



HAL
open science

A computational approach for detecting microdomains and confinement domains in cells: a simulation study

Vincent Briane, Antoine Salomon, Myriam Vimond, Charles Kervrann

► To cite this version:

Vincent Briane, Antoine Salomon, Myriam Vimond, Charles Kervrann. A computational approach for detecting microdomains and confinement domains in cells: a simulation study. *Physical Biology*, 2020, 17 (2), pp.025002. 10.1088/1478-3975/ab5e1d . hal-02424860

HAL Id: hal-02424860

<https://inria.hal.science/hal-02424860v1>

Submitted on 28 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A computational approach for detecting microdomains and confinement domains in cells: a simulation study

Vincent Briane^{1,2}, Antoine Salomon^{1,3}, Myriam Vimond², & Charles Kervrann^{1,3,*}

¹ Inria, Centre de Rennes Bretagne Atlantique, Serpico Team, Rennes 35042, France

² CREST (Ensaï, Université Bretagne Loire), Bruz 35170, France

³ Institut Curie, PSL Research University, CNRS UMR 144, UPMC, STED Team, 75005 Paris, France

* Corresponding author: charles.kervrann@inria.fr

Abstract. In this paper, we aim at detecting trapping areas (equivalently microdomains or confinement areas) within cells, corresponding to regions where molecules are trapped and thereby undergo subdiffusion. We propose an original computational approach that takes as input a set of molecule trajectories estimated by appropriate tracking methods. The core of the algorithm is based on a combination of clustering algorithms with trajectory classification procedures able to distinguish subdiffusion, superdiffusion and Brownian motion. The idea is to automatically identify trapping areas where we observe a high concentration of subdiffusive particles. We evaluate our proof of concept on artificial sequences obtained with a biophysics-based simulator (FLUOSIM), and we illustrate its potential on real TIRF microscopy data.

1. Introduction

As the interior of a living cell is a fluctuating environment, the trajectories of intracellular particles can be modeled with stochastic processes having continuous paths. Diffusions belong to this class of processes and can model a large range of intracellular movements. They are widely used in the biophysical literature [Qian et al., 1991, Saxton and Jacobson, 1997]. Biophysicists distinguish four main types of diffusion, namely Brownian motion (also referred to as free diffusion), superdiffusion, confined diffusion and anomalous diffusion. These different diffusion types correspond to specific biological scenarios as briefly described in [Briane et al., 2019a]. A particle evolving freely inside the cytosol or along the plasma membrane is modeled by free diffusion. Its motion is due to constant collisions with smaller particles undergoing thermal fluctuations. Thus, the particle does not travel along any particular direction and can take a very long time to go to a precise area in the cell. Active intracellular transport suggests that particle motion

is direct. The particles (called in this context cargos) are carried by molecular motors along microtubules and actin filament networks. Superdiffusion models the motion of molecular motors and their cargos. Confined or restricted diffusion [Metzler and Klafter, 2000, Hoze et al., 2012] is characteristic of trapped particles: the particle encounters a binding site, then it pauses for a while before dissociating and moving away. Anomalous diffusion includes particles that encounters dynamics or fixed obstacles [Saxton, 1994, Berry and Chaté, 2014], or particles slowed down by the contrary current due to the viscoelastic properties of the cytoplasm. Very recently, Smith et al. [2017] studied the effect of heterogeneous crowding on the motion of intracellular particles. They assumed that the motion of the particle is hindered by spherical objects, and then derived an advection-diffusion equation to describe the motion of a small particle traveling across an arbitrary distribution of large crowder molecules. Finally, we emphasize that particles can switch from one motion type to another over time; for instance, Fusco et al. [2003] showed experimentally that mRNA molecules undergo cycles between anchoring, free diffusion and superdiffusion (active transport).

An amount of work has been dedicated to the description of motions of individual particles in the cell. The second step consists in understanding the spatial distribution of these motions within the cell and interpreting the underlying biological mechanisms. As a pioneer work, Hummer [2005] studied the spatial mapping of diffusion coefficients using Bayesian inference. This paper inspired other mapping methods (e.g. see [Masson et al., 2014] and [Hoze et al., 2012]). In this paper, we are interested in localizing areas where subdiffusion occurs as in [Hoze et al., 2012]. Regulation of cellular physiological processes relies on molecular interactions (binding and unbinding) at specific places and involves trafficking in confined local microdomains. Our objective is to automatically detect these microdomains from groups of particle trajectories. Hoze et al. [2012] model these microdomains as potential wells that attract intra-cellular particles. The authors use an Eulerian method based on a non-parametric estimation of the drift parameter of the underlying diffusion process. This method assumes a high concentration of particles over the spatial domain of interest to be meaningful. Moreover, this Eulerian approach is not designed to handle a mixture of different diffusion types observed in a small region. All the trajectories in a small region are uniformly averaged, potentially leading to wrong conclusions. To overcome this problem, we propose an original procedure to detect microdomains.

Our method is based on the combination of the clustering algorithm DBSCAN [Ester et al., 1996] and a three-label classification procedure [Briane et al., 2018] which assigns a label (Brownian motion, subdiffusion and superdiffusion) to each trajectory. Our results on simulations mainly demonstrate that:

- (i) our method detects satisfyingly the microdomain shapes;
- (ii) our method is able to handle a mixture of several diffusion types.

The remainder of this paper is organized as follows. In Section 2 we present the mathematical models and the notations used in the paper. In Section 3, we present

related works and focus on the Eulerian method proposed by [Hoze et al., 2012]; we emphasize the underlying assumptions, the key parameters, and the properties of this approach used to detect potential wells within cells. In Section 4, we describe the simulation framework corresponding to the FLUOSIM simulator [Lagardère et al., 2019] used in our experiments. In Section 5, we describe the procedure that combines the DBSCAN clustering algorithm [Ester et al., 1996] with the test procedure described in [Briane et al., 2018]. In Section 6, we evaluate the performance of the computational method on simulated sequences, and we illustrate its potential on real TIRF microscopy image data depicting small zones with a high concentration of subdiffusive vesicles. In Section 7, we discuss the main features of our approach.

2. Notations and mathematical framework

In the sequel, we assume a population of m independent trajectories, with m of order of magnitude of 10^3 particles. We denote $X_t^{(i)} \in \mathcal{D}$ the position of the i^{th} particle at time t , where $\mathcal{D} \subset \mathbb{R}^2$ denotes a spatial bounded domain. Let us define P subregions $\mathcal{S}_1, \dots, \mathcal{S}_P$ such that $\mathcal{S}_p \subset \mathcal{D}$ in which the particles can be trapped (see Fig. 1(a)). The union of subdomains is denoted $\mathcal{S} = \cup_{p=1}^P \mathcal{S}_p$, and $\bar{\mathcal{S}} = \mathcal{D} \setminus \mathcal{S}$ is defined as the complementary domain where the particles undergo a Brownian motion with diffusion coefficient $\sigma_{\bar{\mathcal{S}}}$. We impose that the particles are normally reflected at the boundaries of $\mathcal{D} = \mathcal{S} \cup \bar{\mathcal{S}}$.

Inside a subregion $\mathcal{S}_p \subset \mathcal{D}$, a particle can switch between two states: *trapped* or *non-trapped* (also denoted as *free*):

- (i) the particle is *trapped* in \mathcal{S}_p : the particle motion is a confined Brownian motion normally reflected at the boundary $\partial\mathcal{S}_p$ of \mathcal{S}_p , with a diffusion coefficient $\sigma_{\mathcal{T}} \leq \sigma_{\bar{\mathcal{S}}}$;
- (ii) the particle is *not trapped* in \mathcal{S}_p : the particle motion is a Brownian motion with diffusion coefficient $\sigma_{\mathcal{S}}$.

Formally, we introduce the following indicator variable:

$$\phi(X_t^{(i)}) = \begin{cases} 1 & \text{if the particle } i \text{ is trapped at time } t, \\ 0 & \text{if the particle } i \text{ is not trapped at time } t. \end{cases} \quad (1)$$

Given the binding rate k^+ and the unbinding rate k^- , the probabilities of a particle belonging to \mathcal{S} to be trapped are defined as follows:

$$\begin{aligned} P(\phi(X_{t+h}) = 1 | \phi(X_t) = 0, X_{t+h} \in \mathcal{S}) &= k^+ h + o(h), \\ P(\phi(X_{t+h}) = 0 | \phi(X_t) = 1, X_{t+h} \in \mathcal{S}) &= k^- h + o(h), \end{aligned} \quad (2)$$

where $h > 0$ and $o(h)/h \rightarrow 0$ as $h \rightarrow 0$. In the sequel, the binding and unbinding rates k^+ and k^- are two constants which do not depend on space. Hence, a trapped particle close to the boundaries $\partial\mathcal{S}_p$ has the same probability to unbind to the trapping region \mathcal{S}_p than a trapped particle in the middle of the region \mathcal{S}_p . For the sake of simplicity, we assume that all the subregions \mathcal{S}_p share the same common parameters k^+ , k^- , $\sigma_{\mathcal{T}}$ and $\sigma_{\mathcal{S}}$.

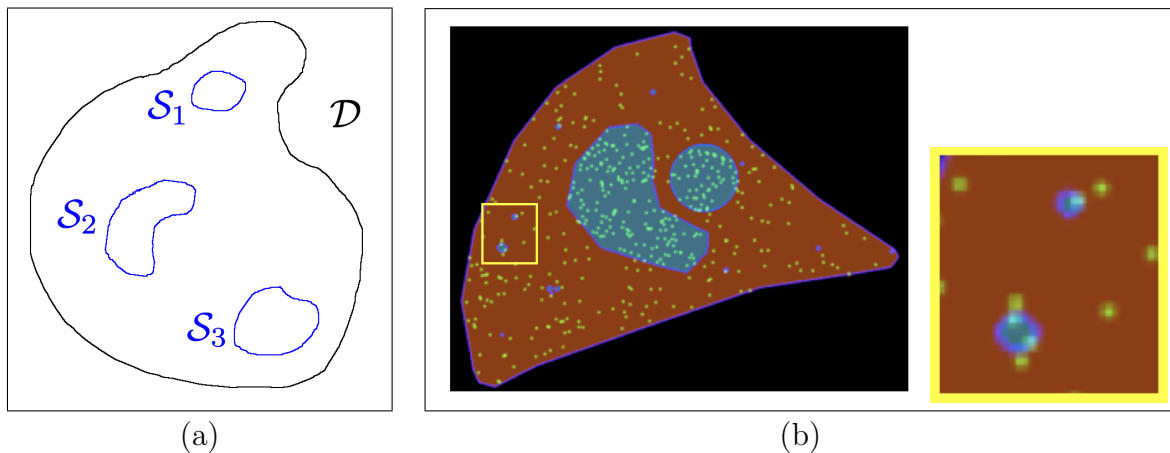


Figure 1. Illustrations of configurations with trapping areas. (a) sketch with three trapping areas ($\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$); (b) example of a simulated image at a given time t , depicting a crossbow-shaped cell and green particles in small trapping (blue) areas and in the cytosol $\bar{\mathcal{S}}$ (orange).

