, though both performed quite well.

4.3 Change in the dynamics a

In the second set of experiments, the diffusion coefficient was fixed to 0.01 [$\mu m^2/s$]. The parameter a_t was changed from 1 to 0.86 at time 12.5 [s], and then again back to 1 at time 25

[s]. Once again, ten sets of data were acquired. The results of this estimation can be seen in Fig. 6 for the LRT and in Fig. 7 for the KLD.

As with the change in diffusion coefficient, both the LRT and the KLD are effective at finding the change. Once again, the LRT shows smaller variance than the KLD, indicating it is the preferred choice in this application.

5. CONCLUSIONS

We have compared two of the most utilized change detection techniques, namely the CUSUM algorithm based on the LRT and on the KLD. Our work shows that while both statistics are able to detect changes in SPT, the LRT outperforms in terms of sensitivity and robustness and in terms of the variance of the change point estimate. This can be explained by the fact that, in general, we have very short data sets, and the KLD decision rule seems to require a highly accurate model for the data in order to perform well.

Our analysis shows that this technique is well suited for segmenting SPT data that contain runs with abrupt changes in model parameters, all without prior information about the number of changes or the model parameters. It also has the advantage of a straightforward extension to models more complex than the ARX one chosen here and it is the subject of ongoing research to explore this avenue.to detect changes in models that can be more complex than a simple ARX.

Acknowledgments

This work was supported in part by the NIH through 1R01GM117039-01A1.

REFERENCES

Andersson SB (2008). Localization of a fluorescent source without numerical fitting. Optics Express, 16(23), 18714–18724. [PubMed: 19581957]

Ashley TT and Andersson SB (2014). A Sequential Monte Carlo Framework for the System Identification of Jump Markov State Space Models. In American Control Conference, 1144–1149.

Ashley TT and Andersson SB (2015). Method for simultaneous localization and parameter estimation in particle tracking experiments. Physical Review E, 92(5), 052707.

Baseville M (1981). Edge detection using sequential methods for change in level. Part II: Sequential detection of change in a mean. IEEE Trans. Acous., Speech, Sig. Proc, 29, 32–50.

Basseville M and Benveniste A (1983). Sequential detection of abrupt changes in spectral characteristis of digital signals. IEEE Trans. on Inf. Theory, 29(5), 709–724.

Basseville M and Benveniste A (eds.) (1986). Detection of abrupt changes in Signals and Dynamical Systems. Springer-Verlag.

Basseville M and Nikiforov I (1993). Detection of Abrupt changes: Theory and Application. Prentice Hall.

Berglund AJ (2010). Statistics of camera-based single-particle tracking. Physical Review E, 82(1), 011917.

Bodenstein G and Praetorius H (1977). Feature extraction from the encephalogram by adaptive segmentation. IEEE Proc., 65, 221–259.

Bohlin T (1977). Analysis of EEG signals with changing spectra using a short word Kalman estimator. Math Biosci., 35, 221–259.

Calderon CP (2016). Motion blur filtering: A statistical approach for extracting confinement forces and diffusivity from a single blurred trajectory. Physical Review E, 93(5).

Cheezum MK, Walker WF, and Guilford WH (2001). Quantitative comparison of algorithms for tracking single fluorescent particles. Biophysical Journal, 81(4), 2378. [PubMed: 11566807]

Godoy BI, Lin Y, Agüero JC, and Andersson SB (2019). A 2-step algorithm for the estimation of single particle tracking models using maximum likelihood. In 12th Asian Control Conf. (ASCC). Kitakyushu, Japan.

Godoy BI, Lin Y, and Andersson SB (2020a). A time-varying approach to single particle tracking with a nonlinear observation model. In American Control Conf. (ACC). Denver, CO, USA.

Godoy BI, Vickers N, Lin Y, and Andersson SB (2020b). Estimation of general time-varying single particle tracking linear models using local likelihood. In European Control Conf. (ECC). Saint Petersburg, Russia.

Godoy BI, Vickers NA, and Andersson SB (2021). An estimation algorithm for general linear single particle tracking models with time-varying parameters. Molecules, 26(4).

Gray A and Markel J (1976). Distance measures for speech processing. IEEE Trans. Acous., Speech, Sig. Proc, 24(5), 380–437.

Gray R, Buzo A, Gray A, and Matsuyama Y (1980). Distorion measures for speeach processing. IEEE Trans. Acous., Speech, Sig. Proc, 28(4), 367–376.

