Bioinformatics, 36(1), 2020, 317–329
doi: 10.1093/bioinformatics/btz489
Advance Access Publication Date: 19 June 2019
Original Paper

Bioimage informatics

A sequential algorithm to detect diffusion switching along intracellular particle trajectories

Vincent Briane[1,2], Myriam Vimond[2], Cesar Augusto Valades-Cruz[3,4], Antoine Salomon[1], Christian Wunder[3] and Charles Kervrann[1,4,*]

[1]INRIA, Centre de Rennes Bretagne Atlantique, Serpico Project-Team, Rennes 35042, France, [2]CREST (Ensai, Université Bretagne Loire), Bruz 35170, France, [3]Institut Curie, PLS Research University, Cellular and Chemical Biology, U1143 INSERM/UMR 3666 CNRS, 26 Rue d'Ulm, Paris Cedex 05 75248, France and [4]Institut Curie - Centre de Recherche, PLS Research University, Space-Time Imaging of Endomembranes and Organelles Dynamics Team, 26 rue d'Ulm, Paris Cedex 05 75248, France

*To whom correspondence should be addressed.

Associate Editor: Robert Murphy

Received on October 14, 2018; revised on May 10, 2019; editorial decision on May 23, 2019; accepted on June 12, 2019

Abstract

Motivation: Recent advances in molecular biology and fluorescence microscopy imaging have made possible the inference of the dynamics of single molecules in living cells. Changes of dynamics can occur along a trajectory. Then, an issue is to estimate the temporal change-points that is the times at which a change of dynamics occurs. The number of points in the trajectory required to detect such changes will depend on both the magnitude and type of the motion changes. Here, the number of points per trajectory is of the order of 10^2, even if in practice dramatic motion changes can be detected with less points.

Results: We propose a non-parametric procedure based on test statistics computed on local windows along the trajectory to detect the change-points. This algorithm controls the number of false change-point detections in the case where the trajectory is fully Brownian. We also develop a strategy for aggregating the detections obtained with different window sizes so that the window size is no longer a parameter to optimize. A Monte Carlo study is proposed to demonstrate the performances of the method and also to compare the procedure to two competitive algorithms. At the end, we illustrate the efficacy of the method on real data in 2D and 3D, depicting the motion of mRNA complexes—called mRNA-binding proteins—in neuronal dendrites, Galectin-3 endocytosis and trafficking within the cell.

Availability and implementation: A user-friendly Matlab package containing examples and the code of the simulations used in the paper is available at http://serpico.rennes.inria.fr/doku.php?id=software:cpanalysis:index.

Contact: charles.kervrann@inria.fr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

High-resolution fluorescence imaging have made possible the observation of the dynamic behaviour of single molecules. The study of these new data with state-of-the-art statistical and image analysis techniques will allow to decipher the dynamic coordination and organization of interacting molecules, which determine the different functions of the cell, see Kervrann *et al.* (2016) and references therein. Indeed living cells are dynamic structures and their constituent particles are constantly moving within and between cellular compartments, domains and microdomains. Tracking algorithms (Chenouard *et al.*, 2014; Maroulas *et al.*, 2015; Roudot *et al.*, 2017) allow to reconstruct frame by frame the trajectory of a particle over a fixed time period. This trajectory is a time series describing the behaviour of the particle which can change over time. In this context, change-point detection (see Truong *et al.*, 2018 for

a review) is an importance task since change-points are often indicative of a new interaction of the particle with another component of the cell.

In biophysics, the dynamics of these trajectories are usually classified into three groups: subdiffusion, superdiffusion and Brownian motion. The definition of these dynamics is related to the criterion of the mean square displacement, a function of time widely used in biophysics and in cellular imaging to quantify motion. If the mean square displacement (MSD) function is linear, the trajectory is Brownian (Qian *et al.*, 1991). If the MSD is sublinear (respectively, superlinear) the trajectory is a subdiffusion (respectively, a superdiffusion) (see Bressloff, 2014, Chapter 7; Metzler and Klafter, 2000). The biological interpretation of subdiffusion is that the particle is confined in a domain or evolves in an open but crowded area (Berry and Chaté, 2014; Bressloff and Newby, 2013, Section 3). Superdiffusion occurs when the particle is transported actively via molecular motors along the microtubules (Bressloff and Newby, 2013, Section 4). Finally, when the particle evolves freely inside the cytosol, it undergoes Brownian motion (Bressloff and Newby, 2013, Section 2).

There exist several examples of proteins whose dynamic mode switches over time. Transmembrane proteins such as AMPA receptors oscillate between subdiffusion and Brownian motion (Hoze *et al.*, 2012). As another example, a virus invading a cell switches motion between superdiffusion along microtubules and Brownian motion in the cytosol (Lagache *et al.*, 2009). Here, we focus on the family of Galectins that is known to be involved in important physiological processes such as immune response, cellular development and cancer progression. Galectins act through binding to specific carbohydrates on intracellular and extracellular proteins and lipids. We study trafficking of one member of this lectin family, Galectin-3 (Gal-3). Extracellular Gal-3 is able to bind to the plasma membrane and gets internalized via the formation of vesicular transport carriers (Lakshminarayan *et al.*, 2014). Those endocytic transport carriers are able to fuse with cellular structures, first with early endosomes, and later on with recycling endosomes or late endosomes. New transport carriers are formed via a fission process, when Gal-3 leaves one of those cellular structures. Movement of Gal-3 containing transport vesicles between those compartments is facilitated with the help of motor proteins. Therefore trafficking of Galectins within the cellular environment can be seen as a constant switching between different types of motion. Estimating the change-points and identifying the different modes of motions between these change-points is essential to characterize the trafficking behaviour of the Gal-3 trajectories. It is worth noting that switch of motion also occurs at the scale of the cell during cell migration. Pankov *et al.* (2005) suggest that a change in the signalling G protein Rac1 activity can lead to a switch between random versus directionally persistent cell migration. Here, we will only focus in intracellular trafficking.

In this paper, we derive an algorithm to estimate the times at which the particle changes from one type of diffusion (superdiffusion, subdiffusion or Brownian motion) to another type of diffusion. This algorithm is based on the statistical test procedure of Briane *et al.* (2018). Indeed, Briane *et al.* (2018) defined a non-parametric three-decision test to distinguish the three types of diffusion and showed that their procedure is consistent under parametric alternatives. Now, the main principle of the proposed algorithm takes its inspiration from the sequential procedure of Cao and Wu (2015) and can be summarized as follows. Using a sliding window, we detect a candidate change-point as a point having a high difference between two test statistics which are computed in a local neighbourhood of the point. Then we define clusters of candidate change-points. For each cluster, our change-point estimate is chosen as the change-point candidate having the highest difference between the two local test statistics. An aggregation strategy is proposed to combine the detections obtained with different window sizes. Then the aggregated procedure does no longer depend on the size of the window, which is a critical parameter in window-based methods.

The present paper is organized as follows. In Section 2 we present the state of the art about change-point detection applied to biophysics. In Section 3, we exhibit the inference model and recall the test procedure of Briane *et al.* (2018). In Section 4, we present our sequential procedure for detecting change-points along a trajectory. In Section 5, we assess the performance of the procedure on Monte Carlo simulations and on real 2D and 3D data, depicting the motion of mRNA complexes—called mRNA-binding proteins (mRNP)—in neuronal dendrites, and Gal-3 endocytosis and movement within the cell.

2 State of the art

In this section, we review the methods dealing with multiple change-point detection problem applied to biophysics. First, some approaches use a sliding window to detect a dynamic change in the trajectory. Simson *et al.* (1995) and Meilhac *et al.* (2006) propose a procedure that computes locally the largest displacement from the starting point of each segment or a confinement index based on the MSD; below a critical threshold, a short segment will be labelled as a confinement zone. The parameters of Simson *et al.* (1995) are tuned through both experimental and simulation data making the method hard to tackle general problems. To circumvent this problem, Meilhac *et al.* (2006) assume that the particle diffuses in a square box of particular length. Then the settings of parameters is rigorous but the motion model is not flexible. The algorithm of Bouzigues and Dahan (2007) is based on the computation of a speed correlation index on sliding windows in order to detect directed motion from Brownian motion. Again the thresholding step of the method is obtained by simulation and is highly dependent on the values of the parameters like the diffusion coefficient, the drift velocity and even the duration of directed motion. Arcizet *et al.* (2008) fit locally the MSD to the power law in order to segment the trajectory into subtrajectories driven by either Brownian motion or directed Brownian motion. The major drawback of this method is that it assumes a parametric form of the MSD which does not fit simple case like Brownian motion with drift. Vega *et al.* (2018) use an initial window-based approach to segment the trajectories and then a decision tree to merge some detected segments. The calibration of their parameters is based on simulation. Hence parameter values are sensitive to the models used in the simulation. A general issue with these window-based approach is that they are very sensitive to the choice of the window size.

Other methods are based on feature parameter classification using a supervised support vector classification (Helmuth *et al.*, 2007) or back-propagation neural network (Dosset *et al.*, 2016). In this context, the critical step is the training of the machine-learning algorithm. They both use simulated data for this purpose. As pointed out by Helmuth *et al.* (2007), the trajectories used for training the algorithm must cover all the different regimes in the feature space. Due to the diversity of both dynamics and type of transitions, we argue that representing this general feature space without additional assumptions is intractable. Then the machine-learning algorithm will have good results only on the subset described by the training data.

Finally, some approaches assume a parametric model of motion. Yin *et al.* (2018) use log-likelihood ratio tests to divide the trajectory into segments that have different diffusion coefficients and/or velocities; implicitly each segment is assumed to be driven by a Brownian motion with constant drift (equivalently velocity). Monnier *et al.* (2015) propose a Bayesian hidden Markov model. In their model, the particle switches between Brownian motion and Brownian motion with a constant drift. Türkcan and Masson (2013) propose an adaptation of their Bayesian decision tree method in order to discriminate Brownian motion and a parametric confined motion by selecting a model which minimizes the BIC criterion; they use a sliding window approach, computing the BIC on subtrajectories. Relying on parametric models allow to have optimal results if the data fit the model. On the contrary a wrong model is likely to lead to wrong conclusions. Among these procedures, Helmuth *et al.* (2007), Dosset *et al.* (2016) and Vega *et al.* (2018) consider the three modes of diffusions to segment the trajectory.

3 Approach

In this section, we first expose the change-point model. Then, we consider our problem as a statistical test. Finally, we present briefly the classification procedure of Briane *et al.* (2018) that is used in our procedure.

