Amortized inference of biomolecule dynamics

Hippolyte Verdier

THÈSE DE DOCTORAT
de l'Université Paris Cité

Spécialité : Physique

École doctorale n°564 : Physique en Île-de-France

réalisée

au Laboratoire Décision et Processus Bayésiens

sous la direction de
Jean-Baptiste MASSON
et Alhassan CASSE

présentée par

Hippolyte VERDIER

Sujet de la thèse :

Amortized inference of biomolecule dynamics

soutenue le mardi 4 octobre 2022

devant le jury composé de :

M.	BERRY Hugues,	DR INRIA	Rapporteur
M.	PRESSE Steve,	Prof. Université d'Arizona	Rapporteur
Mme.	ALEXANDROU Antigoni,	DR École polytechnique	Examinatrice
Mme.	ALLASSONNIERE Stéphanie,	Prof. Université Paris Cité	Examinatrice
Mme	EL KAROUI Meriem,	Prof. Université d'Edimbourg	Examinatrice
M.	FRANÇOIS Paul,	Prof. A. Université McGill	Examinateur
M.	LOUPE Gilles,	Prof. A. Université de Liège	Examinateur
M.	SAUER Markus,	Prof. Université de Würzburg	Invité
M.	CASSE Alhassan,	Chercheur, Sanofi	Invité
M.	MASSON Jean-Baptiste,	DR CNRS	Examinateur

Remerciements

I would first like to thank members of the jury for the attention that they paid to this work. I am especially thankful to the two reviewers, Hugues Berry and Steve Presse. I also take the opportunity to thank Raphaël Voituriez who, along with Hugues, was part of my thesis advisory committee.

J'aimerais ensuite remercier Jean-Baptiste, qui m'a guidé à travers ces trois années avec un calme des plus rassurants et une sollicitude qu'il sera difficile de ne pas regretter. Merci pour tes conseils, scientifiques ou non, ta disponibilité sans faille (coucou Eugénie et Daphné) et pour m'avoir fait confiance même lorsque je m'éloignais du modèle canonique. Interprétable[1], robuste, fiable et rapide, tu as toutes les qualités d'un bon estimateur. Je voudrais aussi remercier Christian et François, patients relecteurs et correcteurs, pour leur aide précieuse et leur attention tout au long de ces trois années.

Cette thèse n'aurait sûrement pas vu le jour sans le soutien de Sanofi, dont je voudrais ici remercier les collaborateurs en commençant par Souâd Naïmi qui m'a accueilli dans son équipe. Merci particulièrement à Alhassan Cassé, de m'avoir introduit aux usages de la maison et aidé à communiquer mes résultats. Merci aussi à Nicolas Bodier, pour le temps passé à m'expliquer et mettre au point ces manips si capricieuses.

J'aimerais aussi remercier Christian Specht avec qui ce fut un plaisir de collaborer. Merci pour tes explications sur les acquisitions et l'alpha-synucléine, ainsi que pour ta patience. Merci aussi à Mickaël Lelek pour ses avis éclairés sur la SMLM et à Juliette Griffié pour l'organisation de ce mini-symposium à MifoBio. Au chapitre du soutien technique, je tiens à remercier la DSI et le hub de Pasteur, et plus particulièrement Thomas Ménard, Anthony Douin et Bryan Brancotte, sans l'aide de qui Tidiane et moi n'aurions certainement pas pu mettre sur pied une application comme Tracktor. Merci d'être si joignables et patients, ainsi que d'avoir su nous aider à comprendre un petit peu la machinerie que cache le site web déployé.

Voici maintenant pour les férus de larves de mouches, de trajectoires et de *foeta* vus en réalité virtuelle – en un mot : de décisions-et-processus-Bayésiens. Merci à Chloé (qui a même pris le temps de relire ce manuscrit) et Alexandre d'avoir su répondre, ne serait-ce que par votre présence, au besoin d'humanité que suscitent de trop longs tête-à-tête avec un ordinateur refusant de coopérer (et au besoin de partager, le cas échéant, les victoires obtenues sur celui-ci). Merci Charlotte pour tes talents d'animatrice du labo et Alexis pour ton inattaquable bonne humeur. Enfin, la page des prénoms en "Alex-" de l'annuaire du labo était un peu vide avant l'arrivée de M. Barbier Chebbah ; c'était un plaisir de faire petit à petit ta connaissance au cours de cette dernière année de thèse. Longue vie au G5 !

Merci à Julienne et Gustave, camarades de Pasteur : vos péripéties de paillasse racontées au 25 m'aidaient à relativiser mes bugs et votre enthousiasme ravivait le mien. Merci aux copains doctorants avec qui l'on pouvait discuter boutique et notamment à Victor, organisateur de dîners savants, Paul et Vincent. Mais il faut de tout pour faire un monde, et certainement pas que des docteurs : merci aux autres copains qui ont assuré l'équilibre. Merci aussi à Anne-Marie et René, Marie-Elisabeth et Félix, pour ces déjeuners qui changeaient les idées.

Si l'on remonte un peu plus loin, j'aimerais remercier aussi les professeurs qui, à différents âges, m'ont fait aimer les sciences : M. Benthami au lycée, Mme. Gillette en prépa, puis à

[1] La littérature n'est pas claire à ce sujet

l'X les cours de M. Echard, qui m'ont réconcilié avec la biologie, la physique de M. Quéré, toute en schémas, et les petites classes de M. Mallick.

Merci enfin à ma famille et à Emmanuelle, sans qui tout ça n'aurait sans doute rien été du tout.

Résumé en français

Le mouvement des biomolécules (protéines, acides nucléiques, complexes divers...) est par nature aléatoire : l'agitation thermique du milieu cellulaire les soumet à des perturbations sans cesse fluctuantes. Mais ce mouvement, bien qu'aléatoire, n'est que rarement libre : il est bien souvent contraint par son environnement. En effet, les interactions diverses auxquelles prennent part les biomolécules, ainsi que l'encombrement du cytoplasme ou de la membrane, qui tantôt confinent et tantôt guident le mouvement, sont deux facteurs susceptibles d'avoir une influence majeure sur leur dynamique. Caractériser cette dynamique permet donc, indirectement, de mieux connaître le milieu dans lequel évoluent les biomolécules, de comprendre et mesurer les interactions qui y régissent leur déplacement.

Observation de trajectoires de biomolécules en milieu cellulaire

Plusieurs technologies présentées au chapitre I permettent de sonder les propriétés de la diffusion en milieu cellulaire, soit à l'échelle d'un ensemble de biomolécules, soit en suivant celles-ci séparément les unes des autres. Cette thèse se concentre sur l'analyse de mesures obtenues via ce second type d'observations. Plus précisément, elle se consacre principalement à l'étude de trajectoires observées par la technologie de microscopie de localisation par molécule unique (dont le sigle anglais est "SMLM"). Cette méthode permet d'observer le mouvement de biomolécules, trajectoire par trajectoire, avec une précision de quelques dizaines de nanomètres et une résolution temporelle de l'ordre de la dizaine de millisecondes, laissant augurer une caractérisation précise de la topographie cellulaire et des propriétés dynamiques des protéines dans chacune des différentes organelles. Néanmoins, l'inférence, à partir des trajectoires, des propriétés du processus génératif sous-jacent, est une gageure. Un tour d'horizon des méthodes d'analyses développées précédemment est donné au chapitre II. Les défis rencontrés peuvent être généralement répartis en deux ordres : ils proviennent soit de limitations techniques dues au protocole expérimental ou à la technologie d'imagerie utilisée, qui restreignent la précision ou la quantité des mesures, soit de la méconnaissance des phénomènes observés et de la diversité de ceux-ci, qui empêchent que soient faites à leur sujet des hypothèses trop fortes. Les méthodes présentées dans cette thèse sont particulièrement adaptées aux cas où ces deux types de difficultés s'entremêlent.