In our estimation framework, we observe a collection of m two-dimensional trajectories \mathcal{X}_m . We denote $\mathbb{X}_{n_i}^{(k)}$ the vector of n_i spatial positions along the trajectory associated to the i^{th} particle:

$$\begin{aligned} \mathbb{X}_{n_i}^{(i)} &= \left(X_{t_0}^{(i)}, \dots, X_{t_{n_i}}^{(i)} \right), \quad i = 1, \dots, m, \\ \mathcal{X}_m &= \left\{ \mathbb{X}_{n_i}^{(i)}, i = 1, \dots, m \right\}. \end{aligned} \quad (3)$$

We assume that each discrete trajectory is generated by a stochastic process $(X_t^{(i)})$ with continuous path defined on the spatial domain $\mathcal{D} \subset \mathbb{R}^2$ and which is a solution of the stochastic differential equation (SDE),

$$dX_t^{(i)} = \mu(X_t^{(i)})dt + \sigma^{(i)}dB_t^{(i),\mathfrak{h}^{(i)}}, \quad t \in [t_0, t_{n_i}], \quad (4)$$

where $B_t^{(i),\mathfrak{h}}$ is a two-dimensional fractional Brownian motion of unknown Hurst parameter $\mathfrak{h}^{(i)} \in (0, 1)$, $\sigma^{(i)} > 0$ is the diffusion coefficient, and $\mu^{(i)} : \mathcal{D} \rightarrow \mathbb{R}^2$ is the drift term. The different trajectories do not need to have the same drift parameter and the same diffusion coefficients, provided that the drift functions and diffusion coefficients do not depend on time.

3. Most related work

In this section, we focus on the two most related methods developed to detect 2D microdomains. First, we describe the Eulerian method proposed by Hoze et al. [2012], based on the two following steps:

- (i) non-parametric estimation of the drift vector field from multiple trajectories;
- (ii) fitting to a parametric drift function.

As starting point, Hoze et al. [2012] model microdomains as potential wells that attract intra-cellular particles. A high density of particles within a local neighborhood

is assumed. Formally, let us consider a collection of m independent trajectories \mathcal{X}_m such that all the trajectories $\mathbb{X}_{n_i}^{(i)} = (X_{t_0}^{(i)}, \dots, X_{t_{n_i}}^{(i)})$ are generated from the common diffusion process (X_t) solution of the overdamped Langevin equation defined as:

$$dX_t = -\nabla U(X_t)dt + \sigma dB_t, \quad (5)$$

where ∇ is the gradient operator, $U : \mathbb{R}^2 \rightarrow \mathbb{R}$ is the potential function, and $\sigma > 0$ the diffusion coefficient. In the sequel, the two-dimensional drift function (or vector field) is defined as $\mu(x) = -\nabla U(x)$. The drift function is defined through the limit as:

$$\mu(x) = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} E(X_{t+\Delta t} - X_t | X_t = x). \quad (6)$$

In [Hoze et al., 2012], an empirical estimator $\hat{\mu}(x)$ of the drift is computed over a square $r \times r$ window $W_r(x)$ centered at point $x \in \mathcal{D} \subset \mathbb{R}^2$, in order to reflect the local conditioning $X_t = x$ in Equation (6). The estimation of $\hat{\mu}(x)$ amounts to computing the average of all displacements of particles passing through the window $W_r(x)$. The estimation of $\hat{\mu}(x)$ is spatially varying and its accuracy depends on the number of trajectory points in $W_r(x)$. Fortunately, we generally observe a high density of particles in trapping areas. In these microdomains, the drift vector field is thus reliably estimated but it is assumed that, locally in the $r \times r$ neighborhood, all the trajectories undergo the same diffusion process with the same drift function. The parameter r is a free parameter of the algorithm, which can be interpreted as a bandwidth parameter in density estimation. In Fig. 2, we illustrate the estimation of drift from trajectories on a artificial sequence depicting two circular trapping areas. A general view of trapping areas with variable sizes generated with FLUOSIM [Lagardère et al., 2019], is shown in Fig. 1(b)).

In the second step, the authors assume that the potential $U(x)$ has a parametric form and can be approximated by a truncated polynomial of second order, defined as:

$$U(x) = \begin{cases} A(\|x - x_0\|_{a,b}^2 - 1) & \text{if } \|x - x_0\|_{a,b}^2 < 1, \\ 0 & \text{otherwise,} \end{cases} \quad (7)$$

where x_0 is the attractor, and $A > 0$ is the depth of the potential modeling the strength of the attractive force toward x_0 , restricted to the underlying 2D ellipse. The parameters (a, b) are the axis lengths of the ellipse of center x_0 in which the particle are under the action of an attractive force. In the computational scheme, the parameters a, b, x_0 and A are assumed partially known and/or estimated by minimizing a least-square error. To conclude, the approach of Hoze et al. [2012] is very appropriate to analyze a large number of tracks as computed in single-particle tracking PALM imaging. The motivation was to compute the residence time of receptors in dendritic spines using short receptor trajectories. The main advantage is the processing of very short tracks. Nevertheless, it cannot handle mixtures or changes of motion occurring in a local neighborhood.

Another Eulerian method was described in [Masson et al., 2014]. The authors propose a Bayesian framework to infer the drift vector field and diffusion coefficients. The idea is first to decompose the image into non overlapping blocks. All the sub-trajectories inside a block (or mesh element) are supposed to be driven by the same

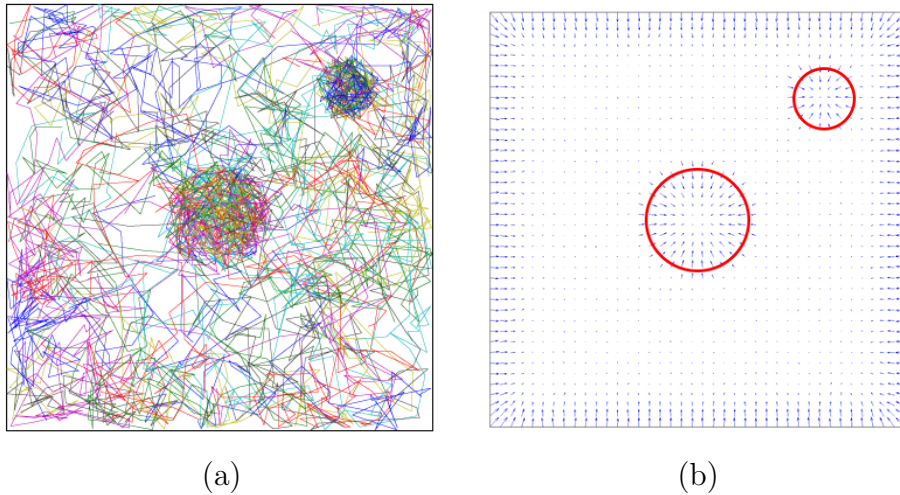


Figure 2. Drift field computed with the method of Hoze et al. [2012]. The FLUOSIM simulation comprises 30 frames of 200×200 pixels as the domain \mathcal{D} is a square of $5\mu m$ and the spatial resolution is $\Delta x = 0.025 \mu m$ (see Table 1). We compute $\hat{\mu}(x)$ on each point of this 200×200 lattice from trajectories (a) ($r = 5$ pixels). We can clearly see the two microdomains \mathcal{S}_1 and \mathcal{S}_2 (b): the vector field converges to the centers of these domains delineated by two red circles (b). We can see that the vector field is orthogonal to the boundaries of the square \mathcal{D} . It is due the normal boundary condition on $\partial\mathcal{D}$. Elsewhere, the $\hat{\mu}(x)$ is almost null, reflecting Brownian motion.

stochastic differential equations (SDE). Instead of using non-parametric estimates of drift and diffusion coefficients as in [Hoze et al., 2012], the authors consider a Gaussian approximation of the SDE to model the likelihood function. Meanwhile, a Jeffreys prior is used as a default prior distribution for the drift and the diffusion coefficient, supposed to be constant in each block. It follows that the *a-posteriori* distribution on the whole space is defined as the product of the *a-posteriori* distributions of each block. An alternative prior is proposed to penalize gradients of the drift field (or the coefficient diffusion field) [El Beheiry et al., 2016]; in that case, the maximum *a-posteriori* (MAP) estimator is more computationally demanding. Finally, once the local drifts (and diffusion coefficients) are estimated, Masson et al. [2014] propose to fit the estimated vector field to a parametric vector field as in [Hoze et al., 2012]. They consider a penalized least square estimator of the vector field to impose a smooth solution. The main advantage of the Bayesian method is that prior information about the drift and diffusion coefficient can be easily incorporated, including errors due to misconnections of particles during tracking.

In this paper, we are interested in the variation over space of particle motion. More specifically, different areas within the cell induce different dynamics. We assume that we can describe these dynamics with a finite collection of motions. We must note that there also exists a vast literature analyzing the switch of motion over time. Again, the motions belong to a finite collection of transport states. Then, Monnier et al. [2015] propose a Bayesian model selection to infer the transient transport states along a trajectory over time. The switches between the different transient states are modeled with a Markov

process. In their case, the different states correspond to Brownian motion with different diffusion coefficients and Brownian motion with drift with different values for both the drift and the diffusion coefficients.

4. Modeling and simulation of trapped particles

In this section, we propose a simulation framework based on general concepts in biophysics to mimic trapping areas. The artificial data sets will serve to evaluate the performance of detection and estimation methods described in Section 4. As said earlier in Section 1, we assume that the particles can switch between the *trapped* and the *free* states. When the particles are trapped, they undergo Brownian motion in confined regions. When the particles are free, they are driven by Brownian motion, only constrained by the boundary conditions on the limits of the domain \mathcal{D} . We model the proportions of *trapped* and *free* particles in the whole domain and in the trapping regions through two systems of differential equations.