3.1 Change-point model

We observe the successive positions of a single particle in a d-dimensional space ($d=2$ or $d=3$) at time $t_0, t_1, \ldots, t_{n-1}$. We suppose that the lag time between two consecutive observations is a constant Δ. The observed trajectory of the particle is

$$\mathbb{X}_n = (X_{t_0}, X_{t_1}, \ldots, X_{t_{n-1}}), \tag{1}$$

where $X_{t_k} \in \mathbb{R}^d$ is the position of the particle at time $t_k = t_0 + k\Delta$, $k = 0, \ldots, n-1$. In the sequel, we propose a simple change-point model. More complex models related to the models studied in Briane *et al.* (2018) are given in Supplementary Section S1. Even continuous time random walk can be considered (see Briane *et al.*, 2018). Here we assume that the discrete trajectory is generated by a d-dimensional ($d=2$ or $d=3$) diffusion process (X_t) that is the strong solution of the stochastic differential equation (SDE):

$$dX_t = \mu(X_t, t)dt + \sigma dB_t, \quad t \in [t_0, t_{n-1}], \tag{2}$$

where B_t denotes a d-dimensional Brownian motion; the unknown parameters of the model are the diffusion coefficient $\sigma > 0$ and the drift term $\mu : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}^d$.

Furthermore, we assume that there exists a sequence of N change-points on $[t_0, t_{n-1}]$, namely $t_0 = \tau_0 < \tau_1 < \ldots \tau_N < \tau_{N+1} = t_{n-1}$ such that

$$\mu(x, t) = \mu_j(x) \text{ for } t \in [\tau_j, \tau_{j+1}), \tag{3}$$

The vector of change-points $(\tau_j)_{j=1\ldots N}$ and the number of change-points N are unknown. We also assume that for each τ_j there exists $0 \leq j^* \leq n$ such that $\tau_j = t_{j^*}$. In other words, the change of motion occurs precisely at a sampling time. The parameters (μ_j, σ) are also unknown. Moreover, we assume that μ_j and μ_{j+1} are associated with different types of diffusion. Let us give two examples. First, suppose the particle evolves freely in the cytosol during $[\tau_j, \tau_{j+1})$ and then is transported actively by molecular motors during $[\tau_{j+1}, \tau_{j+2})$. This situation can be modelled by a switch between Brownian motion and Brownian motion with drift (an example of superdiffusion). Then, in terms of drift, we will have $\mu_j(x) = 0_d$ (Brownian motion) and $\mu_{j+1}(x) = \mathbf{v}$ (Brownian with drift) with $\mathbf{v} \in \mathbb{R}^d$ the velocity of the molecular motor. Secondly, we can imagine that the particle evolves freely in the cytosol during $[\tau_j, \tau_{j+1})$ and then is confined in a domain during $[\tau_{j+1}, \tau_{j+2})$. This situation can be modelled by a switch between Brownian motion and the Ornstein–Uhlenbeck process (an example of subdiffusion). In this case, we have $\mu_{j+1}(x) = -\lambda(x - \theta)$ where $\lambda > 0$ models the restoring force toward the centre of the confinement domain θ. These types of switches are studied in Section 5. We note that the aforementioned switches are just examples and that the method presented in the sequel can handle other types of switches.

In what follows, we present a sequential procedure to estimate both the number of change-points N and the vector of change-points $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_N)$. In the simple motion model (2), we emphasize that our algorithm do not rely on any parametric assumptions about the drift parameter μ. Then our algorithm allows to deal with a wide variety of motions and defines a non-parametric method. Moreover, we previously state that the algorithm can deal with an even larger set of models. In the next section, we present the sequential procedure as a statistical test.

3.2 Test hypotheses for change-point problem

We follow the idea of the multiple testing procedure of Cao and Wu (2015) which is relevant for our problem. In our settings, the global null hypothesis is that (X_t) is Brownian on $[t_0, t_{n-1}]$:

$$H_0 : \mathbb{X}_n \text{ is generated from } (\sigma B_t)_{t_0 \leq t \leq t_{n-1}}. \tag{4}$$

The global alternative hypothesis is that there exist N change-points $\boldsymbol{\tau}$ such that sub-trajectories on $[\tau_j, \tau_{j+1})$ and $[\tau_{j+1}, \tau_{j+2})$ are coming from different types of diffusion (Brownian, subdiffusion or superdiffusion) defined by their drift μ_j and μ_{j+1}.

Remark 3.1. *The case where the whole trajectory is subdiffusive or superdiffusive belongs to the alternative hypothesis. In this case there is no change-point* ($\boldsymbol{\tau} = \varnothing$).

Cao and Wu (2015) observe the outcome of K statistical hypotheses tests H_{0j} versus H_{1j}, $j = 1 \ldots K$. The hypothesis H_{0j} corresponds to the observation of noise at location j while H_{1j} matches with the observation of a true signal. The authors assume that there exist clusters of noise H_{0j} and clusters of true signal H_{1j}. The objective is to detect these clusters. For each location j, they compute two local averages of the P-values associated with the tests before and after location j. If these two local averages are not in the same range of values, j is a potential change-point.

In our context, H_{0j} states that the subtrajectory of chosen size k on $[t_j, t_{j+k}]$ is a Brownian motion. For each time t_j, we adapt the idea of Cao and Wu (2015) by considering two test statistics computed on the subtrajectories of size k $(X_{t_{j-k}}, X_{t_{j-k-1}}, \ldots, X_{t_j})$ (before t_j) and $(X_{t_j}, X_{t_{j+1}}, \ldots, X_{t_{j+k}})$ (after t_j). These test statistics are initially developed in Briane *et al.* (2018) and allow to classify the subtrajectories into three groups of diffusion namely Brownian motion, subdiffusion and superdiffusion. Then, the heuristic is to carry multiple testing on these subtrajectories and define clusters of subtrajectories driven by the same type of diffusion. This way, we obtain a procedure, which controls the type I error of the global null hypothesis (4) at level α. In other words, the procedure will not detect falsely a change-point when the trajectory is fully Brownian with probability $1 - \alpha$.

3.3 Trajectory classification with a three-decision test

If there is no change-point ($\boldsymbol{\tau} = \varnothing$), Briane *et al.* (2018) propose a three-decision test procedure in order to decide if the trajectory $\mathbb{X}_n$ is subdiffusion, superdiffusion or Brownian motion. The null hypothesis

of their test is H_0 ($\mathbb{X}_n$ is Brownian). The two alternative hypotheses are H_1 ($\mathbb{X}_n$ is subdiffusive) and H_2 ($\mathbb{X}_n$ is superdiffusive). Briane *et al.* (2018) use the following test statistic to carry their statistical test:

$$T_n = \frac{\max_{i=1,\dots,n-1}\|X_{t_i} - X_{t_0}\|_2}{(t_{n-1} - t_0)^{1/2}\hat{\sigma}_n(t_0 : t_{n-1})}, \tag{5}$$

where

$$\hat{\sigma}_n^2(t_0 : t_{n-1}) = \left((1/(2(n-1)\Delta))\sum_{j=1}^{n-1}||X_{t_j} - X_{t_{j-1}}||_2^2\right)^{1/2} \tag{6}$$

is a consistent estimate of the diffusion coefficient σ under the null hypothesis. The statistic of the maximum is scaled to have a standardized measure. Consequently, under the hypothesis that $\mathbb{X}_n$ comes from a Brownian motion of diffusion coefficient σ, the distribution of T_n does not depend on σ nor Δ but only on the trajectory size n. Then we can define $q_n(\alpha)$ the quantile of order α of the distribution of T_n under the null hypothesis (which depends only on n).

Briane *et al.* (2018) interpret T_n as follows. If T_n is low, it indicates that the process stays close to its initial position during the period $[t_0, t_{n-1}]$ then it is likely that it is a subdiffusion. On contrary if T_n is large, the process goes far away from its starting point, as a superdiffusion does with high probability. Consequently, Briane *et al.* (2018) declare that the trajectory $\mathbb{X}_n$ is subdiffusive (H_1) if $T_n < q_n(\alpha/2)$, superdiffusive (H_2) if $T_n > q_n(1-\alpha/2)$ and Brownian (H_0) otherwise. The type I error of this three-decision test is controlled at level α, and we can approximate the quantiles $q_n(\alpha)$ by Monte Carlo estimates.

Briane *et al.* (2018) proved the consistency of the test under different parametric models of subdiffusion and superdiffusion, including models, which are not solution of the SDE (2). Then our algorithm, which relies on (5), can in principle be used even if the motion is not governed by the SDE (2) (see Supplementary Section S1).

4 Materials and methods

In this section, we present our procedure for detecting change-points along a trajectory. We first present the different steps of the main procedure. Then, we focus on the tuning of the parameters of the algorithm. Then we determine the parameters of the algorithm such that the algorithm fulfil theoretical constraints (control of the type I error) (see Section 4.2). In particular, the procedure is sensitive to the size k of the local window. The choice of k may be avoided by aggregating the detections with different window sizes (see Section 4.3).

4.1 A sequential procedure for the change-point estimation

Our procedure comprises four main steps: detecting the potential change-points; gathering these potential change-points into clusters; estimating the change-point in each cluster assuming a cluster contains a single change-point; and discard the inconsistent detections.

Step 1: Detecting the candidate change-points. Let $1 \le k \le n/2$ be fixed. For all index $i \in \{k, k+1, \dots n-k\}$, we consider two subtrajectories of size k starting at X_{t_i}: the backward trajectory $\mathbb{X}_i^- = (X_{t_i}, X_{t_{i-1}}, \dots X_{t_{i-k}})$ and the forward trajectory $\mathbb{X}_i^+ = (X_{t_i}, X_{t_{i+1}}, \dots X_{t_{i+k}})$. Then we carry the test H_{0i} ($\mathbb{X}_i^-$ is Brownian) versus H_{1i} ($\mathbb{X}_i^-$ is subdiffusive) or H_{2i} ($\mathbb{X}_i^-$ is superdiffusive). Symmetrically, we carry the test for $\mathbb{X}_i^+$. If the outcome of the test is similar for the backward trajectory $\mathbb{X}_i^-$ and the forward trajectory $\mathbb{X}_i^+$ then t_i is unlikely to be a change-point; hence t_i is not declared as a candidate change-point. Then we carried $2(n-2k+1)$ tests since we test two sets of hypotheses at each time t_i. In this context of multiple testing, we cannot use the thresholds $(q_n(\alpha/2), q_n(1-\alpha/2))$ for defining the test as in Briane *et al.* (2018) (see also Section 2.3); in fact such a choice does not control the type I error rate of the global null hypothesis (4) at level α. Instead, we define two thresholds $\gamma_1 < \gamma_2$ to manage the multiple testing issues and to control the type I error rate of the null hypothesis (4) at level α (see Supplementary Section S2). An alternative multiple test procedure based on the control of the false discovery rate developed by Benjamini and Hochberg (1995) is discussed in Supplementary Section S4.