Inférence amortie de propriétés de marches aléatoires

Je présente au chapitre III une méthode d'inférence des propriétés des marches aléatoires, qui se place dans le cadre de l'inférence amortie basée sur des simulations. Une telle in-

férence est dite "amortie" parce qu'elle ne peut être menée à bien qu'après une longue phase d'entraînement d'un réseau de neurones – entraînement que l'on amortit au fur et à mesure des inférences réalisées par la suite. Cette méthode tire parti de la flexibilité offerte par l'apprentissage profond, permettant d'obtenir une meilleure précision que des estimateurs analytiques et de mieux prendre en compte les biais induits par l'incertitude de localisation et la courte longueur des trajectoires telles que celles observées par SMLM. Plus précisément, j'ai adapté à l'analyse de trajectoire une architecture de réseau de neurones sur graphes, en représentant justement sous forme de graphe le mouvement de chaque particule. Cette représentation permet de prendre naturellement en compte l'invariance du problème par rotation de la trajectoire ainsi que les dépendances temporelles de la dynamique à toutes les échelles. Elle a de plus l'avantage, par rapport à d'autres architectures d'apprentissage profond, de comporter un nombre relativement limité de paramètres à optimiser. Ce réseau de neurone est entraîné à inférer des propriétés physiques des marches aléatoires, sur des trajectoires générées numériquement à l'aide d'une variété de modèles présentant des types de dynamique différents.

Plus précisément, les variables inférées sont l'exposant de diffusion anormale α ainsi que le modèle de marche aléatoire dont chaque trajectoire est une réalisation. L'exposant de diffusion caractérise l'évolution en loi de puissance du carré de la distance à l'origine au cours du temps : $\langle (r(t) - r(0))^2 \rangle \propto t^\alpha$. L'estimation de cette quantité présente deux difficultés principales : d'abord, c'est une moyenne sur un ensemble de réalisations indépendantes d'un même processus ; ensuite, la loi de puissance est asymptotique, valable à temps long, alors même que les mesures dont on dispose expérimentalement sont souvent courtes et n'atteignent pas un tel régime (il est rare que leur longueur dépasse une dizaine de points). On comprend donc aisément que la détermination de l'exposant α à partir d'une seule trajectoire, courte de surcroît, n'est, en général, pas chose aisée – et jamais dénué d'ambiguïté. De même, la vraisemblance de la plupart des modèles de marches aléatoires n'admet pas, à ce jour, de formule analytique, ce qui complique la comparaison rigoureuse de plusieurs modèles à l'aide de méthodes exactes. Par ailleurs, quand bien même des quantités ad-hoc se révéleraient capable de discriminer entre l'un ou l'autre modèle, il est rare que le comportement de ces estimateurs soit stable sur des trajectoires courtes ou bruitées. Pour toutes ces raisons, on peut s'attendre à ce qu'un réseau de neurones convenablement entraîné sur un grand nombre de trajectoires, à même de détecter un large spectre d'interdépendances temporelles et spatiales, atteigne de meilleures performances que des estimateurs conventionnels.

Nous montrons qu'en effet, le réseau est capable d'inférer α et de classifier les trajectoires de manière très satisfaisante, et ce même lorsque les positions sont bruitées et les trajectoires courtes. Cette robustesse à l'incertitude de localisation est un des avantages de l'inférence basée sur des simulations : il suffit d'ajouter du bruit sur les trajectoires présentées au réseau pendant l'entraînement pour qu'il "apprenne" à fournir une mesure fiable de leur propriétés. Cette bonne performance révèle la richesse de la représentation des trajectoires apprise par le réseau, en amont de la sortie de ce dernier. Nous nous attachons en effet à montrer que le vecteur *latent* par lequel le réseau représente chaque trajectoire et à partir duquel il fournit une estimation de ses propriétés, capture un grand nombre d'aspects physiques intéressants (coefficient de diffusion, nature du modèle génératif...). Notamment, le vecteur décrivant une trajectoire issue d'un mélange de deux modèles de marche aléatoires sera souvent situé entre les deux vecteurs décrivant des trajectoires issues de chacun de ces modèles (alors même que de telles trajectoires "composites" n'ont jamais été présentées au réseau pendant

l'entraînement). Les modèles génératifs des trajectoires observées dans les cellules étant vraisemblablement composites (en raison des hétérogénéités de l'environnement cellulaire), la robustesse de la représentation latente des trajectoires par le réseau est précieuse. Elle sera exploitée plus avant au chapitre V de cette thèse.

Inférence variationnelle et comparaison à l'optimum

Avant de tirer parti de la représentation latente de trajectoires courtes, nous avons cherché à comparer la performance atteinte par notre méthode d'inférence à celle qu'atteindrait un estimateur optimal. Afin de pouvoir déterminer la performance optimale, je me restreins au chapitre IV à des trajectoires de mouvement Brownien fractionnaire (fBM), modèle dont la vraisemblance s'exprime de façon analytique. À l'aide de l'inégalité de Cramér-Rao, nous pouvons donc estimer la variance minimale d'un estimateur non biaisé de l'exposant α. Nous proposons pour l'inférence de α un réseau de neurones basé sur l'architecture sur graphe présentée plus haut, auquel nous couplons un réseau réalisant une transformation inversible et paramétrable entre un espace doté d'une loi de probabilité Gaussienne et l'espace des paramètres à inférer. Ceci nous permet d'inférer une distribution a posteriori du paramètre α avec, une fois passée la phase d'entraînement, une complexité algorithmique croissant linéairement avec la longueur de la trajectoire. La complexité de l'inférence classique, basée sur la vraisemblance, estimée en inversant la matrice de covariance du processus, est quadratique en nombre de pas. Nous montrons que la performance atteinte par notre méthode est proche de l'optimum prédit par la formule de Cramér-Rao (qu'atteint l'inférence classique), et ce sur deux ordres de grandeur de longueurs de trajectoires.

En outre, nous étudions la capacité d'une telle méthode d'inférence à mesurer un temps de corrélation fini, au-delà duquel les dépendances temporelles s'estompent exponentiellement. Ce phénomène, qui tend à faire diverger le mouvement du cadre théorique du fBM, est attendu en milieu cellulaire (où il est peu probable que le mouvement soit doté d'une mémoire infinie) mais très rarement pris en compte du fait de la difficulté d'estimer ce temps caractéristique. Nous démontrons ainsi la capacité de méthodes d'inférence amortie à inférer des propriétés encodées dans des dépendances à longue portée avec un gain substantiel de complexité algorithmique.

Comparaison statistique d'ensembles de trajectoires

Jusqu'ici, nous ne nous sommes intéressés qu'à un ensemble restreint de quantités décrivant les trajectoires, que nous avons estimées à l'aide de réseaux de neurones. Mais lorsque le processus génératif est inconnu, se restreindre à ces quelques variables revient à faire une hypothèse forte selon laquelle les autres degrés de liberté du système sont sans importance. Alors que les performances atteintes par les réseaux de neurones suggèrent que ceux-ci apprennent une représentation des trajectoires riche en information, nous proposons au chapitre V d'utiliser directement cette représentation pour décrire les trajectoires. Afin de comparer des ensembles de trajectoires, nous proposons d'utiliser la différence moyenne maximale (en anglais: "maximum mean discrepancy") et le test par permutation qui y est associé. Ces deux étapes (caractérisation puis test statistique) sont indépendantes l'une de l'autre et peuvent être chacune remplacées par des alternatives selon que l'on veuille se concentrer sur un

aspect particulier des trajectoires ou tester une forme spécifique de variabilité. Les options présentées ici pour chacune de ces deux étapes ont l'avantage, pour la caractérisation de couvrir une grande variété de dynamiques, et pour le test de fournir une interprétation directe des différences relevées.

L'architecture du réseau est la même qu'au chapitre III et il est entraîné sur la même tâche, à ceci près que l'on se concentre cette fois-ci sur des trajectoires courtes et assez bruitées, afin d'être au plus près de celles observées expérimentalement. Nous validons d'abord sur des données expérimentales la pertinence de la méthode et sa sensibilité, puis nous validons sa robustesse en l'appliquant à l'étude de la dynamique de l'α-synuclein. La méthode permet d'identifier des différences entre chacune des conditions étudiées (contrôle, avec ajout de sodium dans le milieu, puis fixé). Elle permet aussi de vérifier que la différence détectée n'est pas due à une expérience aberrante, mais qu'elle se retrouve bien dans chacune des répliques. De façon générale, cette approche de caractérisation du mouvement à l'échelle de la trajectoire permet de tester des différences entre des ensembles définis de plusieurs manières (par champ d'acquisition, par neurone, par condition...), ce qui permet de sonder précisément l'hétérogénéité des données recueillies. Ce qui es perdu en clarté d'interprétation par le recours à la description par des vecteurs latents est *in fine* compensé par un gain en diversité des critères pris en compte.