4.1. Modeling of particle dynamics

In the aforementioned situation, the dynamics of *trapped* and *free* particles are described by two connected processes:

- (i) the dynamics of individual particles corresponding to normally reflected Brownian motion in \mathcal{D} and in \mathcal{S} when trapped;
- (ii) the trapping process.

The trapping process induces a different motion for the trapped particles; inversely the particles can be trapped only if their motion drive them inside \mathcal{S} .

Formally, we assume m particles independently drawn from the uniform distribution over \mathcal{D} : $X_0^{(i)} \sim \mathcal{U}(\mathcal{D})$, $i = 1, \dots, m$. Note that $n_i = n$ if the trajectory length is the same for all the particles. Instead of considering the exact dynamics of a particle, we simply model its probability to be in any trapping region \mathcal{S}_p (equivalently to be in $\mathcal{S} = \cup_{p=1}^P \mathcal{S}_p$) by:

$$p_s := P(X_t^{(i)} \in \mathcal{S}) = \frac{|\mathcal{S}|}{|\mathcal{D}|}, \quad i = 1, \dots, m. \quad (8)$$

Consequently, it is assumed that at every time t , the spatial point process $(X_t^{(1)}, \dots, X_t^{(m)})$ is a binomial point process over \mathcal{D} of parameter m and which density function is the uniform density [Baddeley et al., 2007]. Furthermore, when the number m of particles is large enough, the evolution of the two populations can be modeled by a system of differential equations depending on parameters p_s , k^+ and k^- .

Let us denote $b(t) = P(\phi(X_t) = 1)$ and $f(t) = P(\phi(X_t) = 0)$ the proportions of trapped particles and free particles respectively. The functions $t \rightarrow b(t)$ and $t \rightarrow f(t)$

are solutions of

$$\begin{cases} \frac{db}{dt} = \gamma_1 f(t) - \gamma_2 b(t), \\ b(t) + f(t) = 1, \end{cases} \quad (9)$$

where γ_1 and γ_2 are respectively the global binding rate and global unbinding rate, such that:

$$\begin{aligned} P(\phi(X_{t+h}) = 1 | \phi(X_t) = 0) &= \gamma_1 h + o(h), \\ P(\phi(X_{t+h}) = 0 | \phi(X_t) = 1) &= \gamma_2 h + o(h), \end{aligned} \quad (10)$$

with $h > 0$ and $h \rightarrow 0$. Consequently, $(\phi(X_t))$ is a continuous-time homogeneous Markov chain with states $\{0, 1\}$ with infinitesimal generator parameters k^+ and k^- [Brémaud, 2013][Chapter 8, Section 2.2]. Finally, we can show that (see Appendix A):

$$\begin{aligned} \gamma_1 &= k^+ p_s, \\ \gamma_2 &= k^-. \end{aligned} \quad (11)$$

The solution of the system (9) is as follows:

$$\begin{cases} b(t) &= \left(b(0) - \frac{k^+ p_s}{k^+ p_s + k^-} \right) e^{-(k^+ p_s + k^-)t} + \frac{k^+ p_s}{k^+ p_s + k^-}, \\ b(t) + f(t) &= 1. \end{cases} \quad (12)$$

As a uniform distribution is assumed, we have the constraint that $b(0) \leq p_s$. In other words, only particles inside \mathcal{S} can be trapped. The situation $b(0) = p_s$ matches with the situation where initially all the particles inside \mathcal{S} are trapped. From Equation (12), the transitory regime is exponential and it converges to an equilibrium proportion:

$$\lim_{t \rightarrow \infty} b(t) = \frac{k^+ p_s}{k^+ p_s + k^-}. \quad (13)$$

If $b(0) < k^+ p_s / (k^+ p_s + k^-)$, function $b(t)$ is decreasing toward its equilibrium point $k^+ p_s / (k^+ p_s + k^-)$. If $b(0) > k^+ p_s / (k^+ p_s + k^-)$, function $b(t)$ is increasing toward its equilibrium point. If $b(0) = k^+ p_s / (k^+ p_s + k^-)$, function $b(t)$ is constant and equal to its equilibrium point. The characteristic time is defined as $\tau_c = 1 / (k^+ p_s + k^-)$. We can consider that the stationary regime is reached when $t > 5\tau_c$.

In the same way, we can model the proportions of trapped particles and free particles inside \mathcal{S} . If $t \rightarrow b_s(t)$ denotes the proportion of trapped particles inside \mathcal{S} and $t \rightarrow f_s(t)$ denotes the proportion of free particle inside \mathcal{S} , we have:

$$\begin{cases} \frac{db_s}{dt} = k^+ f_s(t) - k^- b_s(t), \\ b_s(t) + f_s(t) = 1. \end{cases} \quad (14)$$

As before, the solution has an exponential form. The equilibrium proportion of trapped particles inside any trapping regions \mathcal{S}_p (same for all regions) is as follows

$$\lim_{t \rightarrow \infty} b_s(t) = \frac{k^+}{k^+ + k^-}. \quad (15)$$

Below, we describe the simulation of microdomains generated by running the FLUOSIM software [Lagardère et al., 2019] based on the set of aforementioned equations.

Table 1. Parameters of the first FLUOSIM simulation.

Type	Parameters	Value
Biophysics parameters	σ^2	$1 \mu m^2 sec^{-1}$
	k^+	$0.2 sec^{-1}$
	k^-	$0.05 sec^{-1}$
Microscopy parameters	Δt	$0.1 sec$
	Δx	$0.025 \mu m$

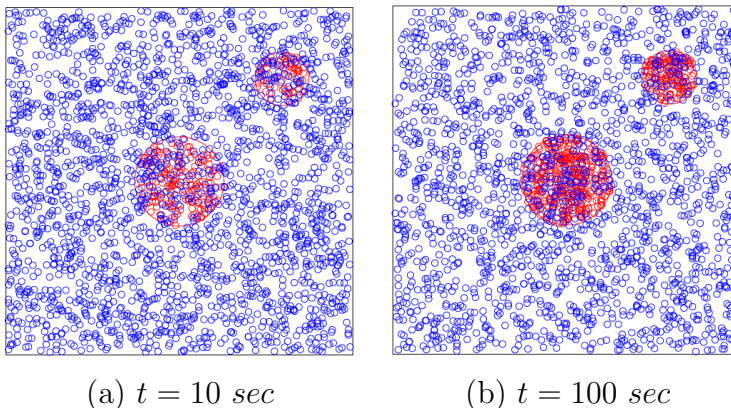


Figure 3. Positions of the particles simulated with FLUOSIM at time $t = 10 \text{ sec}$ and $t = 100 \text{ sec}$ (blue: free particles; red: trapped particles). The red circles delineate the regions \mathcal{S}_1 (large disk) and \mathcal{S}_2 (small disk).

First, we evaluate the model (12) describing the evolution of the proportion of trapped and free particles in the whole domain \mathcal{D} . Second, we assess the model (14) describing the evolution of the proportion of *trapped* and *free* particles in the set of trapping areas \mathcal{S} .

4.2. Simulations of trajectories and microdomains

In our first experiment, we designed two trapping regions \mathcal{S}_1 and \mathcal{S}_2 in a square domain $\mathcal{D} \subset \mathbb{R}^2$ with radius $5 \mu m$ (origin $(0, 0)$ at the bottom left corner of \mathcal{D}):

- (i) \mathcal{S}_1 is a circle with radius $r_1 = 0.65 \mu m$ and center $(2.5, 2.5)$.
- (ii) \mathcal{S}_2 is a circle with radius $r_2 = 0.39 \mu m$ and center $(4, 4)$.

This situation typically corresponds to the analysis of a region of interest as illustrated in Fig. 1(b) (yellow square). The parameters of simulation are given in Table 1. Here, 2500 particles are uniformly distributed over \mathcal{D} . FLUOSIM allows us to identify which particle is trapped at time t . In that case $\phi(X_t^{(i)})$ is known, meaning that if the particle i is trapped or not trapped at time t . In Fig. 3, we plot the positions of the particles labelled as *free* or *trapped* at time $t = 10 \text{ sec}$ (transitory regime) and $t = 100 \text{ sec}$ (stationary regime).

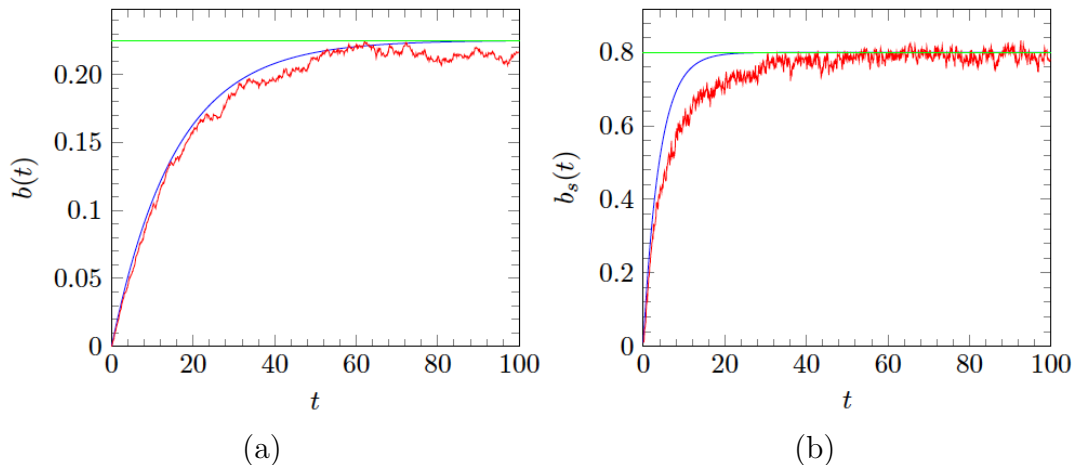


Figure 4. Evolution of the proportions of *trapped* particles over time. (a) proportions of *trapped* particles in \mathcal{D} ; (b) proportions of trapped particles in the trapping region \mathcal{S} . red: true proportion of *trapped* particles computed from the data; blue: proportions computed with model (12); green: asymptote of the model (12). We computed the red curve over $m = 2500$ trajectories simulated with FLUOSIM (see Table 1).