Gustafson D, Willsky A, Wang J, Lancaster M, and Trieb-wasser J (1978a). ECG/VCG rhythm diagnosis using statistical signal analysis, Part I: Identification of persistent rhythms. IEEE Trans. Biomedical Eng, 25(4), 344–353.

Gustafson D, Willsky A, Wang J, Lancaster M, and Trieb-wasser J (1978b). ECG/VCG rhythm diagnosis using statistical signal analysis, Part II: Identification of transient rhythms. IEEE Trans. Biomedical Eng, 25(4), 353–361.

Isaksson A, Wennberg A, and Zetterberg L (1981). Computer analysis of EEG signals with parametric models. IEEE Proc., 69(4), 451–461.

Lin Y and Andersson SB (2019). Simultaneous localization and parameter estimation for single particle tracking via sigma points based em. In 58th IEEE Conf. on Decision and Control (CDC). Nice, France.

Manzo C and Garcia-Parajo MF (2015). A review of progress in single particle tracking: from methods to biophysical insights. Reports on Progress in Physics, 78(12), 124601. [PubMed: 26511974]

Markel J and Gray A (1976). Linear prediction of Speech. New York: Springer-Verlag.

Monnier N, Guo SM, Mori M, He J, Lénárt P, and Bathe M (2012). Bayesian Approach to MSD-Based Analysis of Particle Motion in Live Cells. Biophysical Journal, 103(3), 616–626. [PubMed: 22947879]

Shen H, Tauzin LJ, Baiyasi R, Wang W, Moringo N, Shuang B, and Landes CF (2017). Single Particle Tracking: From Theory to Biophysical Applications. Chemical Reviews, 117(11), 7331–7376. [PubMed: 28520419]

Vega AR, Freeman S, Grinstein SA, and Jaqaman K (2018). Multistep track segmentation and motionclassification for transient mobility analysis. Biophysical Journal, 114, 1018–1025. [PubMed: 29539390]