Outils logiciels pour l'analyse

Enfin, je présente au chapitre VI des outils développés afin que les analyses présentées dans cette thèse soient accessibles au plus grand nombre. Le premier outil, nommé Palmari, est un module du logiciel de visualisation d'image Napari, et permet de traiter le signal recueilli par la caméra du microscope, lors d'une acquisition de SMLM, afin de détecter les localisations puis de recueillir les trajectoires. Ce processus comporte un certain nombre de paramètres dont il est parfois difficile de déceler les interdépendances. Grâce à cet outil, il est possible d'ajuster ceux-ci tout en ayant un aperçu visuel de leur effet sur les trajectoires finalement recueillies. Une fois le protocole de traitement affiné, l'outil permet, via une interface Python, de le lancer sur toute une série d'acquisition correspondant à une même expérience.

Le deuxième outil présenté est une plateforme web (https://tracktor.pasteur.cloud), qui permet à n'importe quel utilisateur d'effectuer les analyses présentées au chapitre V sur les trajectoires qu'il y dépose. Ainsi, aucune installation n'est nécessaire et aucun code informatique ne doit être écrit par l'utilisateur, simplifiant l'adoption de la méthode par le plus grand nombre. Les résultats peuvent être visualisés sur la plateforme et téléchargés par les utilisateurs.

* * *

Cette thèse introduit donc une nouvelle méthode d'analyse des trajectoires de biomolécules, tirant partie des possibilités offertes par l'apprentissage profond et l'inférence amortie, et proposant une nouvelle caractérisation reposant sur la représentation des trajectoires apprise par un réseau de neurones. La méthode a été éprouvée sur des données simulées et sur des données expérimentales, et des outils logiciels sont proposés pour qu'elle puisse être utilisée facilement à l'avenir.

Contents

List of Figures

Acronyms

API Application Programming Interface.

ATTM Annealed transit time.

CTRW Continuous time random walk.

fBM Fractional Brownian motion.

FCS Fluorescence correlation scpectroscopy.

FPT First passage time.

FRAP Fluorescence recovery after photobleaching.

GNN Graph neural network.

GRATIN Graphs on trajectories for inference.

HMM Hidden Markov model.

HTML Hypertext Markup Language.

HTTP Hypertext Transfer Protocol.

LW Levy walk.

MDS Multi-dimensional scaling.

MLP Multi-layer perceptron.

MMD Maximum mean discrepancy.

MME Mean maximum excursion.

OU Ornstein-Uhlenbeck process.

PSF Point spread function.

RNN Recurrent neural network.

SBI Simulation-based inference.

sBM Scaled Brownian motion.

SPT Single particle tracking.

STED Stimulated emission depletion.

TIRF Total internal reflection fluorescence.

TRamWAy The random walk analyzer.

UMAP Uniform manifold approximation and projection.

Introduction

Context

These last fifteen years, cellular biology has been jostled by two rising trends. Space-wise, developments in super-resolution imaging have pushed biologists to investigate the nanoscale interactions occurring in cells, ruled by thermal agitation and processes combining several time scales. Besides, due to exponential progress in computing and digitisation of measurement devices, the amount of data collected by biologists – whether images, genotypes or physiological measurements – has burst. Both these trends lead to an increasing role of statistical analysis in cellular biology.

A challenge remains in unifying the modelling of cellular processes at the manometer scale and the statistical analysis required to make sense of the experimental data. Modelling detailed aspects of the dynamics properties often leads to new challenges in statistical analysis of experiments. Conversely, large scale data recording coupled to advanced statistical analysis can reveal complex heterogeneity or phenotypic diversities that may be difficult to include in models.

Overview of the approach

Scientific question

My PhD project focused on the general question of characterizing the dynamics of biomolecules in complex environments.

Although the asymptotic behaviour of many canonical random walk models is well characterized, this knowledge cannot be used as such to probe properties of experimental trajectories of biomolecules, both because of their intrinsically composite nature, which distinguish them from canonical models, and because of the constraints posed by imaging methods, limiting the precision, the length and the number of observed trajectories.

While most random walk analysis methods focused on measuring physical parameters (the diffusion coefficient, the anomalous diffusion exponent, an eventual drift and sometimes the underlying generative model), these quantities are often ambiguous or ill-defined when applied to short trajectories. For instance, the anomalous exponent being an average over an ensemble of realisations, a single trajectory generated by a non-ergodic process only gives access to a part of the information needed to estimate its value. Moreover, not all aspects of random walk dynamics are covered by conventional physical parameters: being the exponent of a long-time power law, the anomalous diffusion exponent notably discards

short-terms dynamics. Likewise, the cellular environment – which sets the motion dynamics of biomolecules – has an important spatial heterogeneity, with patterns of various scales. The assumption that a trajectory evolves in this setting according to a constant generative model is thus legitimately questioned.

How then to quantitatively characterize experimental recordings of biomolecule dynamics ? New features might be relevant to describe short recordings of diffusing particles with more flexibility than canonical estimates. It is nonetheless desirable that trajectories' description be both compatible with statistical tests (for comparison of biological conditions, replicates or cell locations) and interpretable (to facilitate controls and inform biological hypotheses). Although this concern might seem secondary when large amounts of observations are available and can be aggregated, analyses led at the single-trajectory scale remain relevant insofar as they allow one to probe the diversity of dynamics existing within a given ensemble.

Approach

In order to provide a new characterization of biomolecule trajectories, I developed simulation-based inference schemes. More precisely, I trained a neural network to infer well known physical properties of random walks generated numerically using of a variety of canonical generative models and mimicking the experimental conditions (in terms of localization noise and trajectory length). Nevertheless, the originality of the characterization was not to rely on these inferred properties, but on the upstream *latent* representation of walks learnt by the network. The dimension of these latent vectors is higher than the number of estimated properties, thus providing a richer description of random walks.

To extract features from trajectories with a neural network, I developed a graph neural network (GNN) architecture. GNNs were first introduced to handle geometric data (point clouds) and networks of interactions. I adapted them to process trajectories by treating trajectories as graphs, considering positions as nodes and drawing edges corresponding to different scales of time-dependencies. They have the advantage of requiring a relatively low number of parameters and of providing simple means to match symmetries of the problem as well as to incorporate long-range memory effects.

In order to assess the capability of this architecture to accurately capture random walk properties, we performed variational inference of the anomalous diffusion exponent of fractional Brownian Motion. Using the Cramér-Rao bound, we could estimate a lower bound of the variance achievable by an unbiased estimator of this exponent, and observed that the network's performance was close to optimal on a wide range of lengths. This is noteworthy given that the algorithmic complexity of the estimation provided by the network is linear, while the maximum likelihood estimation is at least quadratic. Furthermore, we showed that the architecture was also capable of capturing information about the decorrelation length, something which is rarely addressed and especially hard to measure with conventional estimators on single trajectories. This amortized inference, yielding estimates of the posterior distribution of parameters without requiring expensive Monte Carlo sampling, illustrates the possibility of performing Bayesian inference at low computational cost even on processes without likelihood.

Latent vectors, as such, are not directly interpretable. Nonetheless, they contain a comprehensive description of trajectories and lie in a space whose geometry is smoother than the one which vectors composed of classical estimators would form. Maximum mean discrepancy,

a kernel-based method, provides a mean to perform statistical tests in such multidimensional spaces. We applied the method to compare trajectories obtained under different conditions and from different biological systems *via* their latent representations and assess the statistical significance of the eventual differences according to various null hypotheses. We used these comparisons with various levels of granularity to probe the heterogeneity that exist across individual synapses, cells, replicates of a same experiment, and biological conditions. The comparison also pinpoints the most salient differences between two sets of trajectories, thus guiding the interpretation of findings.

Finally, given the substantial number of steps involved in the analysis of single molecule movies, and the implication that each step has on downstream results, I thought useful to provide the community with a tool allowing one to fine-tune and compare processing pipelines (with visual feedback) as well as to smoothly run an entire pipeline on batches of experimental observations, from raw images recorded by the camera to comparisons of sets of trajectories. Moreover, as the interest of the approach we developed to compare sets of trajectories does not restrict to one experimental modality, I initiated the development of an online platform allowing researchers to perform the analysis presented in this thesis on their trajectories, without writing any code.