Evaluation of proportions of trapped and free particles in the whole domain

We compare the proportion of trapped particles $t \rightarrow b(t)$ corresponding to Equation (12) to the true proportion of trapped particles. The curves are shown in Fig. (4)(a). The fit of the transitory regime ($t < 5\tau_c = 77.5 \text{ sec}$) is rather good even if the stationary regime is not exactly the same as the one predicted by the model. The predicted equilibrium is 22.53% while the mean true proportion of trapped particles computed on the last 100 steps of time is 21.35%. The relative error of the model is 5.24%, mainly due to the fact that the model oversimplifies the underlying trapping process; the dynamics of the particles are only controlled by the parameter p_s , and we assume that the particles are always in the stationary regime of a normally reflected Brownian motion on \mathcal{D} . This assumption holds at the beginning, as the particles are initially drawn from the uniform distribution on \mathcal{D} . Nevertheless, as t increases, the trapping process makes this assumption invalid. This explains the relative good fit of the transitory phase and the relative lack of fit of the asymptotic phase. However, we can consider that it is a rather satisfying modeling, regarding the flexibility and the number of parameters.

Evaluation of proportions of trapped and free particles in trapping areas

We compare the proportion of trapped particles $t \rightarrow b_s(t)$ given by Equation (14) to the true proportion of trapped particles inside \mathcal{S} . Interestingly, we observe the opposite behavior compared to the case of $b(t)$ (proportion of trapped particles in the whole domain \mathcal{D}). There is a clear lack of fit during the transitory regime but a perfect fit during the stationary regime (see Fig. (4)(b)). We can see that the transitory regime has the same duration for $b(t)$ and $b_s(t)$. During this phase, the number of particles

inside \mathcal{S} increases due to the process of Brownian particles entering in \mathcal{S} and getting trapped. However, it is explicitly assumed in Equation (14) that the number of particles is constant (and large). This explains the lack of fit during the transitory phase which matches with a period during which the number of particles increases in \mathcal{S} . On the contrary, once $b(t)$ has reached the stationary regime, the number of particles inside \mathcal{S} is approximately constant (even if it is not the same particles that remain in \mathcal{S} from one time to another). In the latter case, the model (12) is relevant and the stationary regime is well predicted by the model.

5. Clustering of trajectories for microdomain detection

In this section, we describe the computational approach used to automatically detecting microdomains where particles undergo subdiffusion. The proposed approach is not based on the biophysical modeling and the FLUOSIM simulator presented earlier. FLUOSIM serves here to quantitatively demonstrate the performance of our approach on synthetic sequences (ground truth).

Our four-step detection procedure is based on the combination of the clustering algorithm DBSCAN [Ester et al., 1996] and the labeling of trajectories as explained in [Briane et al., 2018]. The procedure can be used for a large range of application problems in live cell imaging, and is summarized as follows (see Fig. 5):

- Step #1: Classification of trajectories:* we run our test procedure [Briane et al., 2018] on the collection of trajectories \mathcal{X}_m (Fig. 5(b)) to label the tracks into three diffusion groups: free diffusion (Brownian motion), superdiffusion, and subdiffusion. We denote $\Omega_1(\mathcal{X}_m)$ the set of trajectory indexes corresponding to the acceptance of hypothesis H_1 (subdiffusion).
- Step #2: Representative point selection:* we choose a unique point \tilde{x}_i to represent each trajectory $\mathbb{X}_{n_i}^{(i)}$ (Fig. 5(c)). We remove points associated to superdiffuse trajectories and Brownian trajectories (Fig. 5(d)).
- Step #3: Point clustering:* We partition the set $\Omega = \{\tilde{x}_i | i \in \Omega_1(\mathcal{X}_m)\}$ into clusters (Fig. 5(e)).
- Step #4: Detection of trapping areas:* We use the clusters to delineate confinement areas and microdomains (Fig. 5(f)).

In the sequel, we successively present *Steps #2, #3 and #4* of the method. As already explained, we will use the simulation framework presented in Section 3 to illustrate the different steps of our method.

5.1. Classification of trajectories

Step #1 is straightforward as it corresponds to the test procedure described in [Briane et al., 2018]. This method is based on a non-parametric three-decision test to distinguish the three types of diffusion (superdiffusion, subdiffusion, free diffusion). The null hypothesis is that the observed trajectory made of n points is generated from a Brownian

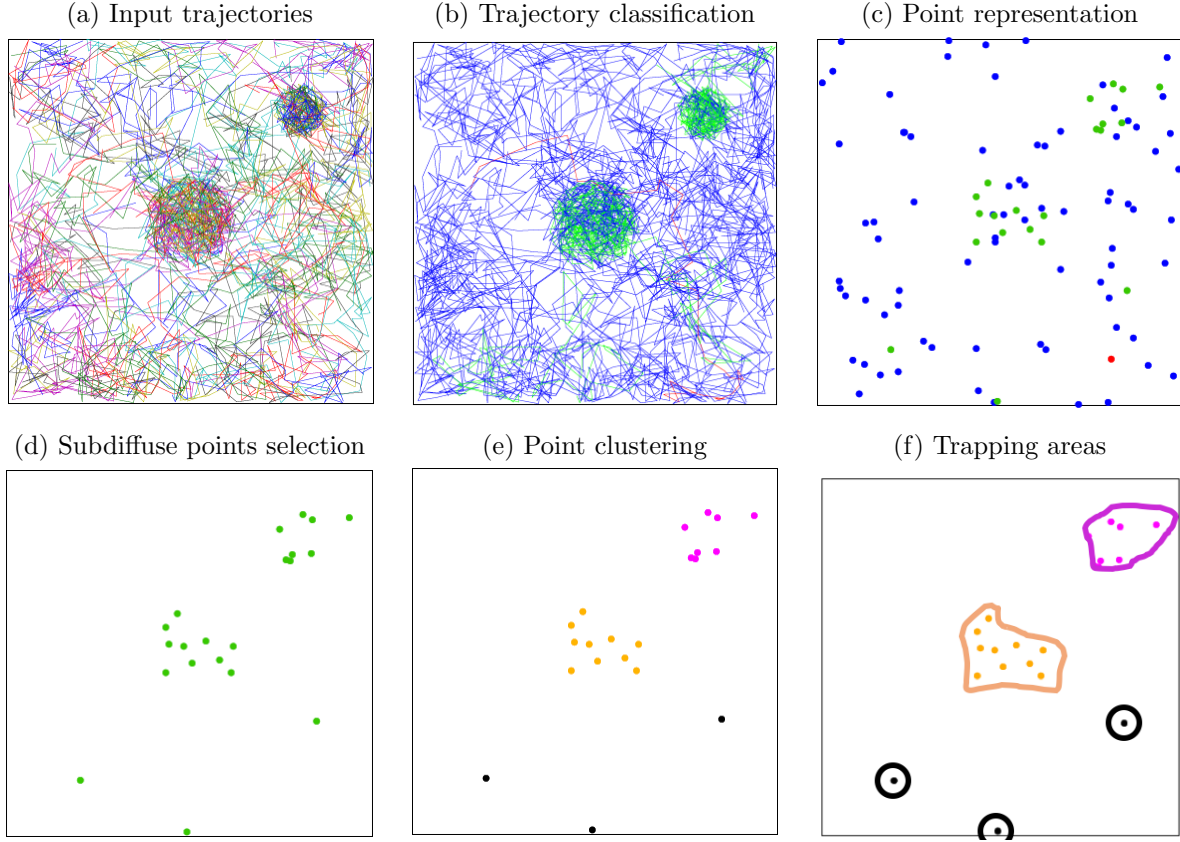


Figure 5. Overview of our computational approach for detecting trapping areas. (a) input trajectories; (b) classification of trajectories into three diffusion groups (superdiffusion (red), subdiffusion (green), free diffusion (blue)); (c) representative points on trajectories; (d) display of subdiffusive representative points (green); (e) estimation of clusters associated to trapping areas; (f) detection of trapping areas.

motion and the two other alternatives are subdiffusion and superdiffusion. The test statistic T_n is the standardized largest distance covered by the particle from its starting point. It can be interpreted as follows: (i) if the value of T_n is low, it means that the process stayed close to its initial position and can be trapped in a microdomain (subdiffusion); (ii) if the value of T_n is high, the particle went far from its initial position and the particle is actively transported (e.g. via a molecular motor) (superdiffusion). The test is appropriately normalized, asymptotically converges under the null hypothesis and under parametric alternatives which are currently considered in the biophysics literature. The test procedure, applied to subdiffusive trajectories, is easily controlled by a p -value.

Our procedure is applied to detect areas where subdiffusion occurs, but it can also be used to detect areas in which superdiffusion or Brownian motion are the main dynamics. In that case, we just need to replace the set $\Omega_1(\mathcal{X}_m)$ by respectively $\Omega_0(\mathcal{X}_m)$ (set of trajectory indexes corresponding to Brownian motion) or $\Omega_2(\mathcal{X}_m)$ (set of trajectory indexes corresponding to superdiffusion).

Moreover, if we observe long trajectories, dynamics can vary, switching from one diffusion model to another. In this case, labelling a trajectory with a single diffusion class would be misleading. To overcome this problem, we can use the algorithm described in [Briane et al., 2019b] (also based on the classification proposed in [Briane et al., 2018]) to detect the change points where the diffusion switches occur and segment the trajectories accordingly. Then we can use these sub-trajectories as inputs of the proposed algorithm.

Finally, it is possible that, due to the spatial extensions of the trapping regions, a trapped particle never hits the edges of the region during the period of observation. In this case, these particles have no chance to be detected as subdiffusive by the procedure described in [Briane et al., 2018]. If we assume that the trapped particles have a significantly lower diffusion coefficient than the untrapped particles – a realistic biological assumption – we can use this supplementary assumption to detect the trapped particles even in this case. Consequently, we proceed as follows. We keep all the particles classified as subdiffusive as before. For all the particles classified as Brownian we can compute an estimator of the diffusion coefficient as follows:

$$\hat{\sigma}_n^2 = 1/(2n\Delta) \sum_{j=1}^n \|X_{t_j} - X_{t_{j-1}}\|_2^2.$$

In the next step, we classify the trajectories first detected as Brownian by using the method [Briane et al., 2018] based on the estimated diffusion coefficient, and we re-label as subdiffusive all the trajectories detected as Brownian by the method [Briane et al., 2018] for which the estimated diffusion coefficient is low enough. A threshold must be set by either a statistical procedure – for example based on the value of the diffusion coefficients of the trajectories detected as subdiffusive in the first place – or by an expert. These trajectories would correspond to the subdiffusive trajectories that never hit the edges of the subregions. We can then run the next steps of the procedure on this new set of subdiffusive trajectories.