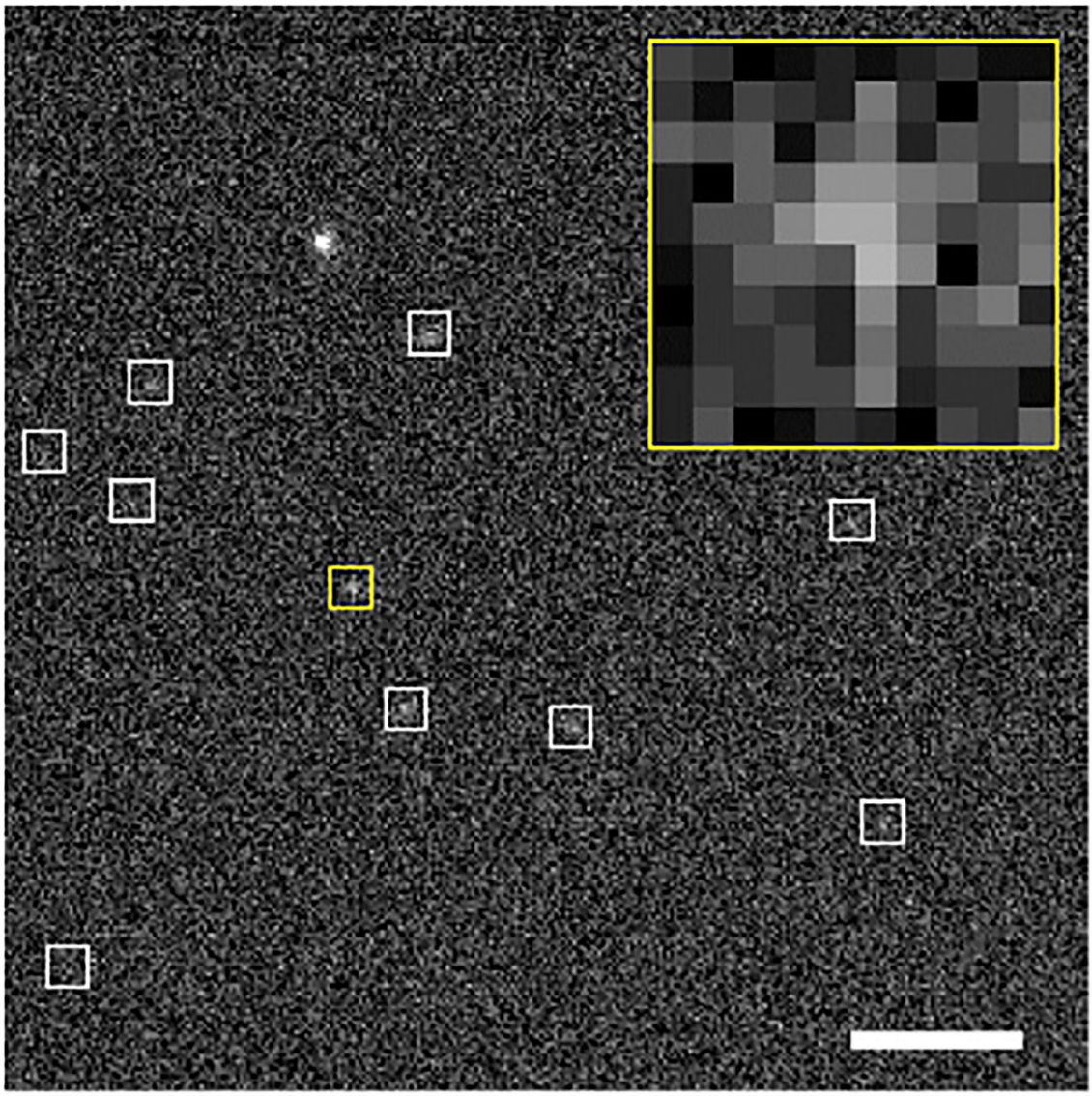

Fig. 1.
Scale bar is 8 [μm], inset is 1.74 [μm]. Each pixel is 174 [nm] wide.

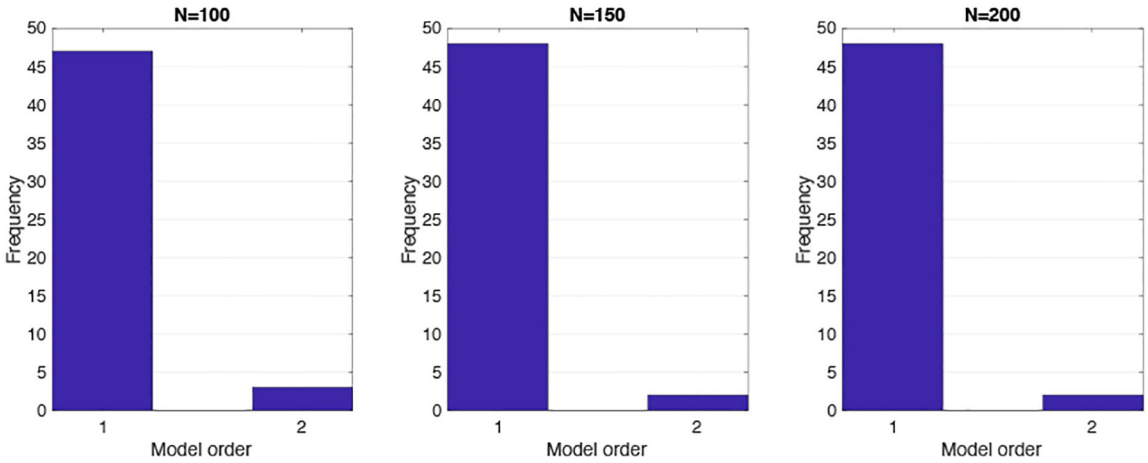

Fig. 2.

Optimal models orders under the BIC using 50 sample paths of the same model using data of length (left) $N = 100$, (center) $N = 150$, and (right) $N = 200$.

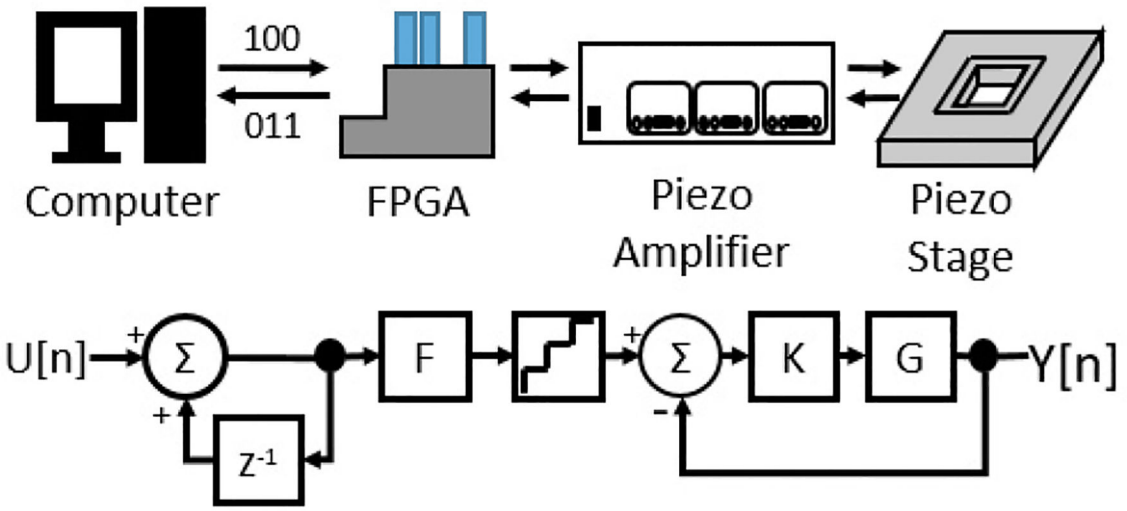

Fig. 3.