Now we explain in a pragmatic way how to proceed. We compute the test statistic (5) for the backward and forward trajectory as,

$$B_i = \frac{\max_{j=1,\dots,k}\|X_{t_{i-j}} - X_{t_i}\|}{(t_i - t_{i-k})^{1/2}\hat{\sigma}(t_i : t_{i-k})}, \quad A_i = \frac{\max_{j=1,\dots,k}\|X_{t_{ij}} - X_{t_i}\|}{(t_{ik} - t_i)^{1/2}\hat{\sigma}(t_i : t_{i+k})}, \tag{7}$$

where $\hat{\sigma}(t_i : t_{i+k})$ (respectively, $\hat{\sigma}(t_i : t_{i-k})$) denotes the estimate of the diffusion coefficient from the forward trajectory $\mathbb{X}_i^+$ (respectively, the backward trajectory $\mathbb{X}_i^-$). The denomination B_i (respectively, A_i) is for '*Before time* t_i' (respectively, '*After time* t_i'). Then, we compare the values of A_i and B_i through the statistic Q_i,

$$Q_i = \begin{cases} 0 & \text{if } (A_i, B_i) \in [0, \gamma_1)^2, \\ 0 & \text{if } (A_i, B_i) \in (\gamma_2, +\infty)^2, \\ 0 & \text{if } (A_i, B_i) \in [\gamma_1, \gamma_2]^2, \\ 1 & \text{otherwise.} \end{cases} \tag{8}$$

If $Q_i = 0$ it means that the statistics B_i and A_i belong to the same range of values defined by γ_1 and γ_2. Then, both $\mathbb{X}_i^+$ and $\mathbb{X}_i^-$ are expected to be from the same type of diffusion: it is unlikely that t_i is a change-point. On the contrary, if $Q_i = 1$ the subtrajectories $\mathbb{X}_i^+$ and $\mathbb{X}_i^-$ are not from the same type of diffusion and t_i is a potential change-point. The detection step is illustrated on a simulated trajectory in Figure 1.

Step 2: Gathering the candidate change-point into clusters. Originally Cao and Wu (2015) propose to define a cluster as $r = k/2$ successive indexes i such that $Q_i \neq 0$. Due to the high level of randomness of the stochastic processes modelling the trajectory, we observe rarely such clusters of size $k/2$ in our case. With this settings, the procedure would fail to detect change-points. A first investigation was to optimize the minimal size r. From an experimental point of view, such optimization would turn out tricky and can lead to overdetection or underdetection depending on the situation. Therefore we propose another way to build clusters. Even if it is hard to observe successive potential change-points, we argue that a subset of indexes where the concentration of potential change-points is high (even if there are not connected) is likely to contain a true change-point. Then, we define a cluster of potential change-points as a subset of successive indexes $\mathcal{M} = \{i, \dots, i+l\}$ such that, for all subset of size c of successive indexes of $\mathcal{M}$, the proportion of candidate change-points exceeds a proportion p,

$$\sum_{j=m}^{m+c-1} Q_j \ge pc \quad \forall m = i, \dots, i+l-c+1. \tag{9}$$

In particular, a cluster needs to have a minimal size of c and if $p < 1$ some points of the clusters may not be candidate change-points (that is $Q_i = 0$). Cao and Wu (2015) define a cluster as connected components of candidate change-points whose length is larger than $k/2$. Moreover it is logically necessary that the proportion of candidate change-points is larger than proportion of the other points, which is lead us to choose $p > 1/2$. Note that $p = 1$ is equivalent to build clusters as presented in Cao and Wu (2015). As illustrated in Figure 2, the choice $p = 1$ fails to include the true change-point (the

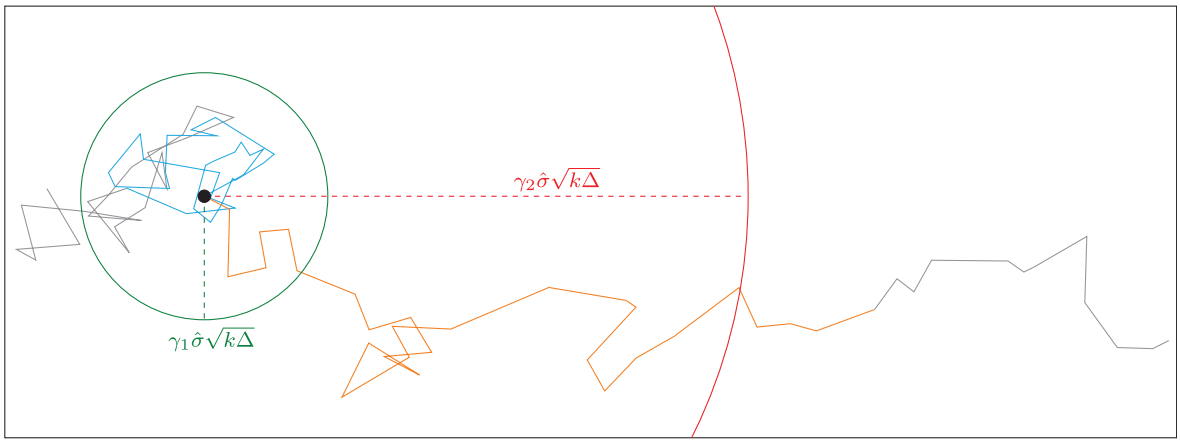

Fig. 1. Illustration of the detection step of the candidate change-points with $k=30$ and $n=90$. The 2D trajectory is Brownian for $i \in \{0,\dots 60\}$, after the process is a Brownian motion with drift. The change-point $\tau_1 = 60$ is represented by the black dot. The blue (respectively, orange) part corresponds to the subtrajectory $\mathbb{X}^-_{\tau_1}$, respectively, $\mathbb{X}^+_{\tau_1}$. Comparing the test statistics (7) to (γ_1, γ_2) is equivalent to determine which area, delimited by the red and green circles centred on X_{τ_1}, the particle reaches. The blue subtrajectory stays inside the inner circle: it is classified as subdiffusive. The orange subtrajectory goes outside the outer circle: it is classified as superdiffusive. This point is detected as a potential change-point

0 0 0 0 0 0 1 0 0 0 1 1 0 0 **0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1** [**1**] **1 0 0 0** 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0.

Fig. 2. Illustration of the cluster using the sequence of Q_i computed on the trajectory presented in Figure 1 (left) (Supplementary Material) with $k = 30$, $c = k/2$ and $p=0.75$. More specifically, the sequence of Q_i correspond to the portion of the trajectory of index $i \in \{140, 200\}$. Hence the detected cluster matches with the second true change-point $\tau_2 = 175$. The bold numbers define a cluster. The true change-point (the boxed number) is contained in this cluster

boxed number) into a cluster. In our context, $c = k/2$ and $p=0.75$ are used as there are optimal from simulation results (see Section 4.2).

Step 3: Estimating the change-point in each cluster. Let $\mathcal{M}_1, \dots \mathcal{M}_{\hat{N}}$ be the clusters defined at the last step. We estimate the change-point of cluster $\mathcal{M}_j$ by the index i for which the difference between Bi and Ai is the highest,

$$\hat{\tau}_j = t_{r_j}, \quad r_j = \max_{i \in \mathcal{M}_j} |B_i - A_i|. \tag{10}$$

Step 4: Discarding inconsistent change-points. Once the change-points are estimated, we classify each subtrajectory $(X_{\hat{\tau}_i}, \dots, X_{\hat{\tau}_{i+1}})$ delimited by the estimated change-points $\hat{\tau}_i$ as a subdiffusion, superdiffusion or Brownian motion using the three-decision test of Briane *et al.* (2018) (see Section 3.2). If two successive subtrajectories $(X_{\hat{\tau}_i}, \dots, X_{\hat{\tau}_{i+1}})$ and $(X_{\hat{\tau}_{i+1}}, \dots, X_{\hat{\tau}_{i+2}})$ belong to the same class, the middle detected change-point $\hat{\tau}_{i+1}$ is inconsistent with the subtrajectory classification. Then we discard $\hat{\tau}_{i+1}$ from the set of detected change-points.

Finally, the method can be summarized as Procedure 1.

Procedure 1 Let $\mathbb{X}_n$ be the observed trajectory of size n. The algorithm to detect change-points is:

1(a). *For a chosen window size k compute B_i and A_i in (7) for $i = k, \dots, n-k$.*

1(b). *For prespecified cut-off values $\gamma_1 < \gamma_2$ compute Q_i from (8), and decompose $\{k, \dots, n-k\} = W_0 \cup W_1$ where $i \in W_0$ if Q_i=0 and $i \in W_1$ if Q_i=1.*

2.*Gather the potential change-points, that is points t_i such that Q_i=1, into clusters $\mathcal{M}_1, \dots, \mathcal{M}_{\hat{N}}$ satisfying Equation (9).*

3. *For each $\mathcal{M}_j$ let $r_j = \max_{i \in \mathcal{M}_j} |B_i - A_i|$ then $\hat{\tau}_j = t_{r_j}$.*

4. Classify *the subtrajectories $(X_{\hat{\tau}_i}, \dots, X_{\hat{\tau}_{i+1}})$ with the three-decision test of (Briane et al., 2018) and discard the inconsistent change-points.*

Remark 4.1. As mentioned in Section 3.2, the distribution of the test statistics (7) does not depend on Δ nor σ under the global null hypothesis (the observed trajectory is a Brownian motion), see also Appendix of Briane *et al.* (2018). It is not the case under the superdiffusive and subdiffusive regimes. For instance, under the Brownian with drift (14), Briane *et al.* (2018) show that the distribution of test statistic (5) depends on $\nu\sqrt{\Delta}/\sigma$. One can also find how the distribution of the test statistic depends on the parameters of an Ornstein–Uhlenbeck process but it is out of the scope of this paper.

4.2 Parameters of the procedure

The parameters of Procedure 1 are the size of the window k, the parameters defining the clusters c and p and finally the cut-off values (γ_1, γ_2). We carried a sensitivity analysis on parameters (c, p) on the simulation scheme presented in Supplementary Section S2. The results are given in Supplementary Table S3. From this analysis, the optimal choice is $c = k/2$ and $p=0.75$. The cut-off values (γ_1, γ_2) are automatically computed and depend on the other parameters of the procedure that is (k, c, p), the trajectory size n and the dimension d. More specifically, as already mentioned in Section 4.1, (γ_1, γ_2) are chosen in order to control the type I error under the global null hypothesis (4) at level $\alpha \in (0, 1)$. In other words, we choose (γ_1, γ_2) such that, when the trajectory is fully Brownian (without any change-point), we have probability α to detect falsely a change-point. We note that we use the standard value $\alpha = 0.05$ and do not consider α as a parameter of our procedure. Values of (γ_1, γ_2) for different trajectory sizes n and dimensions $d=2$, 3 are given in Table 1. There are computed with the Monte Carlo Algorithm 1 (see Supplementary Material). The way to choose the cut-off values (γ_1, γ_2) is carefully explained and mathematically justified in Supplementary Section S2. We propose now a method for aggregating the detections obtained with different window sizes k.