Structure of the thesis

Chapter I briefly introduces random walks and gives orders of magnitude of the cell's geography and physics. Besides, experimental techniques allowing to probe diffusion properties in cells are presented, with the specifics and comparative advantage of each.

Chapter II presents the inference formalism and its application to the analysis of biomolecule random walks. An overview of methods is given, along with examples of previous work in which they are used.

Chapter III introduces in more detail the simulation-based inference procedure, as well as the graph neural network architecture and the resulting latent space encoding of random walk properties learnt by the network.

In chapter IV, I illustrate the performance of the network by using it to infer the anomalous exponent of fractional Brownian motion, a task for which a theoretical lower bound of the variance achievable by an unbiased estimator can be derived. Besides, by merging the graph neural network with a differentiable density estimator, I propose a variational inference scheme with linear complexity.

I introduce in chapter V the statistical tests and tools used to assess the significance of differences between sets of observed trajectories. We used them to shed light on the dynamics in cultured neurons of α-synuclein, a protein involved in Parkinson disease.

Finally, I introduce in chapter VI the main software tools that I released to make my work accessible to the scientific community.

I – Biomolecule random walks in cells

Contents

We designate by "biomolecule" any molecule small or large that contributes to the functioning of the cell. These are elementary components involved in essential processes such as information transmission and storage (using chains of nucleic acids), sensing of the external environment (*via* transmembrane proteins) or which simply constitute the layer separating the interior of the cell from the outside (lipids). Composed of a few dozens atoms for the smallest of them and of billions for the largest, both their chemical composition and their geometry are subject to variations throughout their existence.

Biomolecules are subject to complex and varying free energy landscapes stemming from fluctuating forces, leading to directional motion or changes in conformation – which in turn might affect the forces exerted on proteins by their environment. Surrounded by small molecules (water, lipids etc), they undergo a constant flow of collisions originating from all directions. The combined effect of these collisions constitutes a key driver of a protein's motion. Averaged on long time , these collisions cancel out, but on short time scales, their average effect might exert a significant perturbation on the protein which, depending on the constraints set by its environment, might be put into motion and pushed to another location with new physical constraints.

Hence, a biomolecule's dynamic is revealing of its interaction with its environment. Is the motion fast or slow ? Is it constrained in some way, does it freely explore the surrounding space, is it moved by a motor or blocked by an obstacle ?

Significant theoretical work has been done since the early 20^{th} century about the characterization of the type of motion designated by the term *random walk*, which broadly encapsulates all dynamics driven by random fluctuations. We first introduce very briefly

basic concepts associated to random walks I.1, before detailing the physical interactions of a protein with its environments more thoroughly in I.2. Finally, we introduce the different imaging modalities allowing observation of diffusion phenomenons in cells I.3.

I.1 Random walks: Brownian motion and anomalous diffusion

The adjective *Brownian* originates from the following remark made by the Scottish botanist Robert Brown about the motion of grains of pollens of *Clarckia pulchella* [1].

> While examining the form of these particles immersed in water, I observed many of them very evidently in motion; their motion consisting not only of a change of place in the fluid, manifested by alterations in their relative positions, but also not unfrequently of a change of form in the particle itself (...). These motions were such as to satisfy me, after frequently repeated observation, that they arose neither from currents in the fluid, nor from its gradual evaporation, but belonged to the particle itself.

This observation owed Brown a legacy that spans many scientific fields. Indeed, this kind of phenomena has later been observed in many domains of physical sciences as well as biology and finance. Brownian motion has become the elementary dynamics from which more advanced random walk models – of which a few will be presented in the following work – derive. In the following, we briefly recall the first progresses towards the understanding the physical phenomenons resulting in Brownian motion, made by Einstein and Perrin at the dawn of the twentieth century. We then introduce anomalous diffusion, and mention some quantities of interest to the study of random processes. Finally, we present the random walks models which will be used throughout this work.

I.1.1 From collisions to diffusion: Brownian motion

While other observations of chaotic motion were recorded later on in the nineteenth century, it was only in 1905 that Einstein mathematically bridged the gap between the microscopic and macroscopic scales of motion [2]. He derived the relation between the *diffusion* equation introduced by Adolf Fick in 1855 and the probability law governing the displacements of particles at short time scales. A few years after Einstein's analytic derivation, Jean Perrin experimentally validated the molecular nature of the Brownian motion of particles in suspension in liquids [3] by verifying relation between a particle's radius, its displacements, the viscosity of the fluid and its temperature. It is remarkable that Perrin's experiments were partly motivated by the observation of the self-similarity of the trajectories across scales: trajectories formed of positions recorded at different frequencies have similar shapes. Indeed, commenting on trajectories such as those shown in figure I.1, he wrote that each segment of these, if observed at a higher time resolution, would reveal a trajectory of complexity equivalent to that of the coarse grained trajectories.

In his seminal work on the dynamics of an object undergoing a steady but chaotic flow of fluctuations, Einstein postulated that there exist a time scale τ which is small compared to the interval between consecutive observations of a particle's location, and sufficiently large so that the motion of the particle between two consecutive intervals of duration τ

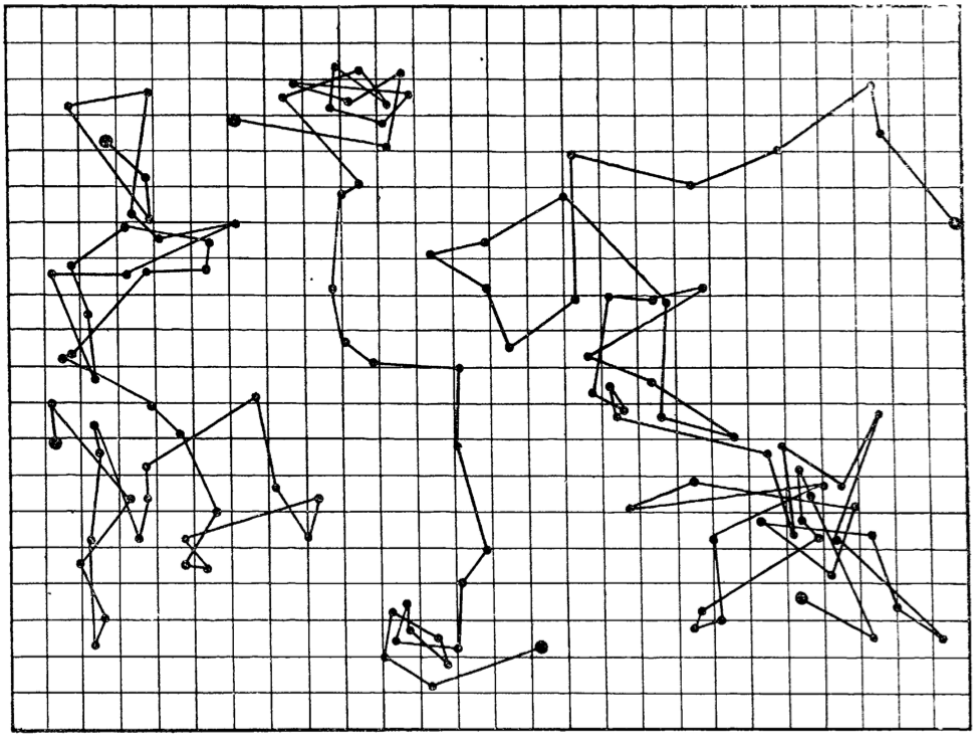

Figure I.1: Three trajectories of mastic spheres observed by Jean Perrin, taken from [3].

can be considered as independent processes. He also assumed that particles' motions are independent processes governed by the same statistical law.

The displacement Δ of a particle along one dimension during a time interval τ obeys the probability law $\phi(\Delta)$, which is assumed symmetric as no direction is favored. It is also assumed to vanish rapidly at long distances, such that its second moment is finite.