A block diagram showing the signal flow. (Upper left) computer generated realizations of our model (bottom) is sent to the controller and is translated to an analog control voltage. This control voltage is sent to the stage amplifier which then generates the high voltage needed to generate motion in the stage.

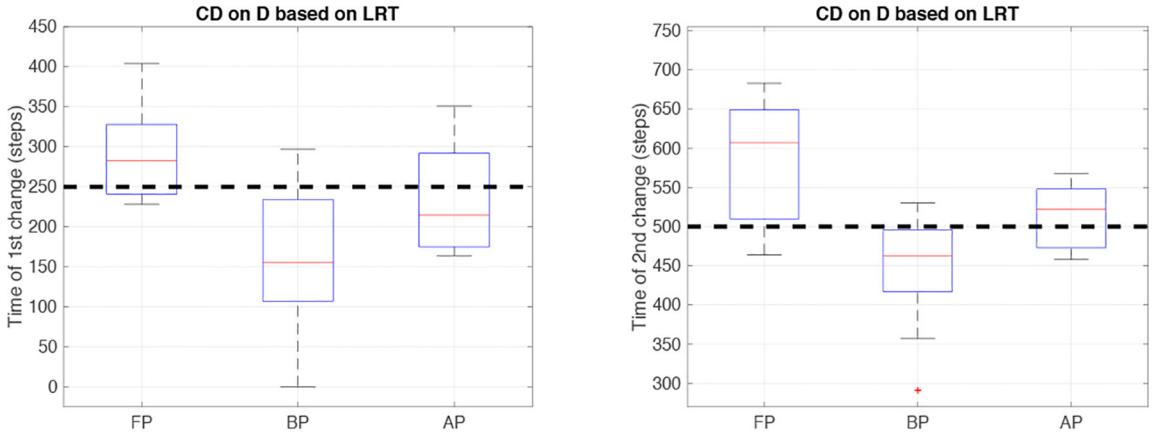

Fig. 4.

Case: Change in D using LRT. Boxplots showing the estimation for time of change t_c. From left to right we show the forward pass (FP), backward pass (BP), and the average of both (AP), respectively. Real value black-dashed line.

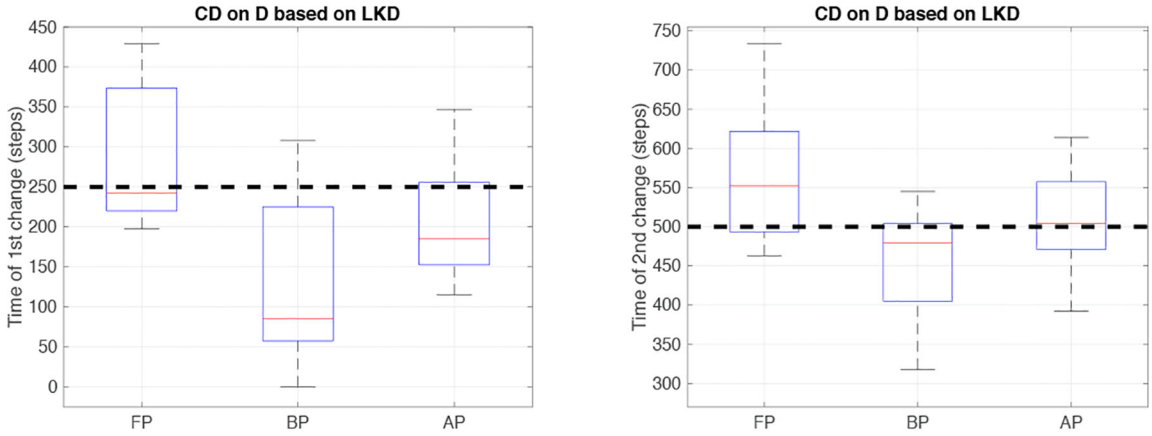

Fig. 5.