4.3 Window aggregation

For a given trajectory, we cannot know *a priori* which window size k is optimal. Moreover, it is also possible that there does not exist an optimal window size for detecting all the change-points along a trajectory due to the variability of spacing and dynamics between change-points. Actually, a large window is able to detect a small

Table 1. Cut-off values (γ_1, γ_2) of Procedure 1 for different trajectory sizes n and window sizes k for dimensions $d = 2, 3$

		$d=2$		$d=3$	
n	k	γ_1	γ_2	γ_1	γ_2
150	20	0.74	3.12	0.96	3.46
150	30	0.79	3.09	1.01	3.37
150	40	0.81	3.05	1.03	3.35
300	20	0.71	3.29	0.91	3.60
300	30	0.74	3.28	0.95	3.59
300	40	0.75	3.27	0.96	3.59

Note: The cut-off values are estimated with the Monte Carlo Algorithm 1 (Supplementary Material) using $V = 10\,001$ replications and the default parameters of Procedure 1 $c = k/2$, $p = 0.75$ and with $\alpha = 0.05$.

deviation from Brownian motion (either subdiffusion or superdiffusion) when the change-points are well separated while a small window can detect large deviation from Brownian motion when the change-points are close together. Then by combining different window sizes we may ether get close to the performances of the optimal window size if it exists or even outperform the results of all the window sizes in more complex situations where there is no optimal window size. To this end, we propose a simple way for aggregating the change-points detected with an arbitrary number q of window sizes $k_1, \ldots, k_q$. Let $\hat{\tau}_1 \leq \hat{\tau}_2 \leq, \ldots, \leq \hat{\tau}_m$ be the aggregated change-points detected with all the q window sizes and sorted in increasing order. We note the vector of aggregated change-points $\hat{\boldsymbol{\tau}}$. We define a cluster of detected change-points $(\hat{\tau}_i, \ldots, \hat{\tau}_{i+r})$ if:

$$|\hat{\tau}_j - \hat{\tau}_{j+1}| < n_{\min} \quad j = i, \ldots, i+r-1. \tag{11}$$

Then the change-points $(\hat{\tau}_i, \ldots, \hat{\tau}_{i+r})$ are removed from $\hat{\boldsymbol{\tau}}$ and replaced by the mean change-point of the cluster:

$$\hat{\tau} = \frac{1}{r+1} \sum_{j=1}^{r+1} \hat{\tau}_j \tag{12}$$

We keep the isolated detected change-points in $\hat{\boldsymbol{\tau}}$. We end up with a vector of aggregated change-points of size $m' \leq m$ noted $\hat{\boldsymbol{\tau}} = (\hat{\tau}'_1, \hat{\tau}'_2, \ldots, \hat{\tau}'_{m'})$ Then we use the test of Briane *et al.* (2018) to classify each subtrajectory $(X_{\hat{\tau}'_i}, \ldots, X_{\hat{\tau}'_{i+1}})$ as a subdiffusion, a superdiffusion or a Brownian motion. If two successive subtrajectories $(X_{\hat{\tau}'_i}, \ldots, X_{\hat{\tau}'_{i+1}})$ and $(X_{\hat{\tau}'_{i+1}}, \ldots, X_{\hat{\tau}'_{i+2}})$ belong to the same class, we discard the middle detected change-point $\hat{\tau}'_{i+1}$ from the vector of aggregated change-points $\hat{\boldsymbol{\tau}}$. The parameter $n_{\min}$ must be large enough to gather into the same cluster the change-points matching to a single common true change-point. On the other hand, the parameter $n_{\min}$ must be small enough not to connect detected change-points corresponding to different true change-points. Then a reasonable range of values for $n_{\min}$ is $[5, 10]$, value close to 5 should be preferred in the case where the trajectory is short.

REMARK 4.2. In practice, we can use the default set of sizes $(k_j)_{j=1,\ldots,q} = \{10, 20, 30, \ldots, n/2\}$. When the window size becomes too large (extreme case $k_q = n/2$) Procedure 1 is very unlikely to find any detection. Then the aggregated Procedure 1 converges to a certain set of detections with a limited number of window sizes. One can see this convergence in the analysis of the β-actin mRNP trajectory (Section 6.5).

4.4 Inference of the diffusion parameters

Once we run Procedure 1 or even aggregate the different detections obtained with different windows, we obtain both the locations of the change-points and the classification of the corresponding subtrajectories into the three groups of diffusion. Then the user can consider parametric models in order to characterize quantitatively the motion of each subtrajectory as a post-processing step.

First, we may estimate the diffusion coefficient of the subtrajectories classified as Brownian with the estimator (6). This estimator is in fact the maximum likelihood estimator of the diffusion coefficient in the Brownian case. Secondly, we can choose to fit a parametric model to the subdiffusive and superdiffusive subtrajectories. For example, we can consider the Brownian with drift or the fractional Brownian motion with Hurst index $\mathfrak{h} > 0.5$ for superdiffusion. On the other hand, we can choose the Ornstein–Uhlenbeck process or the fractional Brownian motion with Hurst index $\mathfrak{h} < 0.5$ for subdiffusion. One can use model selection to choose the best parametric model to fit superdiffusion/subdiffusion. Once the parametric model is chosen, we can fit the parameters of the model using standard estimation techniques such as maximum likelihood estimation or estimation based on the moments of the model.

We emphasize that one advantage of our method is that we do not *a priori* constraint the subdiffusive and superdiffusive subtrajectories to fit to a parametric model in order to detect the points of change. This is due to the non-parametric nature of the statistical test procedure. On the other hand the non-parametric nature of the statistical test procedure is not consistent to detect switches of parameters over the time inside a subtrajectory of a given type of diffusion (see Remark 6.1 for more details). Indeed, mis-specified parametric models might to have a bad influence on the change-points estimation. The whole algorithm is summarized in Figure 3.

5 Results

In this section, we first conduct a Monte Carlo study of the procedure in order to evaluate and compare the performance of the method according to four scenarios (see Tables 2 and 3) in the 2D case. Then, we analyse 2D data depicting long range transport of mRNAs (Monnier *et al.*, 2015). In Supplementary Section S2, we compare our method to two parametric competing procedures proposed, respectively, by Türkcan and Masson (2013) and Monnier *et al.* (2015).

5.1 Models of subdiffusion and superdiffusion

First, we expose the parametric models illustrating subdiffusion and superdiffusion that we use in our simulation. Subdiffusion is modelled by the Ornstein–Uhlenbeck process:

$$\mathrm{d}X_t^i = -\lambda(X_t^i - \theta_i)dt + \sigma dB_t^i, \quad i = 1, 2, \tag{13}$$

where $\lambda > 0$ models the restoring force toward the equilibrium point $\theta = (\theta_1, \theta_2)$; $\sigma > 0$ is the diffusion coefficient. For modelling superdiffusion, we use the Brownian motion with drift solution of the SDE:

$$\mathrm{d}X_t^i = (\nu/\sqrt{2})dt + \sigma dB_t^i, \quad i = 1, 2, \tag{14}$$

where $\sigma > 0$ is the diffusion coefficient and $\nu > 0$. Then the constant drift $\mathbf{v} = (\nu, \nu)/\sqrt{2}$ verifies $||\mathbf{v}|| = \nu$.

In the sequel, we study two different simulation schemes. For each scheme, we simulate two scenarios: one involving subdiffusion and Brownian motion, one involving superdiffusion and Brownian motion.

5.2 A first simulation scheme

We simulate trajectories of size $n = 300$ with two change-points occurring at $\tau_1 = 100$ and $\tau_2 = 175$ (see examples of trajectories in Supplementary Fig. S1). We study two different scenarios (see Table 2).

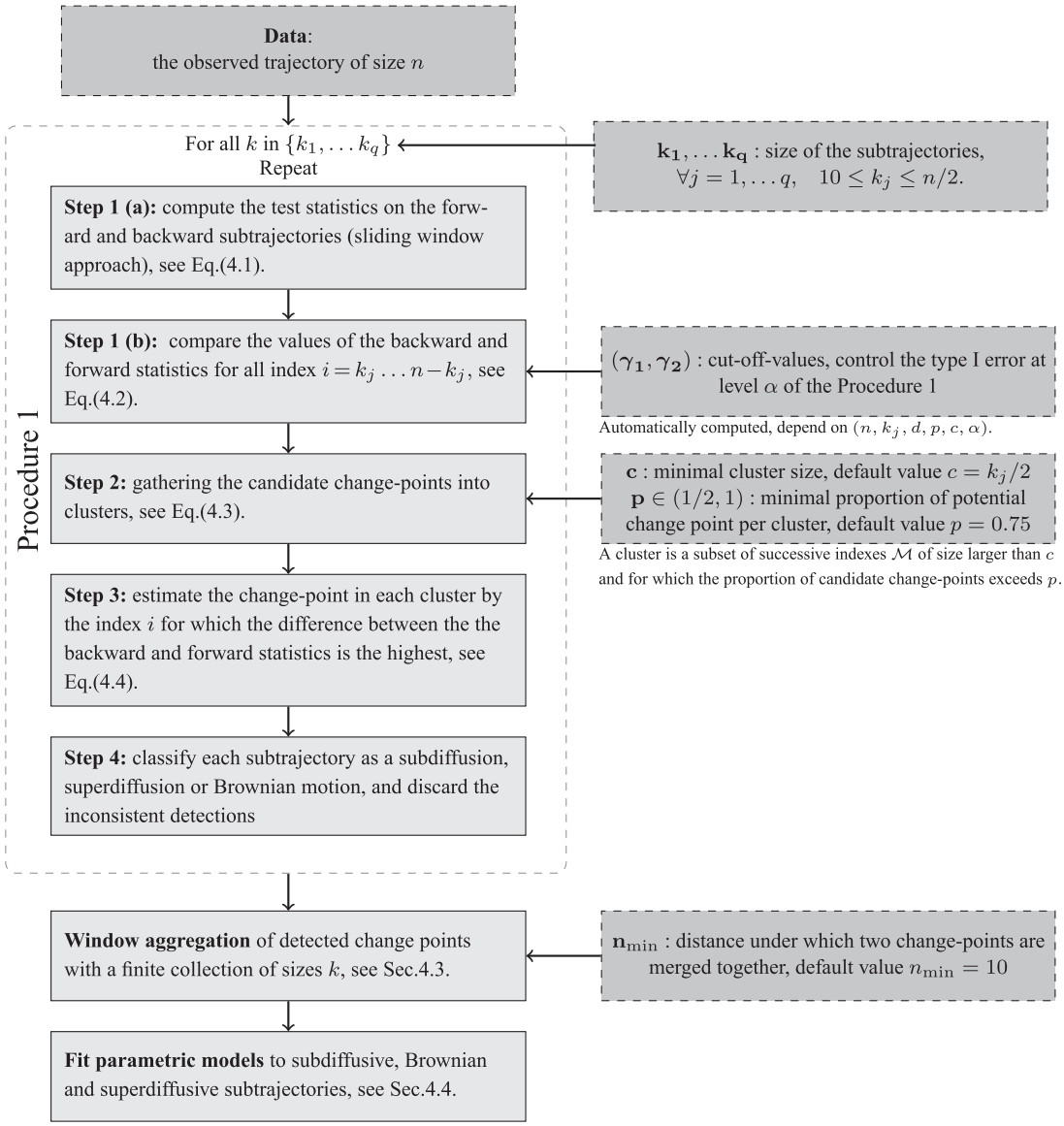

Fig. 3. Flowchart combining Procedure 1 and the aggregation step with their corresponding parameters on the right column

Table 2. Simulation scenarios for the Monte Carlo study

Times	Scenario 1	Scenario 2
[1, 100]	Brownian	Brownian
[101, 175]	Brownian with drift	Ornstein–Uhlenbeck
[176, 300]	Brownian	Brownian

Table 3. Simulation scenarios for the Monte Carlo study with a variety of motions

Times	Scenario 3	Scenario 4
[1, 100]	Brownian	Brownian
[101, 175]	Brownian with drift v=0.8	Ornstein–Uhlenbeck with λ=1
[176, 200]	Brownian	Brownian
[201, 300]	Brownian with drift v=10	Ornstein–Uhlenbeck with λ=10

We set $\sigma = 1$ for the diffusion coefficient of all the processes and $\Delta = 1$ for the step of time. For the Ornstein–Uhlenbeck process (13), we define the equilibrium point as $\theta = X_{\tau_1}$ where X_{τ_1} is the position of the particle at τ_1. As noted in Remark 4.1, the relationship between (Δ, σ) and the distribution of the tests statistics can be used to extrapolate the results obtained with $\sigma = 1$ and $\Delta = 1$ to other values of σ and Δ under non-Brownian regimes.