From these hypotheses, we obtain the two following relations relating the density of particles $f(x,t)$ and the probability law governing their microscopic motion:

$$D = \frac{1}{2\tau}\int_{-\infty}^{\infty} \Delta^2 \phi(\Delta) d\Delta$$

$$\frac{\partial f}{\partial t} = D\frac{\partial^2 f}{\partial x^2}.$$

It follows from the previous equations that the mean square displacement (MSD) along one dimension of particles starting from the origin is

$$\langle x(t)^2 \rangle = \overline{x(t)^2} = 2Dt$$

where $\langle\cdot\rangle$ denotes the average over particles (ensemble average), while $\bar{\cdot}$ denotes the average over disjoint time intervals originating from a same particle.

I.1.2 Anomalous diffusion

a) Introduction

Biology provides many examples of random walks whose MSD does not follow the linear trend: these walks are said to exhibit *anomalous* diffusion. Yet, anomalous diffusion was first described and studied in condensed matter physics, both to describe the motion of particles (in disordered media for instance) and the phase-space dynamics of high-dimensional systems (such as a spin glass) [4].

Biological anomalous motion are summarised under the criteria that their ensemble MSDs follow a power law of the form

$$\langle x(t)^2 \rangle = K_\alpha t^\alpha$$

where α is the anomalous diffusion exponent and K_α is a generalized diffusion constant that sets the scale of the process. The random walk is said to be subdiffusive when $\alpha < 1$, superdiffusive when $\alpha > 1$ and Brownian (i.e. non anomalous) when $\alpha = 1$.

Anomalous diffusion is associated to at least one of the following properties [5]:

1. random walkers or their environment exhibit spatially or temporally varying properties,
2. displacements are not statistically independent at any sufficiently small time scale,
3. displacements at small time scales exhibit anomalies (e.g. heavy-tailed distributions) that prevent the central limit theorem from applying.

b) Some statistical properties of random processes

The equality between time and ensemble average defines *ergodicity*, and Brownian motion is thus an *ergodic* process. For non-ergodic processes, single trajectories do not explore the entire phase space even in the limit of infinite time. In other terms, recording a single infinitely long trajectory of a non-ergodic process is not sufficient to observe the entire dynamics available to walkers of this process. Another aspect characterizing random walks is whether or not they exhibit *ageing*, that is, whether or not observables such as the ensemble MSD depend on the elapsed time between the initialisation of the system and the start of the measurement. Quantities independent of time are said to be *stationary*. More precisely, some of the random walks on which we will focus are *weakly* non-ergodic: they *could* possibly explore the entire phase space, but the probability of doing so is so low that it would require an infinite amount of time. Finally, a random walk is said to be self-similar if its properties are conserved by up- or down-sampling the walk.

c) Examples of anomalous random walks relevant to this work

Throughout this work, we will use and refer to multiple canonical models of anomalous random walks, illustrated in figure I.2 and of which we here describe the main properties.

- **Fractional Brownian motion** (fBM) [6, 7]. fBM is a Gaussian process characterised by long temporal correlations in the noise driving the process. It is generated by a Langevin equation [6] of the form $\frac{d\mathbf{r}(\mathbf{t})}{dt} = \sqrt{K_\alpha}\boldsymbol{\eta}(t)$, where $\boldsymbol{\eta}$ is a zero-mean Gaussian noise process with covariance structure $\langle \boldsymbol{\eta}(t_1)\boldsymbol{\eta}(t_2)\rangle = \alpha(\alpha-1)|t_1-t_2|^{\alpha-2}$. fBM is a

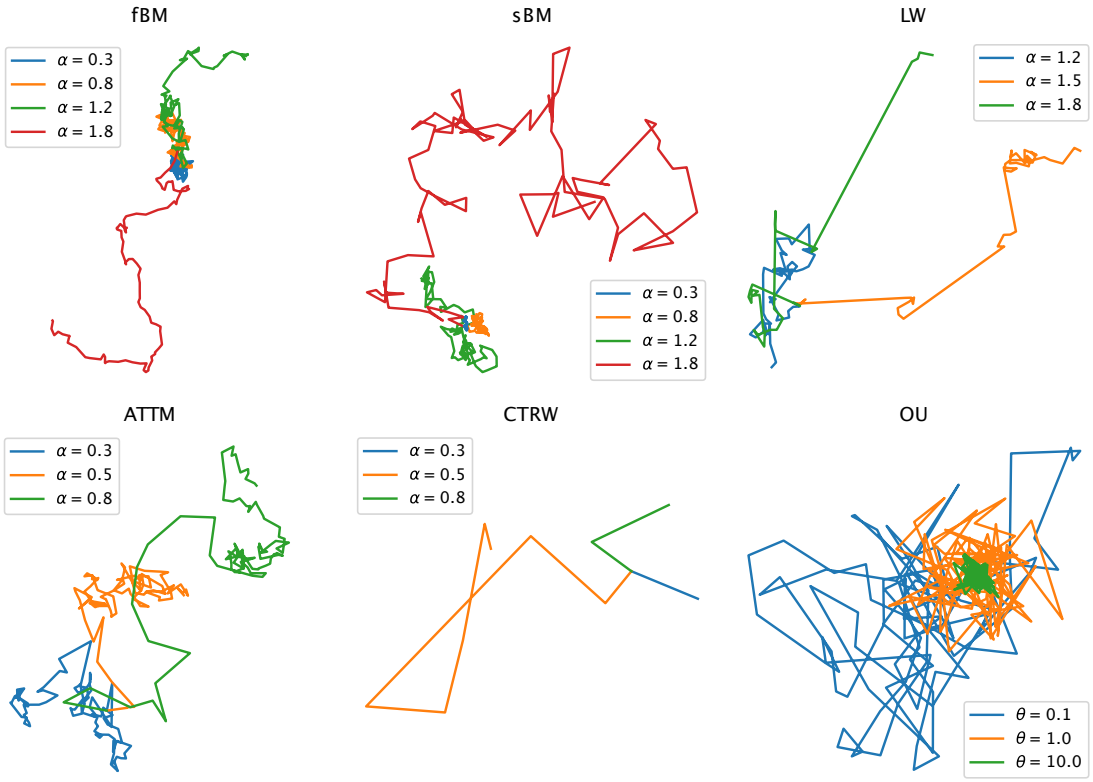

Figure I.2: Example of numerically generated anomalous random walks, realisations of the models studied in this thesis. The diffusion coefficient is constant.

self-similar Gaussian process with stationary increments [6, 7]. Its associated noise is anti-persistent and negatively correlated in the subdiffusion regime but persistent and positively correlated in the superdiffusion regime. As such, this random walk model displays anomalous property no. 2 as defined above. It is stationary and ergodic [8], and its likelihood is analytically tractable. The tractability of its likelihood makes it a model of choice to benchmark the performance of various inference methods. We use it as such in chapter IV.

- **Scaled Brownian motion (sBM)** [9, 10]. sBM is generated by a Langevin equation [6] with a time-dependent diffusion coefficient of the form $K(t) = \alpha K_\alpha t^{\alpha-1}$ and driven by an uncorrelated (white) noise. It can generate both subdiffusive and superdiffusive processes. sBM displays anomalous property no. 1, it is weakly non-ergodic [11] and has the same marginal probability density for the time-evolution of the walker's position as the fBM [12], but a different autocorrelation structure.

- **Continuous time random walk (CTRW)** [13]. In the CTRW the random walker's motion is generated by a renewal process consisting of discrete jumps with a given waiting time distribution between jumps, $\psi(\tau)$, and a distance distribution of jump lengths, $f(\mathbf{\Delta})$. Continuous time random walks can be either sub or super-diffusive. The most studied case is the sub-diffusive one, where f has finite first and second

moments, and where $\psi(\tau) \propto \tau^{-\alpha-1}$, with $0 < \alpha < 1$, such that the mean waiting time is infinite. If the mean waiting time is finite, superdiffusion occurs either when the distribution of displacements is heavy-tailed (has diverging variance), or when there is a coupling between step lengths and waiting times. The definition of the CTRW used here corresponds to a physical model associated to an annealed environment as waiting times are randomly drawn at each jump independently of its location [14]. The subdiffusive CTRW model displays anomalous property no. 3 and does not have a tractable likelihood. It shows weak ergodicity breaking [15], ageing [16, 17], and non thermal plateau convergence when confined [18], and it has discontinuous paths. Throughout this work, these subdiffusive walks will be labeled CTRW. We also consider an example of superdiffusive ones, Levy walks (see below).