Case: Change in D using KLD. Boxplots showing the estimation for time of change t_c. From left to right we show the forward pass (FP), backward pass (BP), and the average of both (AP), respectively, for the (left plot) first and (right plot) second change point. Real value black-dashed line.

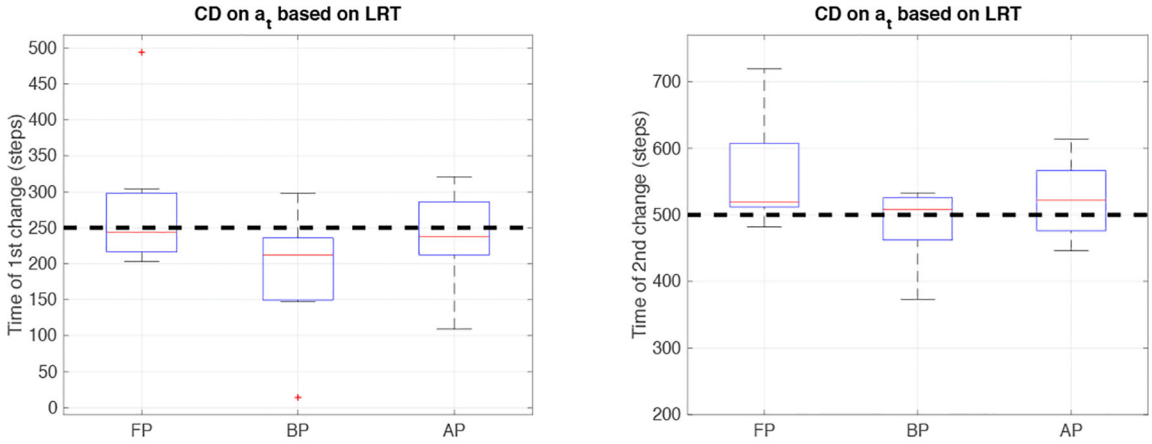

Fig. 6.

Case: Change in a using LRT. Boxplots showing the estimation for time of change t_c. From left to right we show the forward pass (FP), backward pass (BP), and the average of both (AP), respectively. Real value black-dashed line.

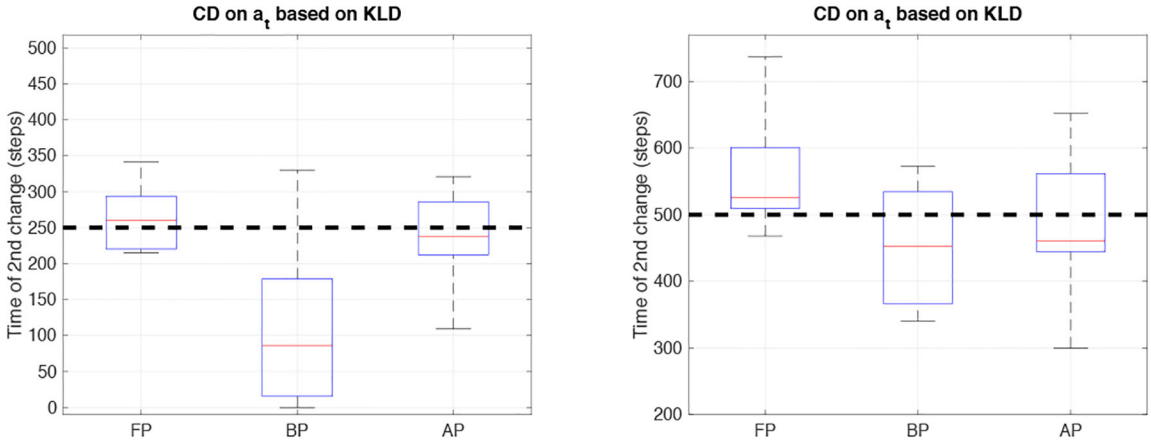

Fig. 7.

Case: Change in a using KLD. Boxplots showing the estimation for time of change t_c. From left to right we show the forward pass (FP), backward pass (BP), and the average of both (AP), respectively, for the (left plot) first and (right plot) second change point. Real value black-dashed line.

Table 1.

Parameter set for simulated data

parameter	$t < 25.0$ [s]	$t > 25.0$ [s]
a_t	1	1
b_t	0	0
$D_t = \frac{Q_t}{2\Delta t}$	1	4
r_t	r_o	r_o

Table 2.

Confusion Matrix.

	LRT			KLD		
Pred \ Actual	**Change**	**No change**		**Change**	**No change**	
Change	29	5	34	6	3	9
No change	1	25	26	24	27	51
	30	30	60	30	30	60