5.2. Representative trajectory points and spatial distribution

In *Step #2*, a single point is required to represent a trajectory made of n points. In what follows, we consider mapping functions $f : \mathbb{R}^{2 \times n} \rightarrow \mathbb{R}^2$ of the following form:

- Mean point : $f(x_1, \dots, x_n) = \frac{1}{n} \sum_{i=1}^n x_i$,
- j -th point located on the trajectory: $f(x_1, \dots, x_n) = x(j)$, with $j \in \{1; \dots, n\}$.

As already mentioned, the trapped particles are modeled by confined Brownian motion normally reflected at the boundaries of the trapping regions. In that case, we can see that different representative trajectory points yield very different spatial patterns. As illustrated in Fig. 6, we represent the trajectory $\mathbb{X}_n = (X_{t_0}, \dots, X_{t_n})$ by the mid-point $X_{\lfloor n/2 \rfloor}$ (related function $f(x_1, \dots, x_n) = x(\lfloor n/2 \rfloor)$). It turns out that the spatial distribution of the representative points $X_{\lfloor n/2 \rfloor}^{(i)}$ of the trapped trajectories $\mathbb{X}_n^{(i)}$ is uniform with this choice, over the trapped region \mathcal{S}_1 and \mathcal{S}_2 (Fig. 6(b)). This is due to

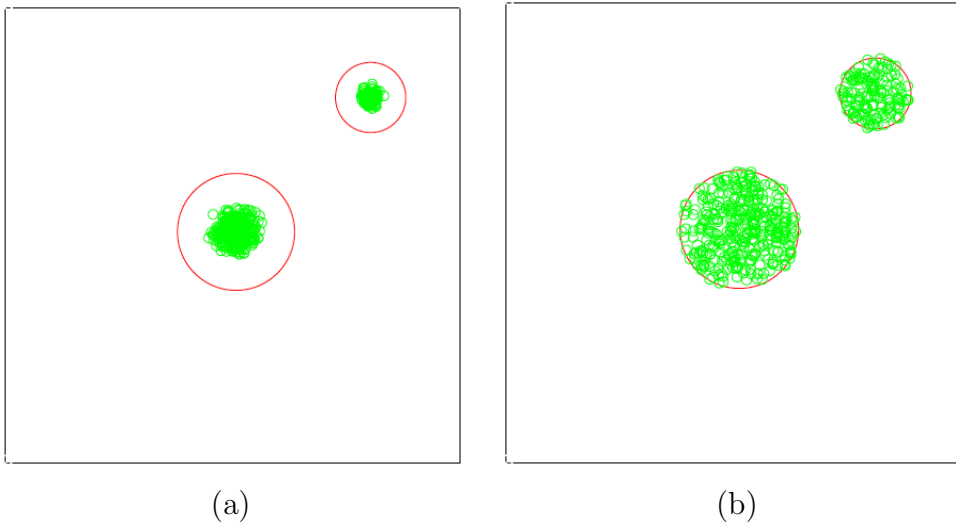


Figure 6. Spatial distribution of the trapped particles in the FLUOSIM simulation; (a) the trajectories are represented by their mean point; (b) the trajectories are represented by $X_{[n/2]}$. A trajectory is considered to be trapped if it is trapped during the whole period of observation. The red circles represent the boundaries of the trapping regions \mathcal{S}_1 (large circle) and \mathcal{S}_2 (small circle).

Proposition 1, as the trapped particles undergo a normally reflected confined Brownian motion in \mathcal{S}_1 and \mathcal{S}_2 .

Proposition 1 Pinsky [2003]. Let (X_t) be a normally reflected Brownian motion on a finite volume domain \mathcal{D} . Assume that the process (X_t) has a stationary uniform distribution over \mathcal{D} : $\mathcal{U}(\mathcal{D})$. If the process (X_t) has reached its stationary distribution, then for any $t > 0$ and $\mathcal{B} \subset \mathcal{D}$ it follows that

$$P(X_t \in \mathcal{B}) = \frac{|\mathcal{B}|}{|\mathcal{D}|}, \tag{16}$$

where $|\mathcal{B}|$ denotes the area of domain $|\mathcal{B}|$.

Actually, the average position of the trapped trajectories are concentrated in the center of the trapped regions (Fig. 6(a)). Therefore, in the purpose of accurately estimating the boundaries of \mathcal{S}_1 and \mathcal{S}_2 , it seems more appropriate to represent the trajectory i with $X_{[n/2]}^{(i)}$ rather than by the average position, not necessary located on the trajectory.

5.3. A clustering algorithm: DBSCAN

In *Step #3*, we apply the popular clustering algorithm DBSCAN [Ester et al., 1996] to robustly estimate point clusters from *noisy points* due to presence of false detections. DBSCAN adds a point into a cluster if it is surrounded by at least κ neighbouring points in a disk of radius ϵ – this type of point is called a *core point* – or if it is within a disk of radius ϵ from a core point – this type of point is called a *boarder point* –. On the contrary, the point does not belong to any cluster and is considered as noise. Other

more sophisticated clustering algorithms could be applied. For instance, the so-called *a contrario* method allows a finer analysis of the clusters than DBSCAN through the concept of meaningful clusters [Desolneux et al., 2003, Cao et al., 2007] organized in a hierarchical structure. We focus on the DBSCAN algorithm since it is very popular spatial statistics. It is mainly controlled by two parameters (ϵ, κ) as explained below.

5.3.1. Selection of the DBSCAN parameters The set of parameters (ϵ, κ) should be adapted to each cluster, as each cluster does not have the same density (or concentration) of points. However this information is available once the clusters are estimated. In practice, a unique set of parameters (ϵ, κ) is used for all clusters. A good choice is to set the parameters (ϵ, κ) in order to detect the least dense cluster. Indeed, the clusters with more points are more easily detected than the least dense clusters. Ester et al. [1996] argue that, in two-dimensional problems, we can set $\kappa = 4$ as a rule of thumb. Once the parameter κ is set, we can choose ϵ by considering two different strategies:

- (i) a data driven method based on the observed distribution of κ -nearest neighbors;
- (ii) a parametric method to approximate the distribution of *noisy points*.

In the following, we present two data driven methods and two parametric methods for selecting ϵ . Then, we compare the outcomes of the DBSCAN algorithm with different methods for selecting ϵ , applied to FLUOSIM simulations.

Data driven methods The data driven methods enable to determine ϵ from the sample $d_1(\kappa), \dots, d_m(\kappa)$ where $d_i(\kappa)$ denotes the distance of the κ -nearest neighbors of point $x_i \in \Omega$. Let us denote $d_{(1)}(\kappa), \dots, d_{(m)}(\kappa)$ the increasing-ordered sample. Note that if $\epsilon = d_{(i)}(\kappa)$, the i points corresponding to the values $d_{(1)}(\kappa), \dots, d_{(i)}(\kappa)$ will be *core points* while the other points will be either *boarder points* or *noisy points*. The choice $\epsilon < d_1(\kappa)$ corresponds to the case when all the points are *noisy points* (no *core point*), whereas the choice $\epsilon \geq d_m(\kappa)$ corresponds to the case where all the points are *core points*. In the latter case, all the points belong to one single cluster. If the number of *noisy points* \aleph is known, a natural choice proposed by Ester et al. [1996] is $\epsilon = d_{(m-\aleph+1)}(\kappa)$. A alternative graphical approach is also described in [Ester et al., 1996] to determine ϵ . The authors plot the sequence $d_{(m)}(\kappa), \dots, d_{(1)}(\kappa)$ and ϵ is determined as the first $d_{(i)}$ in the first valley of the sequence.

Parametric methods In [Daszykowski et al., 2001], it is implicitly assumed that the *noisy points* are uniformly distributed. The parameter ϵ is defined as the α quantile of the distribution (e.g. $\alpha = 0.05$) of the κ -nearest neighbors of $|\Omega|$ points drawn from a uniform distribution over \mathcal{D} . If the observed *noisy points* are really drawn from a uniform distribution, none of them will be chosen as a *core point* with approximately $1 - \alpha$ probability.

As the number \aleph of noisy points and the clusters C_i are unknown, it is not possible to propose a method which exactly controls the probability of detecting a *noisy point* as

Table 2. Values of the parameter ϵ of DBSCAN obtained with different methods. The parameter ϵ is estimated assuming $\kappa = 4$ as recommended in [Ester et al., 1996]. The parameter ϵ is expressed in pixel units. The image size is 200×200 pixels.

Type	Method	ϵ
Data driven	Graphical method	5.64
Parametric	[Daszykowski et al., 2001]	5.94
	[Friedman et al., 1975]	5.86

a *core point*. In [Daszykowski et al., 2001], ϵ is estimated with Monte-Carlo simulations, which is time consuming if $|\Omega|$ is large.

It is worth noting that Friedman et al. [1975] also studied the distribution of the κ -nearest neighbors of m points uniformly distributed over a finite domain \mathcal{D} . If $|\Omega|$ is large enough to neglect boundary effects, the ratio of the volume of a two-dimensional sphere centered at a point containing κ neighbors and the volume of the whole space \mathcal{D} is governed by a Beta distribution f with parameters $(\kappa, |\Omega| - \kappa)$:

$$f(x) = \frac{m!}{(\kappa - 1)!(m - \kappa)!} x^{\kappa-1} (1-x)^{m-\kappa}, \quad 0 \leq x \leq 1. \quad (17)$$

Finally, we can define ϵ as explained in [Daszykowski et al., 2001]. If $F^{-1}(\alpha)$ denotes the quantile of order α of the distribution f , we get:

$$\epsilon = \left(\frac{|\mathcal{D}| F^{-1}(\alpha)}{\pi} \right)^{1/d}, \quad (18)$$

where $|\mathcal{D}|$ is the volume (or area in the two-dimensional case) of \mathcal{D} .

5.3.2. Comparison of strategies on FLUOSIM simulations We apply the DBSCAN algorithm on the set Ω obtained with the test procedure [Briane et al., 2018] (see Fig. 5(b)). The parameter ϵ is estimated with the different methods. All the methods provided similar estimations of ϵ in our experiments (e.g. see Table 2). Consequently, we got the same set of clusters when we run DBSCAN as shown in Fig. 7 by setting $\kappa = 4$. We found two clusters as expected, corresponding to the two trapping regions \mathcal{S}_1 and \mathcal{S}_2 . We got similar results in additional experiments, suggesting that selecting one method or another one would not affect the detection of trapping areas.

5.4. Detection of trapping areas

In *Step #4*, we simply use the convex hull to define the borders of the clusters. By definition of the convex hull, all the points of the cluster will be either inside the convex hull (or convex polygon) or be some vertexes of the polygon. We note that several extensions of the convex hull have been developed, including α -shapes [Edelsbrunner et al., 1983]. In that case, an extra parameter $\alpha > 0$ is used to control the smoothness of the estimated shape.