For each scenario, we compute the performances of our procedure for different values of the parameters v (for the Brownian motion with drift) and λ (for the Ornstein–Uhlenbeck process). We run Procedure 1 with the different window sizes $k = 20$, 30, 40 for each situation. Then we run the aggregated Procedure 1 with window sizes (20, 30, 40) and $n_{\min} = 10$. We assess the performances of our algorithm with respect to two criteria: the number of change-points detected and the location of these change-points. The change-point location is assessed only on the trajectories for which we detect the right number of change-points that is $N = 2$. We compute the average and standard deviation of the locations. We also carry the parameter inference as a post-processing step as explained in

Table 4. Performance of the Procedure 1 for Scenario 1 (Table 2) for different window sizes k and different values of the drift v

k	v	$\hat{N}-N$					τ_1	τ_2
		-2	-1	0	1	≥ 2		
20	0.6	47.0	18.2	32.2	0.3	2.4	123.3 (18.7)	156.2 (19.5)
30	0.6	26.2	17.7	55.4	0.2	0.5	112.6 (16.6)	164.0 (17.6)
40	0.6	17.2	17.8	64.6	0.3	0.1	108.7 (13.7)	168.5 (14.3)
Agg.	0.6	16.7	4.7	73.4	1.7	3.5	114.0 (17.1)	165.0 (17.0)
20	0.8	8.4	23.5	61.1	1.8	5.2	113.2 (14.6)	163.7 (15.5)
30	0.8	2.5	10.7	85.5	0.3	1.0	107.2 (11.0)	170.7 (12.2)
40	0.8	1.1	7.4	90.5	0.9	0.1	104.4 (9.8)	172.3 (10.1)
Agg.	0.8	1.6	2.9	86.1	2.4	7.0	107.1 (10.8)	169.9 (11.4)
20	1	1.0	12.8	80	0.7	5.5	106.6 (10.5)	171.1 (9.6)
30	1	0.1	5.5	92.6	1.1	0.7	102.9 (5.8)	174.0 (6.3)
40	1	0	3.4	95.6	0.7	0.3	102.8 (7.1)	174.4 (8.0)
Agg.	1	0.0	2.8	88.8	1.7	6.7	103.8 (7.1)	173.5 (7.9)
20	2	0	4.6	93.5	1.0	0.9	101.2 (2.6)	176.4 (4.6)
30	2	0	4.0	94.9	0.8	0.3	101.3 (3.4)	175.9 (2.8)
40	2	0	3.6	95.9	0.3	0.2	101.6 (2.7)	176.1 (4.9)
Agg.	2	0.0	2.6	94.7	1.5	1.2	101.4 (2.7)	176.2 (5.7)

Note: The results of the aggregation (Agg.) is in bold. The computations are based on 1 001 simulated trajectories from Scenario 1. The columns τ_1 and τ_2 gives the empirical average (and the empirical standard deviation in brackets) of the first and second detected change-point on 300 trajectories among which we detect the right number of change-points ($\hat{N}-N=0$).

Section 4.4. We fit the subdiffusive subtrajectories to the Ornstein–Uhlenbeck (13) and the superdiffusive subtrajectories to the Brownian motion with drift (14). As there are the generative models of the simulation, we can compare the estimated parameters to the real parameters of simulation. We analyse the results of the simulation on the different scenarios in the next paragraphs.

Scenario 1

Table 4 gives us the results associated with Scenario 1 (Table 2). We can see clearly that, as $||v||$ increases, the performance of the method increases with respect to both criteria. For a given window size k, we observe that, on the one hand, the proportion of trajectories for which we detect the right number of change-point ($\hat{N}-N=0$) tends to 1 as v increases; on the other hand, given $\hat{N}-N=0$, the bias and the variance of the estimated change-point decrease to 0 as v increases. Furthermore, we remark that the aggregation of detections give better results than all the windows for $v=0.6$ in terms of number of change-points detected. For the other values of v, the aggregation does not do as well as the optimal window (number of detections and location criteria) but is not too far. Then, this simulation illustrates the efficiency of the proposed aggregation strategy.

Table 5 shows the inference of the parameters of both Brownian and Brownian with drift subtrajectories. For the Brownian with drift, both the diffusion coefficient σ and drift v are estimated with the maximum likelihood estimator. In general, the estimates of the parameters are all close to the real values. In the case $v=0.6, 0.8$, the locations of the detected change-points are less accurate (Table 6). Thus the estimated parameters take into account a piece of a Brownian subtrajectory that corrupts the estimates of the drift (and in a smaller extend the diffusion coefficient).

Scenario 2

Table 6 gives us the results associated with Scenario 2 (Table 2). As in Scenario 1, for a window size $k=20$ the performance of the algorithm increases as λ increases. However, it does not behave the same way if the window size is 30 or 40. For $k=30$, the performance increases from $\lambda=1$ to $\lambda=2$ but remains the same for larger values of λ. For the window size $k=40$, the proportion of trajectories with the correct number of detected change-points dramatically drops from 83.6% with $\lambda=1$ to 54.1% for $\lambda=4$. At the same time, the proportion of trajectories with $\hat{N}-N=-1$ increases. It means that when λ becomes too high the algorithm mixes up the two change-points and finds only one. As λ is high (clear subdiffusion), we detect a potential change-point very early in the trajectory: as soon as few points of the forward subtrajectory $\mathbb{X}_i^+$ enter in the subdiffusion regime ($t \geq \tau_1$) we classify it as subdiffusive. For example, if λ is big enough we can suppose that the subtrajectory of size k $\mathbb{X}_i^+ = (X_{t_i}, \ldots X_{\tau_1}, X_{\tau_1+1}, X_{\tau_1+2})$ will be classified as subdiffusive with only three points in the subdiffusive regime. Then, we get a long sequence of potential change-points. But as k is large, the forward subtrajectory has already reached the second change-point τ_2. Consequently, it begins to detect potential change-points corresponding to the second change-point τ_2. As there is a single cluster of potential change-points, the algorithm only detects one change-point instead of the two expected. From our simulations, we observe that the change-point detected is either close to τ_1 or τ_2: it estimated correctly one change-point out of the two real change-points. The idea is that, in a way, a large λ (a very clear subdiffusion) makes the two change-points get closer artificially. Then, a large window cannot separate them. By aggregating the detections of the different

Table 5. Inference of the parameters for Scenario 1 (Table 2)

	Brownian	Brownian with drift	
v	$\hat{\sigma}$	$\hat{\sigma}$	$\hat{v}$
0.6	1.01 (0.09)	0.92 (0.16)	0.77 (0.14)
0.8	1.01 (0.09)	0.95 (0.12)	0.89 (0.12)
1	1.02 (0.10)	0.97 (0.12)	1.04 (0.14)
2	1.02 (0.10)	1.01 (0.12)	1.96 (0.15)

Note: Change-points are estimated from the aggregation of the detections obtained with window sizes (20, 30, 40). Parameters are inferred on the trajectories with the right number of detected change-points and right classification of subtrajectory motion.

Table 6. Performance of the Procedure 1 for Scenario 2 (Table 2) for different window sizes k and different values of parameter λ

k	λ	$\hat{N}-N$					τ_1	τ_2
		-2	-1	0	1	≥ 2		
20	1	58.6	7.0	33.4	0.6	0.4	110.5 (18.5)	168.6 (15.4)
30	1	15.5	5.6	78.2	0.3	0.4	104.8 (8.0)	170.2 (8.5)
40	1	14.9	6.5	77.9	0.6	0.1	105.0 (10.9)	170.4 (11.5)
Agg.	1	2.4	3.7	90.0	2.3	1.6	105.6 (9.6)	169.6 (10.7)
20	2	23.3	6.5	69.2	0.5	0.5	106.3 (10.2)	170.2 (9.4)
30	2	6.9	5.7	86.3	0.5	0.6	107.4 (9.5)	169.1 (8.1)
40	2	21.9	6.9	70.4	0.6	0.2	108.6 (12.5)	169.0 (12.6)
Agg.	2	1.1	3.6	89.9	2.8	2.6	106.9 (9.6)	169.7 (9.6)
20	3	16.7	5.9	76.3	0.3	0.8	106.3 (5.5)	169.3 (7.3)
30	3	6.7	5.6	86.1	0.9	0.7	108.4 (9.7)	167.5 (9.6)
40	3	30.6	8.2	60.1	0.8	0.3	109.4 (13.7)	166.4 (13.5)
Agg.	3	1.3	3.8	89.0	2.1	3.8	108.2 (9.3)	168.5 (7.6)
20	4	12.7	5.9	79.1	1.3	1.0	107.0 (5.6)	169.6 (9.4)
30	4	8.1	6.6	83.7	1.1	0.5	109.8 (9.6)	167.1 (9.3)
40	4	41.9	8.6	49.1	0.3	0.2	112.1 (14.1)	166.1 (13.0)
Agg.	4	2.3	5.0	85.5	3.2	4.0	109.2 (8.4)	169.0 (9.2)

Note: We use the same simulation protocol as for Table 4.

windows, we solve this problem: the aggregated Procedure 1 shows better performances (number of detected points) than all the windows for every λ.

Table 5 shows the inference of the parameters of both Brownian and Ornstein–Uhlenbeck subtrajectories. For Ornstein–Uhlenbeck, the parameter λ is estimated by the method of moments using the moment of order 2 (covariance). The diffusion coefficient σ is estimated with the maximum likelihood estimator in which we plug the estimate of λ. The estimates of the parameters are all close to the real values except for the case $\lambda = 4$. Again, it is due to the fact that the locations of the change-points are not exact and the subtrajectory contains some Brownian motion, corrupting the estimates. When the real subtrajectory is very confined (large λ), the mix of motion in the detected subtrajectory alters even more the estimate. That is why the worst estimate is for $\lambda = 4$.