- **Superdiffusive Levy walk** (LW) [19, 20, 21]. LW belongs to the CTRW class of models, but instead of performing discontinuous jumps their motion is continuous and composed of a series of uncorrelated "flights" of constant speed generated from given flight time and distance distributions, $\psi(\tau)$ and $f(\mathbf{\Delta})$, respectively. Here, we consider a subset of the LW class corresponding to superdiffusive motion with a flight time distribution scaling as $\psi(\tau) \sim \tau^{-1-\sigma}$, and with a distance distribution that is conditional on τ and given by $f(\mathbf{\Delta}) \propto \delta(|\mathbf{\Delta}| - \boldsymbol{v}\tau)$, where v is a constant speed parameter and δ is the Dirac delta function. The superdiffusive LW displays anomalous property no. 3, it exhibits weak ergodicity breaking and its likelihood is intractable.

- **Annealed transit time motion** (ATTM) [22, 23]. The ATTM model considers a random walker in an annealed heterogeneous environment where the diffusivity varies over space and time. In the ATTM, the random walker has a diffusivity that is piecewise constant over time with values drawn from $(D) \sim D^{\sigma-1}$ and with resting time at each diffusivity level drawn from a conditional distribution $p(\tau|D)$. We consider here $p(\tau|D) = \delta(\tau - D^{-\gamma})$ and the parameter range $\sigma < \gamma < \sigma + 1$ (defined as Regime I in ref [22]), which leads to subdiffusive motion with an anomalous exponent of $\alpha = \sigma/\gamma$. Subdiffusive ATTM displays both anomalous properties no. 1 and 2, and it exhibits weak ergodicity breaking and ageing. It is a physically irrealist random walk but can be instrumental in describing transiently the dynamics in a heterogeneous diffusive environement.

- **Ornstein-Uhlenbeck process** (OU). It is the motion of a Brownian walker diffusing in a harmonic potential well $U(x) = \frac{\theta}{2}x^2$ (hence linked to property no. 1). Its apparent anomalous diffusion exponent is 1 at short times but 0 at longer time scales, due to the confinement imposed by the potential. Between these two limits, its dynamics exhibit anti-correlations and thus resemble that of a subdiffusive fBM.

I.2 Biomolecules in cells

I.2.1 Physical orders of magnitude

a) Sizes

Cells typically measure between a few microns and a few tens of microns. But between the membrane and the nucleus, the cytoplasm itself is not a homogeneous media, and is

Figure I.3: An artist's view of a cell's interior. Image by Evan Ingersoll and Gael McGill.

structured at different scales. It contains organelles such as the Golgi apparatus, endosomes and mitochondria, whose characteristic scale is of the order of the micrometer. Then, one order of magnitude smaller, the endoplasmic reticulum and cytoskeletal elements such as microtubules and actin filaments provide another kind of structure to the cytoplasm. In the same size range are viruses, spherical structures with a diameter of the order of 100 nanometers. These components naturally interfere with the diffusion of biomolecules present in the cytoplasm, guiding and regulating it. The effect of this structure on the diffusion of a molecule is of course dependent of its size, and it has been shown in [24] that particles larger than 50 nanometers are virtually immobile in the cytoplasm, maintained confined by these obstacles. Finally, the typical length scale of a biomolecule is of the order of a few nanometers (hence the term *nanoscale*), the most notable exception being the strains of DNA which, in the cell, are packed so as to be contained in a region of a few hundreds of nanometers, but when elongated span up to a few millimeters. Proteins constitute about half the mass of all organic molecules in the average cell.

b) Densities

Illustrated in figure I.3, the cytoskeleton's crowding is such that the term *molecular packing* has been employed to describe it: taken together, proteins, lipids and sugar constitute about 40% of the cytoplasm's volume [25, 26]. As predicted by numerical simulations of diffusion in packed environments, this very dense and structured spatial organization of the cell has notably been shown to induce anomalous diffusion [27]. The local variations of protein concentrations throughout the cell are thought to be at the origin of ageing-related protein aggregation [28]. In humans cells, the order of magnitude of protein density is of one million proteins per cubic micrometer of cell. Although the level of expression of proteins types (i.e. the number of copies of one type of molecule) spans order of magnitudes among the 20,000 types of the human proteome, we can roughly estimate the average concentration of proteins of a given type to 50 proteins per cubic micrometer. The characteristic distance between proteins of a same type is thus of the order of a few hundreds of nanometers: relatively to a protein's size, this corresponds to one or two humans per football pitch. Note however that highly expressed proteins might be much closer to each other. The order of magnitude of receptor densities at the membrane is comparable: the concentration of T-cell receptors in mouse T-cells was for instance measured at 140-190 receptors per square micrometer [29], which corresponds to an average distance of the order of 100 nanometers between neighbor receptors.

c) Time scales

Most biomolecules have a limited time where they remain in their active conformation. In mammalian cells, proteins are active on average for one or two days, but the longevity varies a lot across protein types, the fastest ones being degraded in a few minutes while others are sustained for years. The time scale of encounters between proteins of a same type is the second (assuming a free diffusion at the membrane, with a diffusion coefficient of the order of $10^{-1}\mu m^2.s^{-1}$). The folding of proteins and their evolution between different conformations occurs, at the fastest, in a few tens of milliseconds [30]. Finally, the shortest time scale of protein dynamics might be the picosecond, corresponding to the pace at which collisions between a protein and single water molecules occur, as illustrated in figure I.4.

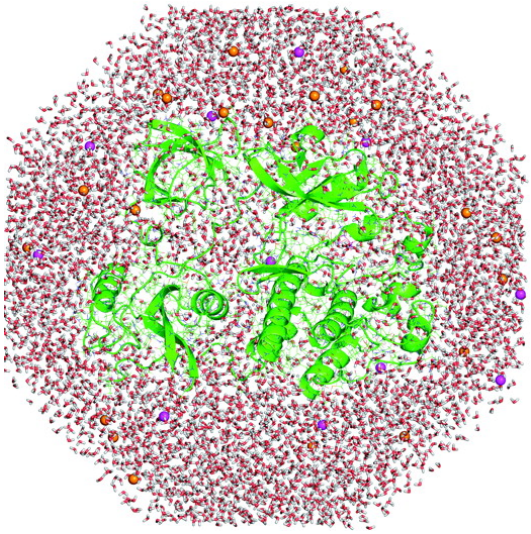

Figure I.4: A protein surrounded by molecules of water and salt. Image obtained by numerical simulation, taken from [31]

I.2.2 Membrane organisation and signal transduction

The cell membrane is a 4 nanometer-thick bilayer of lipid whose structure is widely shared across the living world. It is a highly heterogeneous environment: its physical properties vary following spatial patterns whose sizes cover several scales [32, 33], as it is the theater of a variety of biological mechanisms, notably the transduction of signal coming in and out of the cell, carried by receptors and signaling proteins. This organization, depicted in figure

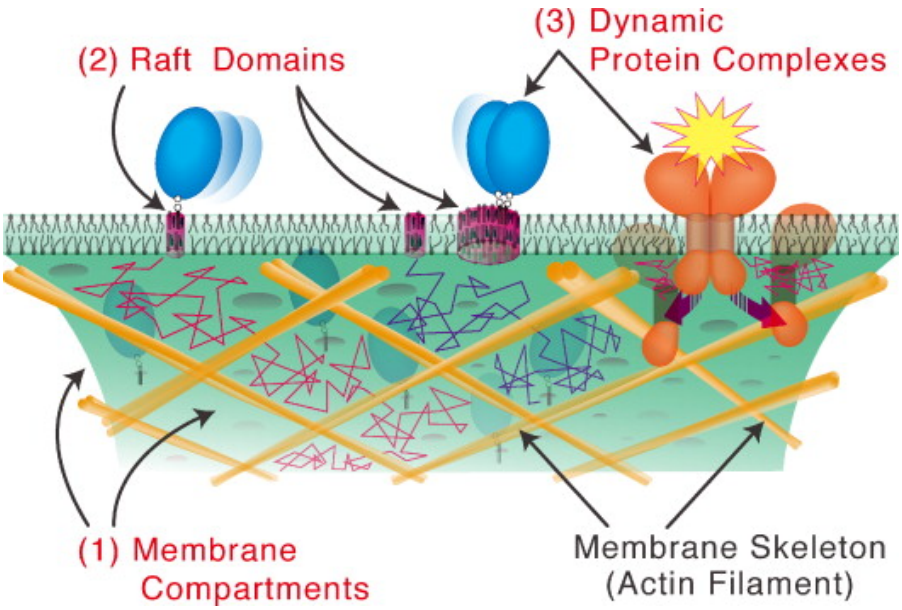

Figure I.5: The three scales of membrane organisation: compartments, raft domains and protein complexes. Taken from [32].