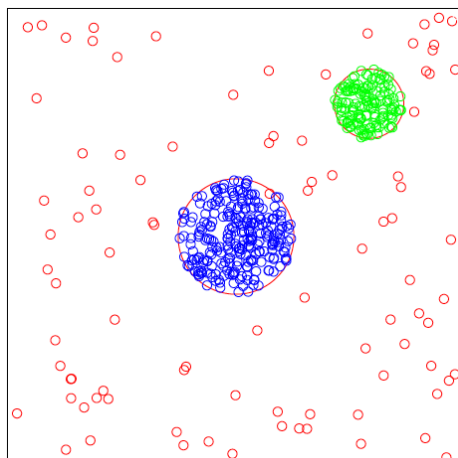


Figure 7. Clusters detected by DBSCAN ($\kappa = 4$) on the set of particles detected as subdiffusive with the test procedure [Briane et al., 2018] (single test with a significance level of 5%). We choose $\epsilon = 5.86$ derived with the Beta distribution assumption. The *noisy points* are in red, and points belonging to the same cluster are labelled with the same color. The red circles represent the boundaries of the trapping regions \mathcal{S}_1 (large circle) and \mathcal{S}_2 (small circle). The black square delineates the borders of the whole domain \mathcal{D} .

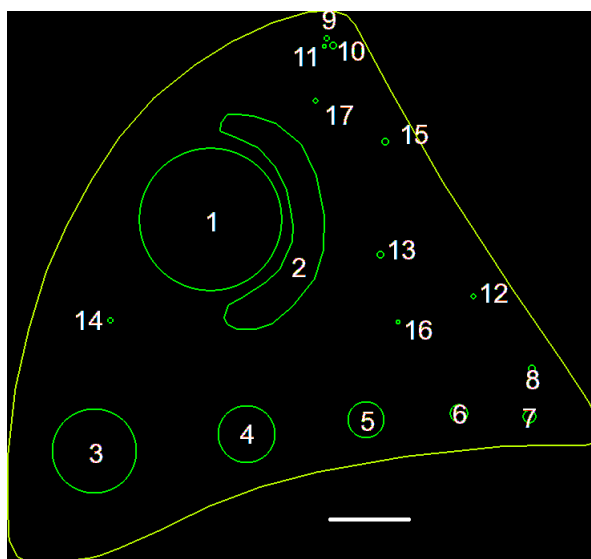


Figure 8. Simulated crossbow-shaped cell with seventeen trapping regions of different sizes and shapes. The white line at the bottom accounts for $16 \mu\text{m}$.

6. Experimental results

6.1. Evaluation on FLUOSIM simulated sequences

In this section, we use the FLUOSIM simulation scheme to illustrate the potential of our approach to detect trapping areas. We consider temporal series depicting 4500 trajectories made of $n = 350$ points. The trajectories are analyzed once the equilibrium

Table 3. FLUOSIM parameters to simulate the first crossbow-shaped cell.

Type	Parameter	Value
Biophysics parameters	σ_S^2	$0.2 \mu m^2 . s^{-1}$
	σ_T^2	$0.02 \mu m^2 . sec^{-1}$
	k^+	$10 sec^{-1}$
	k^-	$2 sec^{-1}$
Microscopy parameters	Δt	$0.02 sec$
	Δx	$0.160 \mu m$

Table 4. Evaluation of the procedure on the crossbow-shaped cell simulation.

	True regions					
Region index (see Fig. 8)	1	2	3	4	5	6-17
Number of overlapping clusters	1	1	1	1	1	0
Maximum overlap (%)	90	36	60	29	15	0

regime is reached. We recall that the trajectories can switch between two diffusion modes:

- (i) confined Brownian in small microdomains \mathcal{S}_p with normal reflection on the boundaries $\partial\mathcal{S}_p$;
- (ii) Brownian motion in the whole domain \mathcal{D} with normal reflection on the boundaries $\partial\mathcal{D}$.

We designed a cell in with a crossbow shape and seventeen trapping regions depicted in Figure 8:

- one circular nucleus surrounded by one crescent-shaped zone,
- five circular zones with decreasing radius,
- seven tiny circular zones,
- three tiny circular zones really close to each other.

The parameters are given in Table 3. In order to assess the efficiency of the method, we compute two criteria for each true (ground truth) trapping region. First, we compute the number of detected regions overlapping with the real trapping regions. Secondly, we compute the proportion of overlap between the closest detected region to the trapping region (useful if multiple detected regions overlap with the true region). We define the proportion of overlap as the area of intersection of the true region and the detected region over the area of the true region. Then, ideally the two criteria should be equal to one, meaning that one detected region corresponds to one real trapping region and that the overlap is perfect. Also, for each detected cluster we compute the proportion of subdiffusive trajectories which should be compared to Equation (15). In this case, the theoretical proportion is $k^+/(k^+ + k^-) = 0.8333$.

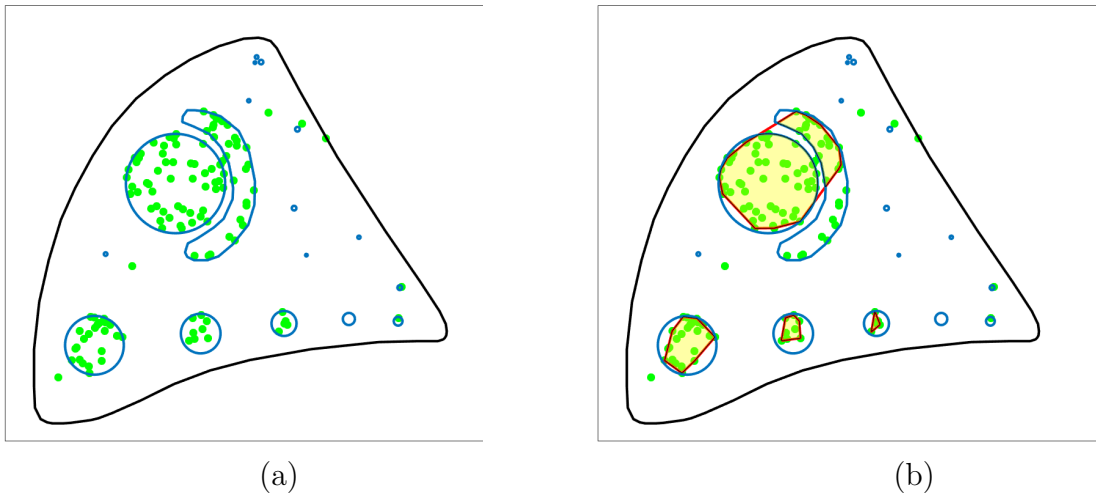


Figure 9. Results of our procedure on the first crossbow-shaped cell simulation; (a) representative points of the trajectories detected as subdiffusive by the test procedure [Briane et al., 2018] with a significance level of 5%; (b) trapping regions (yellow background, red boundaries) estimated with the convex hull from the clusters detected by DBSCAN. The true trapping regions are delineated with blue curves.

Table 5. Proportion of subdiffusive trajectories inside the detected regions from the crossbow-shaped cell simulation.

Cluster index	Proportion of subdiffusive trajectories
1	3.3 %
2	5.6 %
3	18.5 %
4	8.3 %

From Table 4, we can see that the algorithm detects satisfyingly the regions #1 to #5 while it does not detect at all the regions #6 to #17. The best detected region is region #1 with 90% of overlap. The algorithm does not detect the small regions as there is either no detected subdiffusive trajectories or less than $\kappa = 4$ detected subdiffusive trajectories (see Figure 9 (a)). In the latter case, DBSCAN can not build a cluster as the minimum number of points in a cluster must be $\kappa = 4$. We also note that the proportion of subdiffusive trajectories in the five detected regions is much lower as reported in Table 5, than the expected value $k^+ / (k^+ + k^-) = 0.8333$. Then, in the setting of this simulation, the algorithm under-detects subdiffusive trajectories. The parameters that influence the most the outcome of the classification step (*Step #1*) are the trajectory length, the microdomain size, and the diffusion coefficient of trapped particles. An optimal setting is a long trajectory with a high diffusion coefficient in a small domain. In this case, the particles bounces a lot against the edges of the trapping regions and its subdiffusive behavior is obvious. A more detailed simulation scheme would give the interval of parameters for which the classification step performs the best.

Another issue is the proximity between two trapping regions. We can see in Figure 9 (b) that a detected region overlaps both with region #1 (large disk) and region #2 (crescent shape). We note that the minimum distance to resolve two close regions will depend on the concentration of subdiffusive trajectories, that is on the binding parameters k^+ and k^- . In fact, the outcome of DBSCAN depends on the density of points inside the clusters. In Appendix B, we provide another simulation controlled by the parameters given in Table 6. The results are shown in Figure 12 and the quantitative results are reported in Tables 7 and 8. In conclusion, we did not choose the optimal simulation parameters here to assess our method. Still, we are able to detect some trapping regions even if the algorithm under-detects subdiffusive trajectories and merges close trapping regions. A more systematic simulation study should be performed to determine the range of parameters for which the method gives satisfying results.

6.2. Experiments on real TIRF images: dynamics of Rab11 involved in exocytosis

In this section, we demonstrate the potential of our approach on Rab11-mCherry proteins involved in exocytosis mechanisms and observed in 2D TIRF (Total Internal Reflection Fluorescence) microscopy (Nikon TE2000 inverted microscope equipped with a $100\times$ TIRF objective: numerical aperture = 1.49, exposure time $\Delta t = 100$ msec, $dx = dy = 160$ nm). Figure 10 shows three 256×353 images at different time points, depicting fluorescent vesicles close to the plasma membrane. The image sequences are typically composed of 600 images, and show the late steps of exocytosis driven by Rab11 at the plasma membrane, namely the tethering-docking process [Schafer et al., 2014].

Given several hundreds of Rab11 trajectories estimated with the Icy software [Chenouard et al., 2013, 2014], we first applied our test procedure [Briane et al., 2018] to classify the trajectories. In Fig. 11(a), we displayed all trajectory points onto a 2D three-color map: Brownian motion (blue), subdiffusion (green), and superdiffusion (red). In the particular case shown in Fig. 10, the M10 cells were treated with Latrunculin A, which inhibits actin polymerization and then reduces active transport of vesicles [Pécot et al., 2018]. This effect is confirmed on Fig. 11(a) since no track was labeled a superdiffusive trajectory.