5.3 Simulation scheme with a variety of motions

We simulate trajectories of size $n = 300$ with three change-points occurring at $(\tau_1, \tau_2, \tau_3) = (100, 175, 200)$. We study two different scenarios (Table 3). We set $\sigma = 1$ for the diffusion coefficient of all the processes and $\Delta = 1$ for the step of time. For the first (respectively, second) Ornstein–Uhlenbeck process (13), we define the equilibrium point as $\theta = X_{\tau_1}$ (respectively, $\theta = X_{\tau_2}$) where X_{τ_1} (respectively, X_{τ_2}) is the position of the particle at τ_1 (respectively, τ_2) (Table 7).

As in the previous scheme, we simulate trajectories alternating between Brownian subtrajectories and superdiffusive (respectively, subdiffusive) subtrajectories. However, in this scheme, there are two superdiffusive (respectively, subdiffusive) subtrajectories with different parameters and different lengths (Table 3). Actually, this simulation aims to demonstrate the benefit of the aggregated Procedure 1. The idea is that a large window is able to detect a small deviation from Brownian motion (either subdiffusion or superdiffusion) when the change-points are well separated while a small window can detect a large deviation from Brownian motion when the change-points are close together. Then in the situation where both cases arise, a single window will not be able to detect the corresponding change-points. We can clearly see that in Scenario 3 (Table 8) where none of the window sizes perform well. On the contrary, the aggregation strategy allows an accurate detection of all the change-points. This fact is less obvious in Scenario 4 (Table 9) but the aggregation still have the best percentage of the right number of detected change-points compared with all the results obtained with a single window size. As previously, we ran the aggregated Procedure 1 with window sizes (20, 30, 40) and $n_{\min} = 10$.

6 Comparisons with competitive methods

In this section, we present the three competitive methods and compare their performances to Procedure 1 on simulations: Türkcan and Masson (2013), Monnier *et al.* (2015) and Vega *et al.* (2018). At the end of the section, we give a particular emphasis on the speed and stability of the different methods.

Table 7. Inference of the parameters for Scenario 2 (Table 2)

	Brownian	Ornstein–Uhlenbeck	
λ	$\hat{\sigma}$	$\hat{\sigma}$	$\hat{\lambda}$
1	0.97 (0.09)	1.12 (0.28)	1.14 (0.36)
2	0.96 (0.1)	1.18 (0.58)	2.24 (1.17)
3	0.96 (0.1)	1.00 (0.44)	2.87 (1.31)
4	0.95 (0.1)	0.77 (0.32)	2.94 (1.35)

Note: Change-points are estimated from the aggregation of the detections obtained with window sizes (20, 30, 40). Parameters are inferred on the trajectories with the right number of detected change-points and right classification of subtrajectory motion.

6.1 The method of Türkcan and Masson (2013)

The parametric method of Türkcan and Masson (2013) detects change-points between two parametric models: the Brownian motion and the Ornstein–Uhlenbeck process [called diffusion in a harmonic potential in Türkcan and Masson (2013)]. Türkcan and Masson (2013) select the model that minimizes the BIC criterion. For detecting change-points, the BIC criterion is computed on a sliding window along the trajectory. When the BIC indicates a switch of model and that the new model is confirmed in the next r steps of times, a change is assumed to occur.

We reproduce the simulation described in Türkcan and Masson (2013). We simulate $V = 100$ trajectories of size $n = 500$. First the trajectory undergoes an Ornstein–Uhlenbeck process and at time $\tau_1 = 250$ it switches to a Brownian motion. The two processes share the same diffusion coefficient $\sigma = 0.4472$. The specific parameter of the Ornstein–Uhlenbeck process (7) is $\lambda = 7.3870$. The step of time is $\Delta = 0.05$. Results of the two methods are given in Table 10. We can see that our method show better results in both the number $\hat{N}$ of detected change-points and in the location of the change-points. We also emphasize that we do not set $r = 3$ as in Türkcan and Masson (2013) but we set $r = 51$ which corresponds to the size of the window. With $r = 3$, the method of Türkcan and Masson (2013) detects more than four change-points in 91% of the trajectories. Actually, the method is able to detect the change-point, if a collection of about $V = 50$ trajectories showing the same number of change-points at the same location is available. Accordingly, it provides good results in average. However, such a situation is not realistic in practical imaging. In our scenarios, our non-parametric method outperforms the parametric method of Türkcan and Masson (2013).

6.2 The method of Monnier *et al.* (2015)

The method of Monnier *et al.* (2015) use hidden Markov models to fit the displacements of the particle over time. The authors consider the parametric model of Brownian motion with drift with two parameters: the drift $v = (v_1, v_2)$ and the diffusion coefficient σ.

Table 8. Performance of the Procedure 1 for Scenario 3 (Table 2)

k	$\hat{N} - N$					τ_1	τ_2	τ_3
	≤ -2	-1	0	1	≥ 2			
20	36.2	8.9	50.0	0.7	4.3	113.6 (16.6)	166.7 (12.5)	200.0 (8.6)
30	40.8	3.4	55.3	0.3	0.2	106.1 (10.8)	167.1 (10.4)	200.3 (6.4)
40	61.1	1.4	37.4	0.1	0	103.7 (8.6)	159.8 (5.2)	200.8 (2.7)
Agg.	9.4	1.3	82.8	0.9	5.6	107.4 (11.1)	165.7 (10.6)	200.6 (5.6)

Table 9. Performance of the Procedure 1 for Scenario 4 (Table 2)

k	$\hat{N} - N$					τ_1	τ_2	τ_3
	≤ -2	-1	0	1	≥ 2			
20	61.6	4.6	33.1	0.6	0.1	108.3 (13.4)	165 (15.7)	207.5 (9.6)
30	18.0	4.1	77.3	0.5	0.1	104.8 (8.1)	171.1 (9.0)	212.7 (8.9)
40	15.5	5.3	78.9	0.3	0.0	105.6 (11.0)	174.9 (9.3)	221.0 (9.9)
Agg.	4.3	2.8	83.4	3.0	6.5	106.2 (10.3)	171.5 (10.3)	212.4 (9.7)

Table 10. Comparison of the aggregated Procedure 1 with window sizes $(10, 20, \ldots, 50)$ and $n_{\min} = 10$ and the method of Türkcan and Masson (2013) on the simulation of Türkcan and Masson (2013)

	$\hat{N} - N$				
Method	-1	0	1	≥ 2	τ_1
Procedure 1	5	88	1	6	227.8 (34.5)
Method of Türkcan and Masson (2013)	27	59	14	0	176.3 (53.7)

Note: We recall that the true change-point is $\tau_1 = 250$.

In our settings, their method detects change-points between Brownian motion with non-null drift which is an example of super-diffusion and the Brownian motion (i.e. a Brownian motion with a null drift). The hidden states are defined as the sequence of drift parameters and diffusion coefficients s_i over the time that takes value in the finite discrete state space $\{(\nu_k, \sigma_k), k = 1 \ldots K\}$. They estimate both the number of states K (a higher bound for K is given a priori), the state space and the successive (hidden) states along the trajectories. They also add a constrained $\nu = (0, 0)$ for modelling Brownian motion. Model selection is used with a Bayesian criterion to select the best model. If we assume that $K \leq 2$, the competing models are:

Model 1 a single state which is the Brownian with parameter σ_1 [see Equation (4)],

Model 2 a single state which is the Brownian with drift with parameters (ν_1,σ_1),

Model 3 two states which are two Brownian with parameter σ_1 and σ_2,

Model 4 two states which are one Brownian and one Brownian with drift with respective parameters σ_1 and (ν_2, σ_2),

Model 5 two states which are two Brownian with drift with parameters (ν_1, σ_1) and (ν_2, σ_2).

Remark 6.1. One of the appeal of Monnier *et al.* (2015) is that there is only one parameter to set: the higher bound for K (typically 2 or 3 otherwise the number of models to compare become computationally intractable). Due to the parametric choice of the emission model of the hidden Markov model, the procedure of Monnier *et al.* (2015) is able to detect also changes in parameter value for a fixed type of motion, see for instance Models 3 and 5. It is not the case for the proposed algorithm due to the non-parametric nature of the statistical test procedure: the algorithm can detect switching between different types of motions, but it is not designed to detect changes in parameter value for a fixed type of motion. Supplementary Table S5 illustrates this fact for changes in the diffusion coefficient for Brownian motion.

In our experiment, we run the method of Monnier *et al.* (2015) on 100 simulated trajectories from Scenario 1. We choose the optimal higher bound for K in this simulation, that is $K \leq 2$, in order to test the right model (Model 4) against a minimum number of competitive models aforementioned. In the framework of Monnier *et al.* (2015), change-points will be detected if the selected model are the Models 3, 4 or 5. In this simulation though the only right model is Model 4. Results are given in Tables 11 and 12. When the drift is too low $(\nu = 0.6, 0.8)$, the procedure of Monnier *et al.* (2015) fails to select the right model comparing to our procedure. As expected, the performance of the method of Monnier *et al.* (2015) improves as ν increases. Even when the right model (Model 4) is chosen, the rate of detection of the true change-points (that is $\hat{N} - N = 0$) is below to our procedure. Nethertheless, the associated standard deviation of the estimation of the location of these true change-points is lower than our procedure for $\nu = 1$. Finally, Monnier *et al.* (2015) outperform our method in terms of rate of detection of the true change-points and the accuracy of the their location estimations when the drift is sufficiently high $(\nu = 2)$ compared with the diffusion coefficient.

Table 11. Performance of the algorithm of Monnier *et al.* (2015) for Scenario 1 for different values of ν based on 100 simulated trajectories

	$\hat{N} - N$						
ν	-2	-1	0	1	≥ 2	τ_1	τ_2
0.6	99	0	1	0	0	93.0 (0.0)	177.0 (0.0)
0.8	82	0	15	1	2	96.0 (7.9)	173.4 (3.9)
1.0	23	0	68	7	2	99.9 (3.9)	174.9 (4.3)
2.0	0	0	96	1	3	100.0 (1.4)	175.0 (1.2)

Note: The columns τ_1 and τ_2 gives the empirical average (and the empirical standard deviation in brackets) of the first and second detected change-points on trajectories which detect the right number of change-points.