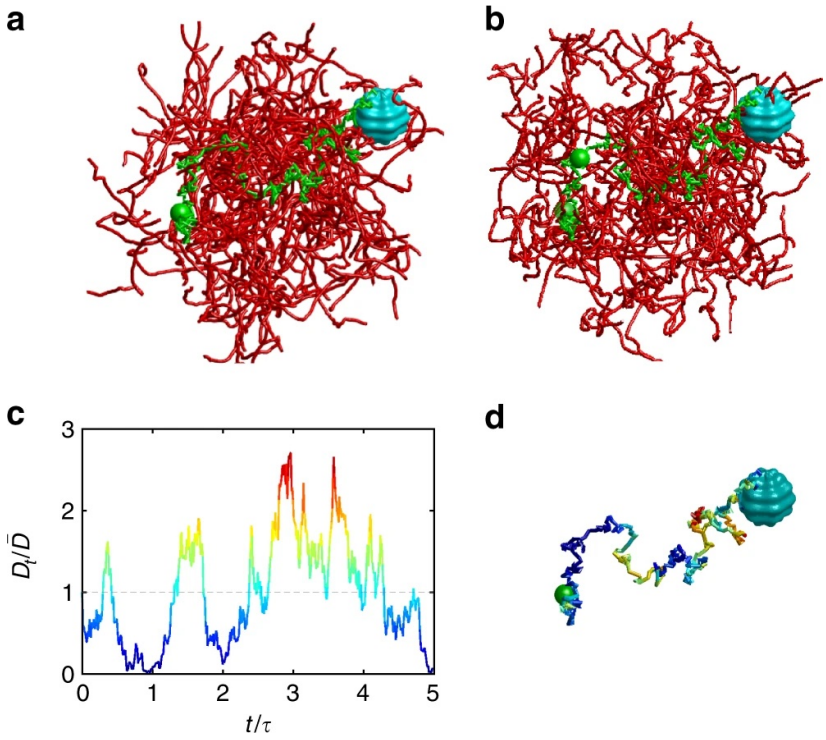

Figure I.6: Illustration of a search process in a crowded environment: the green particle diffuses toward the light blue reaction site and experiences variations of its apparent diffusion coefficient. Taken from [35]

I.5, directly affects the dynamics of proteins diffusing at the membrane, notably inducing non-ergodicity as evidenced by [34].

First, the membrane is compartmentalized in regions of 40-300 nanometers of diameter by actin-membrane skeleton, which forms barriers limiting the diffusion of transmembrane proteins. Intercompoartmental transitions, though hindered, are nonetheless possible when proteins have a sufficiently high kinetic energy, when an actin filament breaks or when thermal fluctuations create a temporary space between the membrane and the underlying cytoskeleton: this creates co-called "hop diffusion" dynamics.

Then, some regions whose size ranges from a few nanometers to a few dozens of nanometers are specifically enriched in cholesterol, GPI-anchored proteins such as receptors or adhesion proteins, and a special type of lipids. Called "raft domains", these regions have different properties depending on whether they are stimulated or not, *i.e.* whether one of their receptor has bound and formed a dimer or a greater oligomer. Prior to stimulation, their lifetime is very short and does not exceed one second, but the stimulation induces stabilization allowing them to exist for 100-1,000 seconds and to grow larger: these raft domains are thus stabilized *on demand*, to allow the realization of a given function in response to a stimulus.

I.2.3 Example of diffusion-based phenomenons: search processes

Although highly ordered phenomenons such as cell division, differentiation and migration make it clear that the inner cell machinery is a well regulated highly dynamical system,

there is a challenge in bridging the gap between the high stochasticity of all phenomena linked to individual biomolecules and the large scale dynamics involved in the natural cell cycle.

The following paragraphs give a few examples of biological functions relying on diffusion-mediated search processes, of which figure I.6 gives an illustration. The study of search strategies and of first passage times is a very rich topic; in the present paragraphs, we only mention a few examples, highlighting the presence of diffusive motion in processes requiring the encounter of two particles.

It has been observed that enzymes meant to bind a specific region of the DNA can efficiently bind to their target. A random walk model combining both one-dimensional diffusion along DNA and rapid three-dimensional jumps between remote regions of the genome was proposed in [36], recovering the efficiency of the search and matching characteristic times measured *in-vivo*. The search dynamics of CRISPR-*Cas9*, the protein used for genome editing, exhibit a similar pattern, observed in [37]. It diffuses throughout the cytoplasm and nucleus before it finds the genome region which its RNA-guide is designed to bind. The diffusion is fast in regions remote from the chromosomes, and much slower when it approaches them, due to interactions of the protein complex with the chromatin. Still in the nucleus, the regulation of genetic expression by geometric structure has been explored by [38]. They show that, in confined environments, a compact walker tends to visit sites close to its initial position much sooner than remote ones, while this difference is less pronounced for non-compact walkers. Hence, a transcription factor with a compact dynamics will favor the expression of genes located close to its starting point much more than that of remote ones.

Chemical processes might be diffusion limited with challenges for reactants to find each other in the cell. Catalysis by enzymes as well as protein aggregation and complexation, which are key chemical processes occurring in the cell, are of this type [39]. Hence, the biological functions of which they constitute the basic components eventually depend on the diffusion properties of their reactants in the cell media. The mean first passage time of a subdiffusive walker in a confined environment (a model for a protein interacting with components of the endoplasmic reticulum, for instance) has notaby been studied in [40]. In [41], it has been numerically shown that the anomalous diffusion of enzymes, due to the presence of obstacles, strongly influence the kinetics of the chemical reactions in which they are involved.

I.3 Imaging biomolecules motion dynamics

Our knowledge of lateral diffusion of proteins in cells, whether at the membrane or in the cytoplasm, is based both on the understanding of fundamental chemical properties of membrane components and on optical observations of the behavior of these components in live cells or synthesized assemblies such as artificial model membranes. Here, we introduce examples of imaging technologies allowing to probe diffusion phenomenons, starting from the diffraction-limited ones and ending with super-resolution techniques, which provide the localizations and tracks analyzed in the rest of this thesis.

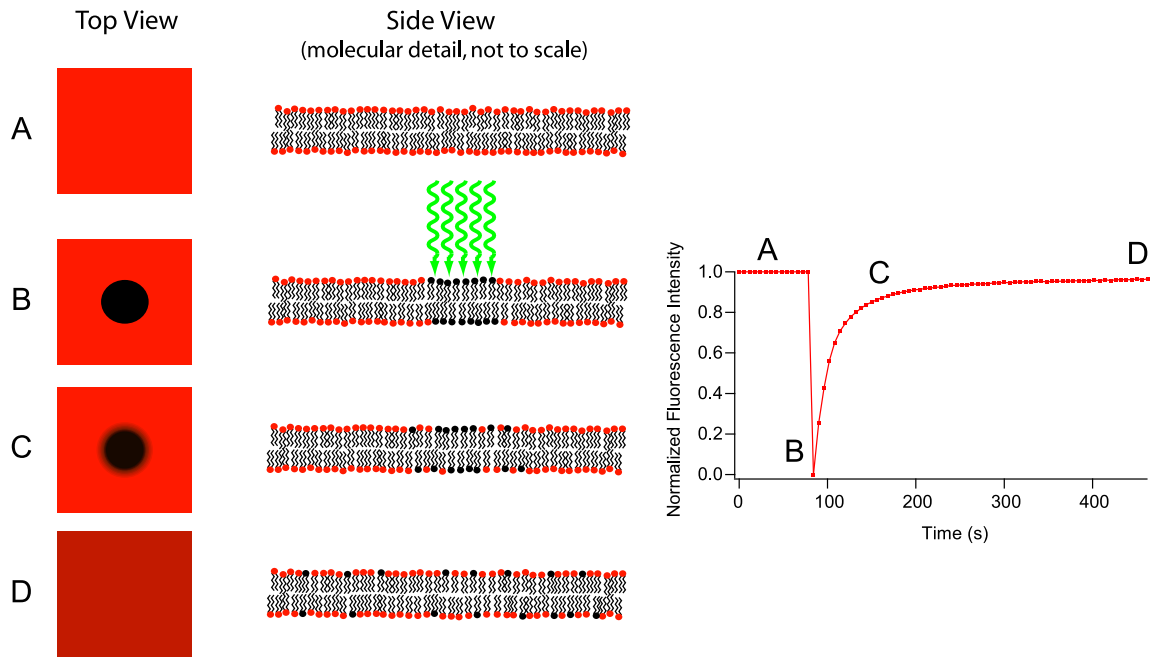

Figure I.7: Principle of FRAP. Adapted from Wikipedia

I.3.1 Diffraction-limited modalities

a) Fluorescence recovery after photobleaching (FRAP)