Second, we applied our computational approach on subdiffusion trajectories (green pixels, left) represented by points in Fig. 11(a) (right). In the sequel, we assume that the tethering-docking process under study is associated to subdiffusion because of multiple interactions with actin filaments and interactions with the plasma membrane. All points reported in Fig. 11(a) (right) correspond to vesicles that fuse at the surface of the plasma membrane. First, we observe an approximately uniform spatial distribution of points (black dots). Nevertheless, we detected a few additional clusters corresponding to higher densities of fusion events in a local spatial neighborhood. These clusters can be interpreted as local domains related to more intensive interactions with the plasma membrane and actin filaments. Several clusters are actually localized in regions with a high concentration of spots (e.g. see top right of the cell). Trajectory analysis is

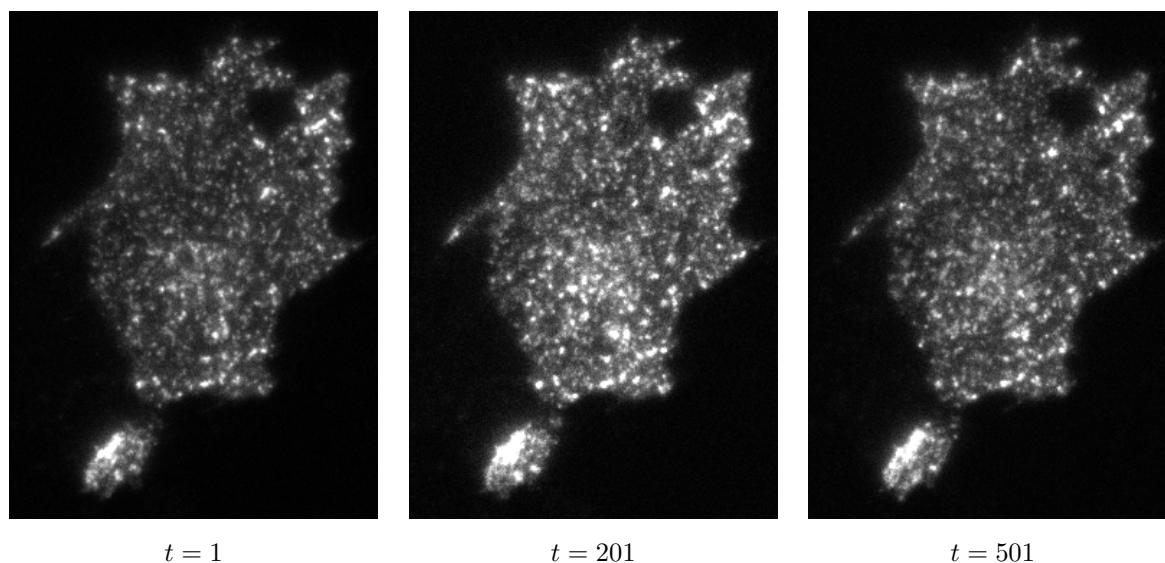


Figure 10. Rab11-mCherry proteins within a M10 cell treated with Latrunculin and observed at several time points. The images are extracted from a 2D TIRFM sequence composed of 600 images (exposure time $\Delta t = 100 \text{ msec}$, $dx = dy = 160 \text{ nm}$).

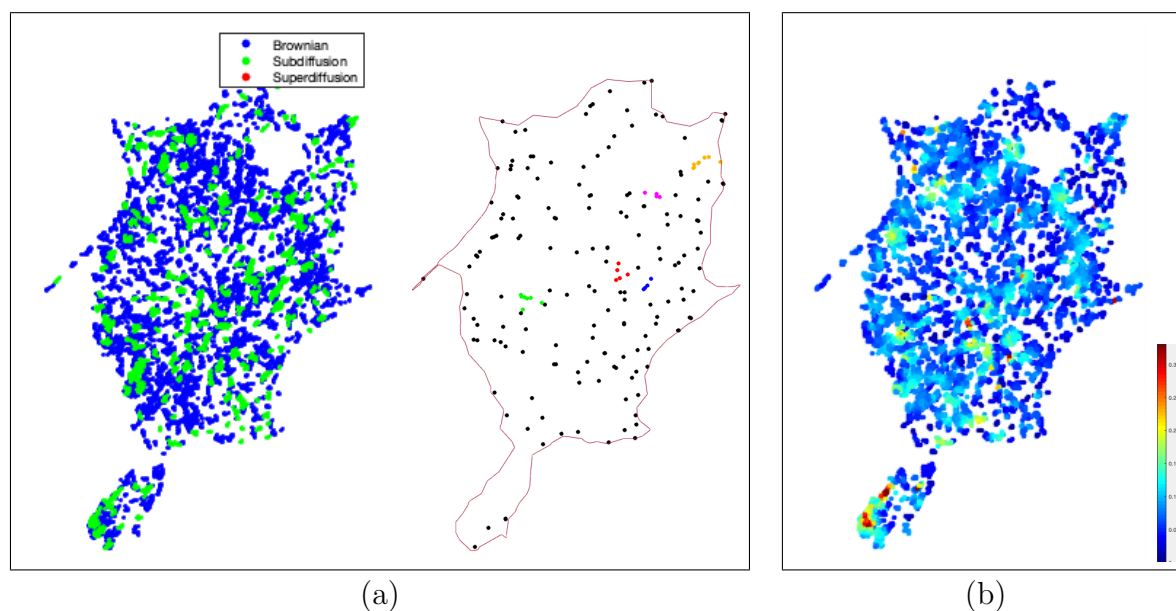


Figure 11. Analysis and clustering of trajectories. (a) Trajectory classification into three diffusion groups [Briane et al., 2018] (left) and selection/clustering of subdiffusion trajectories (right). The point clusters are labeled with colors while individual subdiffusion trajectory are represented with black dots. (b) Adaptation of the Hoze et al. [2012] method to estimate diffusion coefficients on a sparse set of unclassified trajectories [Salomon and Kervrann, 2020].

then helpful to focus the attention on specific areas which cannot be easily detected by examining fluctuations of fluorescence intensities shown in Fig. 10.

Finally, we illustrate in Fig. 11(b) the potential of the Hoze et al. [2012] method on the same sequence. In particular, we reported the spatially varying diffusion coefficients on a 2D map since the drift vectors are very noisy and not interpretable. In our study, the diffusion map is actually not really helpful to draw conclusions if the drift values are noisy. The number of estimated trajectories (about one thousand) is too small to safely apply the methods described in [Hoze et al., 2012, Salomon and Kervrann, 2020].

7. Discussion

In this paper, we presented a method to detect the trapping regions where the particles are confined and thereby undergo subdiffusion. We combined the clustering algorithm DBSCAN [Ester et al., 1996] and a test procedure [Briane et al., 2018] in order to identify the areas with a high concentration of subdiffusive particles, assumed to correspond to trapping regions. With this approach, we were also able to estimate the proportion of particles inside the trapping regions which were indeed confined in the domain. Our approach is more flexible than the Eulerian approach of Hoze et al. [2012], which is not able to handle mixtures of motions, or to capture the motion of particles not confined in the trapping regions but still going through these regions.

We illustrated the potential of our procedure on simulations generated with the FLUOSIM software, and on real TIRF images. However, the simulation results provided here do not intend to assess systematically the procedure for a large range of parameters values (k^+ , k^- , σ_S , σ_T , the shapes of the regions). For future works, it would typically involve:

- determining jointly the region sizes and trajectory lengths for which the classification procedure can detect subdiffusive trajectories (*Step #1*) and further defining a criterion to select the region size for different shapes;
- determining the minimum distance between two regions for the clustering algorithm to be able to detect two distinct regions and not merge them. This distance is expected to depend on the concentration of subdiffusive trajectories in the trapping regions.

In this paper, we focused on the modeling and estimation framework and demonstrated its potential with preliminary experiments. Then, we emphasize the fact that it is possible to use different procedures at each step of the proposed pipe-line to adapt to different scenarios. For instance, in the case of trajectories switching over time, we can use sub-trajectories estimated with the method described in [Briane et al., 2019b], instead of the classification procedure [Briane et al., 2018] used in *Step #1* of our algorithm. The same way, another choice of representative point and other strategies of clustering or shape estimation can also be used respectively in *Steps #2 #3* and *#4*.

We proposed a default choice of procedures for each step of the pipeline but the user is free to select the appropriate procedures depending on the application.

Funding

Funding was provided by Inria Rennes and CREST-Ensaï-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).

Acknowledgement

We thank M. Lagardère (Interdisciplinary Institute for Neuroscience, Bordeaux, France) for assistance with the FLUOSIM software and for helpful discussion.

Appendix A: Derivation of the global binding and unbinding rates

We demonstrate that the equalities (12) hold. Let us consider the equality involving γ_1 :

$$\begin{aligned} P(\phi(X_{t+h}) = 1 | \phi(X_t) = 0) & \quad (19) \\ &= P(\phi(X_{t+h}) = 1, X_{t+h} \in \mathcal{S} | \phi(X_t) = 0) + P(\phi(X_{t+h}) = 1, X_{t+h} \in \overline{\mathcal{S}} | \phi(X_t) = 0), \\ &= P(\phi(X_{t+h}) = 1 | \phi(X_t) = 0, X_{t+h} \in \mathcal{S})P(X_{t+h} \in \mathcal{S} | \phi(X_t) = 0) + 0. \end{aligned}$$

The particle can not be trapped outside \mathcal{S} . Consequently, the second probability of the sum is zero. As the probability to be in region \mathcal{S} at $t + h$ is independent from the event $\{\phi(X_t) = 0\}$ to be a free at t , we finally get

$$P(\phi(X_{t+h}) = 1 | \phi(X_t) = 0) = (k^+h + o(h))P(X_{t+h} \in \mathcal{S}). \quad (20)$$

Normally reflected Brownian motion in a domain \mathcal{D} with finite volume has a stationary distribution. This distribution is the uniform distribution over \mathcal{D} [Pinsky, 2003]. Then, under the uniform distribution assumption, we get:

$$P(\phi(X_{t+h}) = 1 | \phi(X_t) = 0) = (k^+h + o(h))\frac{|\mathcal{S}|}{|\mathcal{D}|} = k^+\frac{|\mathcal{S}|}{|\mathcal{D}|}h + o(h). \quad (21)$$

It follows that $\gamma_1 = k^+|\mathcal{S}|/|\mathcal{D}|$.