Table 12. Selected models with the method of Monnier *et al.* (2015) on 100 simulated trajectories from Scenario 1

ν	Selected model				
	Model 1	Model 2	Model 3	Model 4	Model 5
0.6	97	2	0	1	0
0.8	74	8	0	18	0
1	18	5	0	77	0
2	0	0	0	100	0

6.3 The method of Vega *et al.* (2018)

Vega *et al.* (2018) propose an algorithm that aims to distinguish four kind of motions: immobile, free diffusion (e.g. Brownian motion), confined diffusion (e.g. subdiffusion) and directed diffusion (e.g. superdiffusion). The method consists in three steps. First, they carry an initial track segmentation based on the maximum pairwise distance computed on local windows. Then they classify the segments based on the slope of the Moment Scaling Spectrum (MSS). For $m = 0, \ldots, 6$, the function of time $\mathbb{E}(||X_t - X_0||^m)$ is fitted to the power function $4D_m t_m^{\alpha}$, with D_m the general diffusion coefficient and α_m the scaling power. The MSS is defined as the function of α_m against m. Finally, they use a decision tree to merge some detected segments and produce the final segmentation and classification. There are two main drawbacks to the method. Vega *et al.* (2018) calibrate thresholds on the MSS-slope based on simulated data. The choice of the models and parameters used for calibration affect the thresholds very much. We will see in the sequel that there are not very robust when the data do not come from the models used for calibration. Secondly, theoretically the function $\mathbb{E}(||X_t - X_0||^m)$ do not necessarily fit a power function [see Briane *et al.* (2019) for the case $m = 2$ corresponding to the Mean Square Displacement]. More problematically, the MSS of the models chosen for calibrating the MSS-slopes do not follow a power function [see again Briane *et al.* (2019) for the case $m = 2$].

Figure 4 compare the performances of the aggregated Procedure 1 and the method of Vega *et al.* (2018) on the simulation scheme of Vega *et al.* (2018, in Supplementary Fig. S16c). This simulation scheme is described in Table 13. We compute the percentage of trajectories for which they detect the right number of switches (1) for different localizations of τ_1 ($\tau_1 = 20, 30 \ldots, 80$). The thresholds of

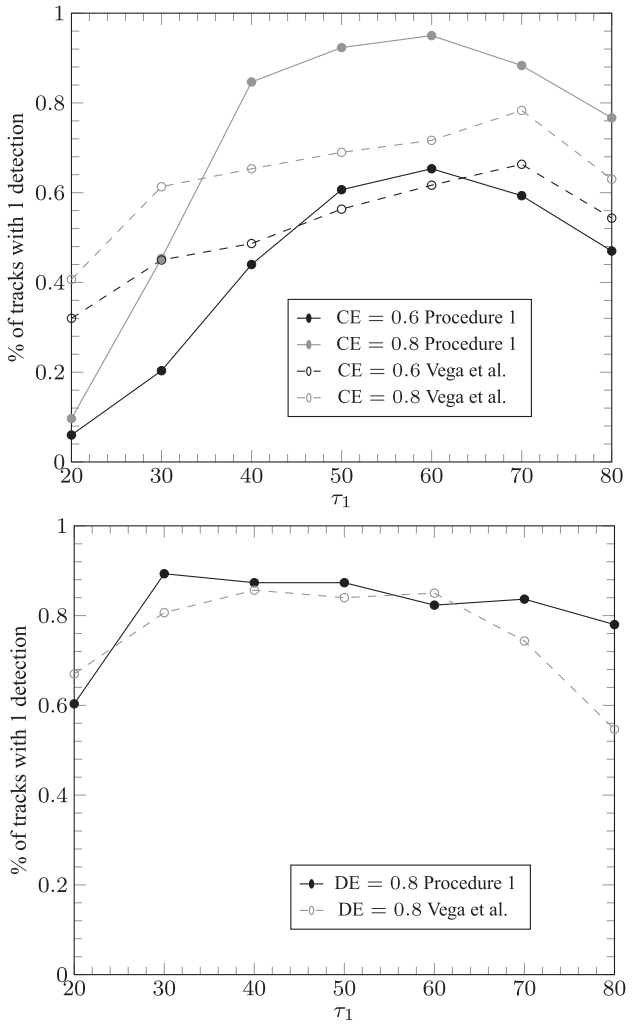

Fig. 4. Comparison of the aggregated Procedure 1 with window sizes $(10, 20, 30, 40)$ and $n_{\min} = 10$ and the method of Vega *et al.* (2018) for Scenario 5 (top) and Scenario 6 (bottom)

Table 13. Simulation scenarios of Vega *et al.* (2018)

Times	Scenario 5	Scenario 6
$[1, \tau_1]$	Brownian confined in a disk of radius R with reflecting boundaries	Brownian with drift
$[\tau_1 + 1, 100]$	Brownian	Brownian

Note: In Scenario 5, the confinement extent $\text{CE} = \sqrt{4D\Delta}/R$ belongs to $\{0.6, 0.8\}$ (with Δ the step of time and $D = \sigma/2$ in our settings). In Scenario 6, the drift extent $\text{DE} = ||\nu||\sqrt{\Delta}/\sqrt{4D}$ is equal to 0.8 corresponding to $\nu \approx 1.13$ in the SDE (14) when $\sigma = 1$ and $\Delta = 1$.

the MSS-slopes were computed from confined diffusion with CE = 1 and from Brownian motion with drift with DE = 0.7 (see Table 13 for definitions). The models tested in the simulation are the same as the ones; Vega *et al.* (2018) used for calibrating the thresholds, the parameters being just slightly different. Then, it is a favourable simulation scheme for Vega *et al.* (2018). Figure 4 (bottom) shows that our method performs almost always better when Brownian alternates with Brownian with drift. For Scenario 5 (at the top of Fig. 4), our method is as good or better compared with Vega *et al.* (2018) for τ_1 ranging from 40 to 70, that is when the change-points is not too close to the boundaries of the observation period. We also assess the method of Vega *et al.* (2018) in Scenarios 1 and 2 in Supplementary Table S2. Our method outperforms (Vega *et al.*, 2018) in all cases suggesting that the choice of the thresholds of MSS-slopes depend very much on the models chosen for calibration.

6.4 Algorithmic considerations

Finally, we compare the speed and stability of the different methods. The method of Monnier *et al.* (2015) is time consuming because of the estimation of the a posteriori distribution by the Metropolis-Hastings algorithm. Assuming $K \leq 2$, it took 115 s in average to deal with one trajectory of the simulation presented in Table 11 (300 points) with four cores working in parallel on a Mac Book Pro version 10.10.1 equipped with 2.8 GHz Intel Core i7, 16 Gb of RAM. In comparison, the aggregated Procedure 1 with window sizes (20, 30, 40) takes 0.12 s to process a trajectory without working in parallel. Both our procedure and the method of Türkcan and Masson (2013) compute quantities on local windows [in our case the statistics (1), the BIC of different models for (Türkcan and Masson, 2013)]. From this aspect, the complexity of these two algorithms is equivalent. However, Türkcan and Masson (2013) needs to estimate the MAP (maximum a posteriori) to compute the BIC. They choose a complex likelihood to model the spatial heterogeneity of the motion. Therefore, they use quasi-Newtonian optimization to find the MAP which is the most time consuming step of their procedure. It took in average 11 s to process a trajectory of the simulation presented in Table 11 (500 points) against 0.22 s for the aggregated Procedure 1 with window sizes $(10, 20, 30, 40, 50)$. In term of stability, different runs of the method of Monnier *et al.* (2015) on the same trajectory can give different results (see Section 4.4). This is due to a bad convergence of the Metropolis-Hastings algorithm. In rare cases, the optimization step of Türkcan and Masson (2013) can fail. Procedure 1 does not suffer any of these problems as it does not involve any parameter inference. Finally, the method of Vega *et al.* (2018) took 1.01 s to analyse a trajectory of size $n = 300$ from Scenario 1 of simulation scheme 1 while the aggregated Procedure 1 with window size (20, 30, 40) processed the trajectory in 0.02 s. We note that we can run in parallel the Procedure 1 with different window sizes to obtain the aggregated Procedure 1.

7 Experiments on real data

We demonstrate the interest of our algorithm on two different sets of data in two and three dimensions.

7.1 2D case: long-range transport of mRNAs

We use the same data (http://hmm-bayes.org/about/) as Monnier *et al.* (2015) depicting long-range transport of mRNAs in complex with mRNPs. In live neuronal cultures, endogenous β-actin mRNP particles alternate between Brownian motion and active transport. In case of active transport (superdiffusion), the particle is driven by molecular motors along microtubule tracks in the neuronal dendrites. The microscopic sequence was obtained using mRNA fluorescence labelling techniques. More specifically, in the experiment of Monnier *et al.* (2015), the MS2 bacteriophage capsid protein was tagged with a Green Fluorescence Protein (GFP). As the MS2 bacteriophage capsid protein binds to β-actin mRNP, it allows to track mRNP.

The time resolution of the sequence is $\Delta = 0.1$ s. The space resolution is not given but when the Brownian motion with drift is chosen, Monnier *et al.* (2015) find a drift parameter with order of

magnitude of 1 μm s^{-1}. We set the parameter $K=2$ for the method of Monnier *et al.* (2015) (see Supplementary Section S2). In this case, Model 3 (two Brownian motion with different diffusion coefficients) is selected by the method. Then, from our point of view, there is no change of dynamic. We note that we run 100 times the algorithm and did not get the same outcome each time. It is due to the fact that the inference is based on a Monte Carlo Markov chains (MCMC) algorithm for computing the a posteriori estimates. Consequently, the selected model was not the same every times (92 times Model 3, 7 times Model 4, 1 time Model 5). Then, the MCMC algorithm can show some problems of stability giving some contrary outcomes from one run to another.

Figure 4 plots the results for the aggregated Procedure 1. Three change-points are detected and we detect the three types of motion for the observed trajectory (Fig. 5). Note that Procedure 1 does not detect any change-point for window sizes greater than 15, that is the window sizes (20, 30, 40) are useless (see Remark 4.2). The results for Procedure 1 with the single window size $k=10$, 15 are plotted in Supplementary Figure S2.

7.2 3D case: Gal-3 proteins in HeLa cells

We study the movements of Gal-3 proteins when entering the cell via endocytosis (Lakshminarayan *et al.*, 2014), followed by active transport via endocytic vesicles within the cytosol. Fluorescence images (199 time points, pixel size in x, y 104 nm, distance in z 369 nm) were acquired using a lattice light sheet microscope (LLSM, 3i, Denver, USA) (Chen *et al.*, 2014). Image volumes of Galactosyltransferase-EGFP (Golgi apparatus for structural orientation) and labelled Gal-3 were recorded every 4.55 s, using 20 ms exposure time. Three-dimensional datasets were deskewed to account for the 32.8° angle of the detection objective. Subsequent deconvolution was performed using the Richardson–Lucy algorithm. For the segmentation algorithm, Golgi and cell masks were defined based on GalT-EGFP images and auto fluorescence signal. The cell contour for each time-point and z-plane were calculated using Matlab Image Processing Toolbox. A similar analysis was performed to estimate the Golgi contour (Fig. 6).