Now we prove the second equality of (12) involving γ_2 .

$$\begin{aligned} P(\phi(X_{t+h}) = 0 | \phi(X_t) = 1) & \quad (22) \\ &= P(\phi(X_{t+h}) = 0, X_{t+h} \in \mathcal{S} | \phi(X_t) = 1) + P(\phi(X_{t+h}) = 0, X_{t+h} \in \overline{\mathcal{S}} | \phi(X_t) = 1), \\ &= P(\phi(X_{t+h}) = 0 | \phi(X_t) = 1, X_{t+h} \in \mathcal{S})P(X_{t+h} \in \mathcal{S} | \phi(X_t) = 1) + 0, \end{aligned}$$

As $\{\phi(X_t) = 1\} \subset \{X_{t+h} \in \mathcal{S}\}$, we get the result:

$$P(\phi(X_{t+h}) = 0 | \phi(X_t) = 1) = (k^-h + o(h)) \times 1.$$

Appendix B: Supplementary evaluation on a second crossbow-shaped cell

In this section, we provide another simulation controlled by the parameters given in Table 6. The results are shown in Figure 12 and the quantitative results are reported in Tables 7 and 8 (see Section 6.1 for comments).

Table 6. FLUOSIM parameters to simulate the second crossbow-shaped cell.

Type	Parameter	Value
Biophysics parameters	σ_S^2	$0.2 \mu m^2.s^{-1}$
	σ_T^2	$0.1 \mu m^2.s^{-1}$
	k^+	$10 sec^{-1}$
	k^-	$1 sec^{-1}$
Microscopy parameters	Δt	$0.02 sec$
	Δx	$0.160 \mu m$

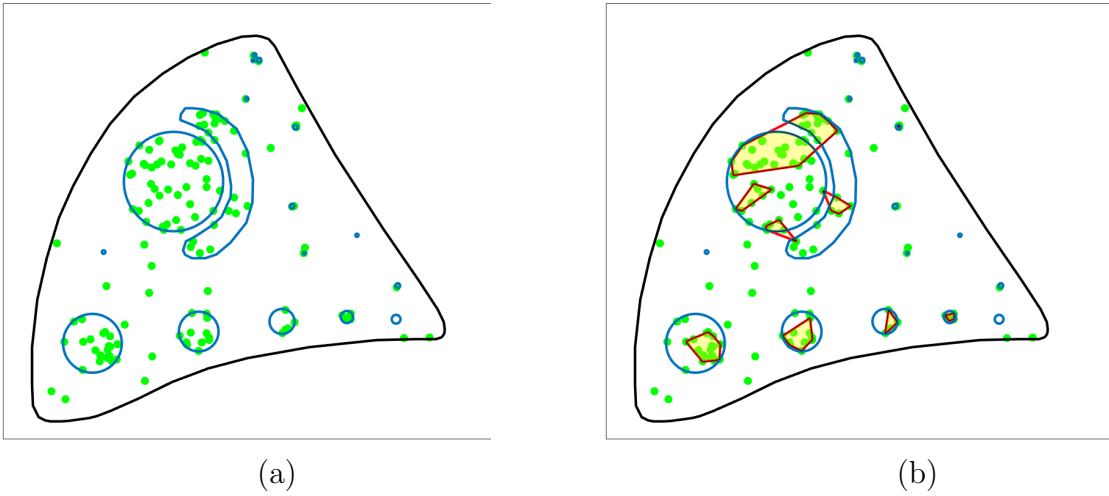


Figure 12. Results of our procedure on the second simulated crossbow-shaped; (a) representative points of the trajectories detected as subdiffusive by the test procedure of [Briane et al., 2018] at level 5%; (b) trapping regions (yellow background, red boundaries) estimated with the convex hull from the clusters detected by DBSCAN. The true trapping regions are depicted in blue.

Table 7. Evaluation of the procedure on the second simulated crossbow-shaped cell.

	True regions									
Region index (see Fig. 8)	1	2	3	4	5	6	7-14	15	16-17	
Number of overlapping clusters	4	3	1	1	1	1	0	1	0	
Maximum overlap (%)	31	17	24	43	25	34	0	7	0	

Table 8. Proportion of subdiffusive trajectories inside the detected regions from the crossbow-shaped cell simulation (2) to be compared ($k^+/(k^+ + k^-) = 0.9091$).

Cluster index	1	2	3	4	5	6	7	8
Proportion (%)	42.0	19.0	5.0	15.0	8.5	11.0	11.0	8.0

References

- A. Baddeley, I. Bárány, and R. Schneider. Spatial point processes and their applications. *Lecture Notes in Mathematics*, Springer-Verlag, 1892:1, 2007.
- H. Berry and H. Chaté. Anomalous diffusion due to hindering by mobile obstacles undergoing brownian motion or ornstein-ulhenbeck processes. *Physical Review E*, 89(2):022708, 2014.
- P. Brémaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer Science & Business Media, 2013.
- V. Briane, C. Kervrann, and M. Vimond. Statistical analysis of particle trajectories in living cells. *Physical Review E*, 97(6):062121, 2018.
- V. Briane, M. Vimond, and C. Kervrann. An overview of diffusion models for intracellular dynamics analysis. *Briefings in Bioinformatics*, 2019a.
- V. Briane, M. Vimond, C. Valades-Cruz, A. Salomon, C. Wunder, and C. Kervrann. A sequential algorithm to detect diffusion switching along intracellular particle trajectories. *Bioinformatics*, 2019b.
- F. Cao, J. Delon, A. Desolneux, P. Musé, and F. Sur. A unified framework for detecting groups and application to shape recognition. *Journal of Mathematical Imaging and Vision*, 27(2): 91–119, 2007.
- N. Chenouard, I. Smal, F. De Chaumont, M. Maška, I. F. Sbalzarini, Y. Gong, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter, et al. Objective comparison of particle tracking methods. *Nature Methods*, 11(3):281, 2014.
- N. Chenouard et al. Multiple hypothesis tracking for cluttered biological image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2736–3750, 2013.
- M. Daszykowski, B. Walczak, and D. Massart. Looking for natural patterns in data: Part 1. density-based approach. *Chemometrics and Intelligent Laboratory Systems*, 56(2):83–92, 2001.
- A. Desolneux, L. Moisan, and J.-M. Morel. Maximal meaningful events and applications to image analysis. *Annals of Statistics*, pages 1822–1851, 2003.
- H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983.
- M. El Beheiry, S. Türkcan, M. U. Richly, A. Triller, A. Alexandrou, M. Dahan, and J.-B. Masson. A primer on the bayesian approach to high-density single-molecule trajectories analysis. *Biophysical Journal*, 110(6):1209–1215, 2016.
- M. Ester, H.-P. Kriegel, J. Sander, and e. a. Xu, Xiaowei. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Int. Conf. Knowledge Discovery and Data Mining (KDD'96)*, pages 226–231, Portland, Oregon, USA, 1996.
- J. H. Friedman, F. Baskett, and L. J. Shustek. An algorithm for finding nearest neighbors. *IEEE Transactions on Computers*, 100(10):1000–1006, 1975.
- D. Fusco, N. Accornero, B. Lavoie, S. M. Shenoy, J.-M. Blanchard, R. H. Singer, and E. Bertrand. Single mRNA molecules demonstrate probabilistic movement in living mammalian cells. *Current Biology*, 13(2):161–167, 2003.
- N. Hoze, D. Nair, E. Hosity, C. Sieben, S. Manley, A. Herrmann, J.-B. Sibarita, D. Choquet, and D. Holcman. Heterogeneity of ampa receptor trafficking and molecular interactions revealed by superresolution analysis of live cell imaging. *Proceedings of the National Academy of Sciences*, 109(42):17052–17057, 2012.

- G. Hummer. Position-dependent diffusion coefficients and free energies from bayesian analysis of equilibrium and replica molecular dynamics simulations. *New Journal of Physics*, 7(1):34, 2005.
- M. Lagardère, I. Chamma, and O. Thoumine. FLUOSIM: Simulator of single molecule dynamics for fluorescence live-cell and super-resolution imaging. In *Quantitative BioImaging (QBI 2019)*, Rennes, France, 2019.
- J.-B. Masson, P. Dionne, C. Salvatico, M. Renner, C. G. Specht, A. Triller, and M. Dahan. Mapping the energy and diffusion landscapes of membrane proteins at the cell surface using high-density single-molecule imaging and bayesian inference: application to the multiscale dynamics of glycine receptors in the neuronal membrane. *Biophysical Journal*, 106(1):74–83, 2014.
- R. Metzler and J. Klafter. The random walk’s guide to anomalous diffusion: a fractional dynamics approach. *Physics Reports*, 339(1):1–77, 2000.
- N. Monnier, Z. Barry, H. Y. Park, K.-C. Su, Z. Katz, B. P. English, A. Dey, K. Pan, I. M. Cheeseman, R. H. Singer, et al. Inferring transient particle transport dynamics in live cells. *Nature Methods*, 12(9):838–840, 2015.
- R. G. Pinsky. Asymptotics of the principal eigenvalue and expected hitting time for positive recurrent elliptic operators in a domain with a small puncture. *Journal of Functional Analysis*, 200(1):177–197, 2003.
- T. Pécot, Z. Liu, J. Boulanger, J. Salamero, and C. Kervrann. A quantitative approach for analyzing the spatio-temporal distribution of 3d intracellular events in fluorescence microscopy. *Elife*, (7):e32311, 2018.
- H. Qian, M. P. Sheetz, and E. L. Elson. Single particle tracking. analysis of diffusion and flow in two-dimensional systems. *Biophysical Journal*, 60(4):910, 1991.
- A. Salomon and C. Kervrann. Precise mapping of intracellular diffusion and drift from SPT data analysis. In *Quantitative BioImaging (QBI 2020)*, Oxford, UK, 2020.
- M. J. Saxton. Anomalous diffusion due to obstacles: a Monte Carlo study. *Biophysical Journal*, 66(2 Pt 1):394, 1994.
- M. J. Saxton and K. Jacobson. Single-particle tracking: applications to membrane dynamics. *Annual Review of Biophysics and Biomolecular Structure*, 26(1):373–399, 1997.
- J. Schafer, N. Baetz, L. Lapierre, R. McRae, J. Roland, and J. Goldenring. Rab11-fip2 interaction with myo5b regulates movement of rab11a-containing recycling vesicles. *Traffic*, 15(3):292–308, 2014.
- S. Smith, C. Cianci, and R. Grima. Macromolecular crowding directs the motion of small molecules inside cells. *Journal of the Royal Society Interface*, 14(131):20170047, 2017.