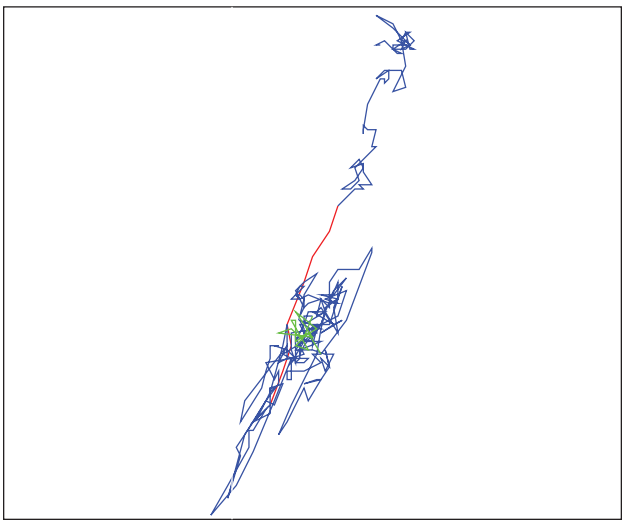

Fig. 5. β-Actin mRNP trajectory analysed with the aggregated Procedure 1 with window sizes $(10, 15, 20, 30, 40)$ and $n_{\min}=5$. The detected change-points are $\tau=(67, 75, 282)$. First, the particle undergoes Brownian motion, then superdiffusion, Brownian motion and finally subdiffusion. The motion type of the subtrajectories is depicted in blue for Brownian, in red for superdiffusion, and in green for subdiffusion

The trajectories were obtained from the sequence of images thanks to the Icy tracker used with the default parameters (Chenouard *et al.*, 2013), available on the Icy software (http:www.icy.org). Then we used a preprocessing step selecting the trajectories with at least 25 distinct positions and that stop at the same position less than $K=\lfloor n/10\rfloor$ times (with n the length of the trajectory). We end up with 408 trajectories with mean length 89. We run the aggregated Procedure 1 on this set of trajectories. We choose $n_{\min}=5$ as some trajectories are quite short. The computational time of our algorithm is 15.50 s (without using parallel computing) on a Mac Book Pro version 10.14.3 equipped with 2.9 GHz Intel Core i7, 16 Gb of RAM. In Figure 6, we represent the segmented trajectories in one HeLa cell.

We observed subdiffusive motion of Gal-3 (Fig. 6, green), which is characteristic for movement of molecules within a confined compartment, like very early endocytic uptake structures or different endosomes. Trafficking between those compartments is often facilitated through molecular motor proteins, characterized by a fast motion, which we identified as superdiffusive trajectories (in red). Another group of Gal-3 tracks is characterized by Brownian motion within the plasma membrane (blue), indicating an early stage of endocytosis, which is characterized by a slow motion of carbohydrate trapped Gal-3. This event is just preceding an endocytic event of internalization from the plasma membrane. The overall picture of trajectories can be described as confined in small regions and Brownian motions of particles (blue and green), which are interconnected by active transport events (red).

8 Discussion

We proposed a non-parametric algorithm to detect the change-points along a particle trajectory. These change-points are defined as the times at which the particle switches between three modes of motion, namely Brownian motion, subdiffusion and superdiffusion. These types of processes are extensively used in the biophysics literature (Bressloff, 2014; Berry and Chaté, 2014). When the trajectory is fully Brownian (our null hypothesis H_0), we control the probability to detect a false change-point at level α. The aggregated version of Procedure 1 allows to combine the detections obtained from a finite collection of window sizes k instead of setting a single window size. This aggregation step allow to combine the benefits of the different window sizes. Also the critical window size k is no longer a parameter to choose thanks to the aggregation step.

We compared our method with the methods of Türkcan and Masson (2013), Monnier *et al.* (2015) and Vega *et al.* (2018). First, we show reliable results on different scenarios due to the non-parametric nature of our procedure. In addition, these results are competitive with existing procedures in the literature that is specific only to a part of the scenarios considered. Secondly, our method is much faster than Türkcan and Masson (2013) and Monnier *et al.* (2015) which is an advantage when dealing with a large numbers of trajectories. We also considered real data depicting neuronal mRNPs (mRNAs in complex with mRNA-binding), and another complex biological example, Gal-3 trafficking from the plasma membrane to different cellular compartments. The analysis of multiple Gal-3 trajectories demonstrates nicely that there is not one typical trajectory signature. Biological trafficking events are very multifaceted. The presented algorithm is capable of identifying and characterizing the multistep biological movement, switching several times between subdiffusive, superdiffusive and Brownian motion.

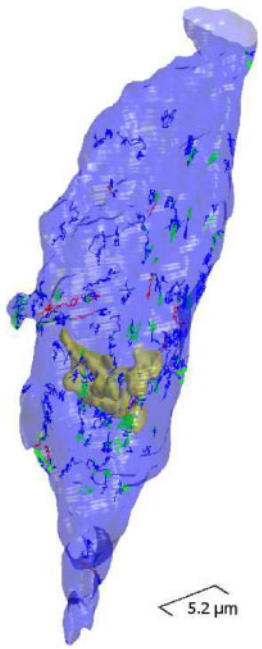

Fig. 6. The 3D Gal-3 proteins trajectories analysed by the aggregated Procedure 1 with window sizes $(10, 15, 20, 30, 40)$ and $n_{\min} = 5$. The motion type of the subtrajectories is depicted in blue for Brownian, in red for superdiffusion, and in green for subdiffusion. In light blue, we plot the cell membrane and in yellow the Golgi apparatus for structural orientation

Acknowledgements

The authors greatly acknowledge Ludovic Leconte from the Cell and Tissue Imaging facility (PICT-IBiSA), Institut Curie, member of the National Research Infrastructure France-BioImaging (ANR-10-INBS-04).

Authors' contributions

V.B.: conceptualization, methodology, modelling, software, formal analysis, investigation, validation, visualization, writing-original draft; M.V.: formal analysis, supervision, investigation, methodology, writing-review and editing. C.A.V.C.: microscopy and image processing; A.S.: image processing and visualization; C.W.: sample preparation and microscopy; C.K.: formal analysis, supervision, investigation, methodology, writing-review and editing.

Funding

Funding was provided by Inria Rennes and CREST-Ensai-Université Bretagne Loire. This work was also supported by the French National Research Agency (France-BioImaging infrastructure–ANR-10-INBS-04, DALLISH–ANR-16-CE23-0005). The lattice light sheet microscope was financed by LabEx DCBiol, LabEx CelTisPhyBio ANR-11-LABX-0038, HFSP (RGP0029/2014).

Conflict of Interest: none declared.

References

Arcizet,D. *et al.* (2008) Temporal analysis of active and passive transport in living cells. *Phys. Rev. Lett.*, **101**, 248103.

Benjamini,Y., and Hochberg,Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc.: Ser. B (Methodol.)*, **57**, 289–300.

Berry,H., and Chaté,H. (2014) Anomalous diffusion due to hindering by mobile obstacles undergoing Brownian motion or Ornstein-Ulhenbeck processes. *Phys. Rev. E*, **89**, 022708.

Bouzigues,C. and Dahan,M. (2007) Transient directed motions of gabaa receptors in growth cones detected by a speed correlation index. *Biophys. J.*, **92**, 654–660.

Bressloff,P.C. (2014) *Stochastic Processes in Cell Biology*. Vol. 41, Springer, Berlin.

Bressloff,P. and Newby,J. (2013) Stochastic models of intracellular transport. *Rev. Modern Phys.*, **85**, 135.

Briane,V. *et al.* (2018) Statistical analysis of particle trajectories in living cells. *Phys. Rev. E*, **97**, 062121.

Briane,V. *et al.* (2019) An overview of diffusion models for intracellular dynamics analysis. *Brief Bioinform.*, bbz052.

Cao,H. and Wu,W.B. (2015) Change-point estimation: another look at multiple testing problems. *Biometrika*, **102**, 974–980.

Chen,B. *et al.* (2014) Lattice light-sheet microscopy: imaging molecules to embryos at high spatiotemporal resolution. *Science*, **346**, 1257998.

Chenouard,N. *et al.* (2013) Multiple hypothesis tracking for cluttered biological image sequences. *IEEE Trans. Patt. Anal. Mach. Intell.*, **35**, 2736–2750.

Chenouard,N. *et al.* (2014) Objective comparison of particle tracking methods. *Nat. Methods*, **11**, 281.

Dosset,P. *et al.* (2016) Automatic detection of diffusion modes within biological membranes using back-propagation neural network. *BMC Bioinformatics*, **17**, 197.

Helmuth,J.A. *et al.* (2007) A novel supervised trajectory segmentation algorithm identifies distinct types of human adenovirus motion in host cells. *J. Struct. Biol.*, **159**, 347–358.

Hoze,N. *et al.* (2012) Heterogeneity of ampa receptor trafficking and molecular interactions revealed by superresolution analysis of live cell imaging. *Proc. Natl. Acad. Sci.*, **109**, 17052–17057.

Kervrann,C. *et al.* (2016) A guided tour of selected image processing and analysis methods for fluorescence and electron microscopy. *IEEE J. Select. Topics Signal Process*, **10**, 6–30.

Lagache,T. *et al.* (2009) Quantitative analysis of virus and plasmid trafficking in cells. *Phys. Rev. E*, **79**, 011921.

Lakshminarayan,R. *et al.* (2014) Galectin-3 drives glycosphingolipid-dependent biogenesis of Clathrin-independent carriers. *Nat. Cell Biol.*, **16**, 595–606.

Maroulas,V. *et al.* (2015) Tracking rapid intracellular movements: a Bayesian random set approach. *Ann. Appl. Stat.*, **9**, 926–949.

Meilhac,N. *et al.* (2006) Detection of confinement and jumps in single-molecule membrane trajectories. *Phys. Rev. E*, **73**, 011915.

Metzler,R. and Klafter,J. (2000) The random walk's guide to anomalous diffusion: a fractional dynamics approach. *Phys. Rep.*, **339**, 1–77.

Monnier,N. *et al.* (2015) Inferring transient particle transport dynamics in live cells. *Nat. Methods*, **12**, 838–840.

Pankov,R. *et al.* (2005) A rac switch regulates random versus directionally persistent cell migration. *J. Cell Biol.*, **170**, 793–802.

Qian,H. *et al.* (1991) Single particle tracking. analysis of diffusion and flow in two-dimensional systems. *Biophys. J.*, **60**, 910.

Roudot,P. *et al.* (2017) Piecewise-stationary motion modeling and iterative smoothing to track heterogeneous particle motions in dense environments. *IEEE Trans. Image Process*, **26**, 5395–5410.

Simson,R. *et al.* (1995) Detection of temporary lateral confinement of membrane proteins using single-particle tracking analysis. *Biophys. J.*, **69**, 989–993.

Truong,C. *et al.* (2018) A review of change point detection methods. arXiv preprint, arXiv: 1801.00718.

Türkcan,S. and Masson,J.-B. (2013) Bayesian decision tree for the classification of the mode of motion in single-molecule trajectories. *PLoS One*, **8**, e82799.

Vega,A.R. *et al.* (2018) Multistep track segmentation and motion classification for transient mobility analysis. *Biophys. J.*, **114**, 1018–1025.

Yin,S. *et al.* (2018) Detection of velocity and diffusion coefficient change points in single-particle trajectories. *Biophys. J.*, **115**, 217–229.