FRAP experiments allow one to measure the time necessary for the population of biomolecules (proteins or lipids) present in a given region of the cell (either at the membrane or in the cytoplasm) to be replaced, as illustrated in I.7. Molecules of interest are tagged by a fluorescent marker, and a specific location is exposed to a strong laser intensity which irreversibly photobleaches the labels, causing a drop in fluorescence intensity. Progressively, the photobleached region is repopulated by fluorescent molecules, and the fluorescence intensity recovers. The temporal evolution of the fluorescence intensity informs about the observed diffusion process. The simplest model allows one to recover a diffusion coefficient from the relaxation time. More complex models can be fitted on the observed curve in order to determine, for instance, the fraction of mobile and immobile molecules: see [42] for an application to α-synuclein, a protein studied in chapter V.

b) Fluorescence correlation spectorcopy (FCS)

In FCS, the fluorescence intensity $I(t)$ of a small volume containing labeled molecules (of the order of one cubic micron of cell, containing a few hundreds of molecules of interest) is recorded for a period of time, in order to access the global auto-correlation $G(\tau) = \frac{\langle I(t+\tau)I(t)\rangle}{\langle I(t)\rangle^2}$ of the signal. The fluorescence intensity being proportional to the number of molecules in the considered volume, the temporal correlation of its fluctuations informs about the diffusion of molecules in and out of the volume. In the same manner as in FRAP experiment analysis, various models can be used to interpret $G(\tau)$ curves, possibly accounting for anomalous

diffusion or the existence of several modes of diffusion.

c) Limitations

While FRAP and FCS provide good means of measuring the diffusion of biomolecules in cells on the micrometer scale, they do not directly account for the small scale heterogeneity of diffusion dynamics. Indeed, both in FRAP and FCS experiments, the measured diffusion coefficient corresponds to an "average" diffusion in the region, not informative of local discrepancies. In addition, as remarked in [43], the diffusion coefficients provided by these techniques depend on the considered region's topology. Furthermore, although FRAP yields a direct measurement of mobile and immobile fractions, the extent at which these two methods allow one to probe the existence of several diffusion modes of a given protein is rather limited, given that all measurements correspond to aggregates over an ensemble of molecules. In particular, one should be aware of the assumptions made when choosing the model used for fitting the curves – model selection will be discussed in paragraph II.1.2.

I.3.2 Single particle tracking (SPT) with super-resolution

a) Why SPT is relevant to study biomolecules motion

In contrary to diffraction-limited techniques, which are bound to measure properties of an ensemble of diffusing particles, single particle tracking allows one to distinguish each unique molecule and follow its dynamics during a limited time frame. This is particularly suited to characterizing the heterogeneity of dynamics which often exist across a population of diffusing biomolecules, be it transcription factors exploring the genome [44], or glycine receptors around synapses [45]. In addition, super-resolution imaging technologies, on which SPT relies, provide a very high spatial resolution which enables one to probe the fine-grained heterogeneity of an environment's physical properties. These technologies are detailed in the following.

The analysis of SPT data however poses more challenges than that of FCS or FRAP experiment. First, the numerous processing steps required to obtain tracks from a microscope recording oblige one to validate that they do not introduce artifacts. Details on this process for PALM are notably provided in VI, where a software tool for tuning and running processing pipelines on batches of PALM movies is presented. Second, the observed data is much more complex than in the two aforementioned modalities: raw trajectories are high-dimensional vectors of varying length, and are thus not easily compared or aggregated. Finally, the quantity and spatial density of observed trajectories do limit the level of precision that one might have when measuring certain properties of the dynamics in a given region. The relevant spatial resolution as well as the nature of measured properties should be carefully set so as to capture relevant information with enough reliability. This work provides methods to take advantage of the granularity offered by SPT measurements while accounting for the statistical challenges posed by the complexity and relative scarcity of the data.

b) Single molecule localization microscopy (SMLM)

The size of the luminous spot created by a single emitter on an optical sensor is bounded by Abbe's law, setting the limit to a few hundreds of nanometers, depending on the characteris-

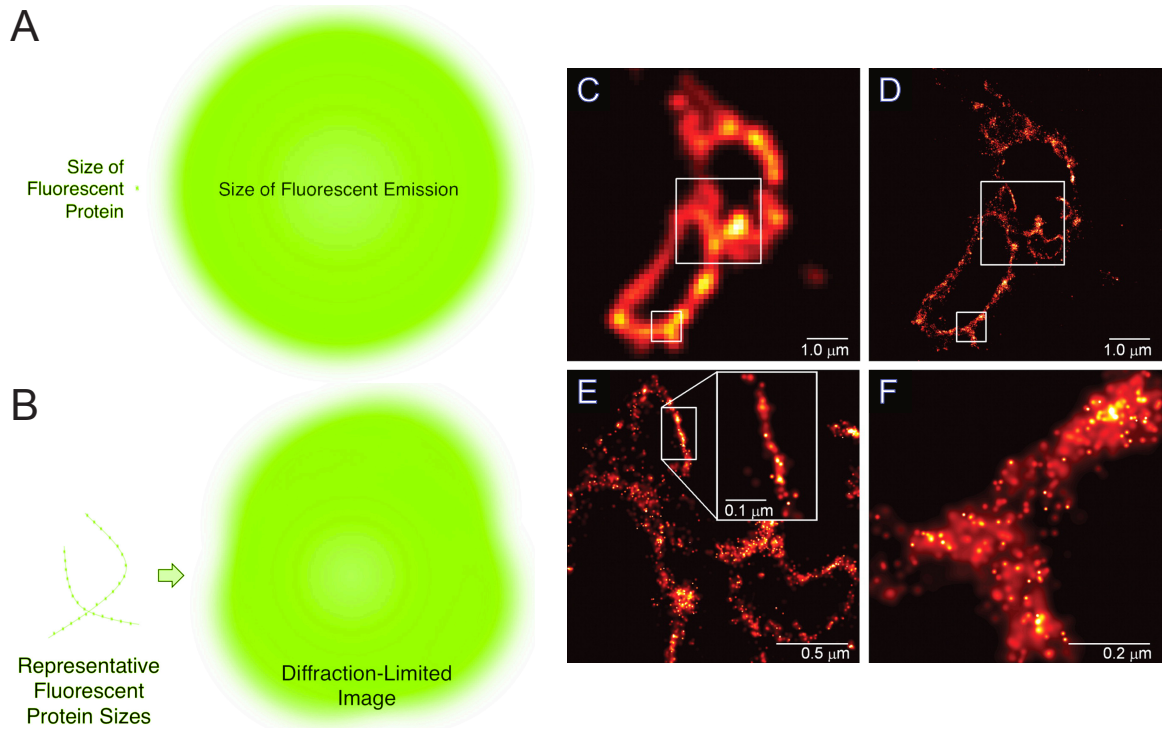

Figure I.8: Principle of single molecule localization microscopy. A: Relative sizes of a molecule and its light spot. B: Example of hardly distinguishable molecules. C: Image of CD63 on a Cos-7 cell, acquired in diffraction-limited mode. D: same region, imaged in PALM. E-F zooms on specific portions of the cell. Adapted from [46] and [47]

.

tics optical system and the wavelength of the signal. This spatially spread spot, designated as the point spread function (PSF) in the field of optics, is due to diffraction and the wave nature of light. However, there are means to circumvent this limitation and separate discrete luminous spots formed by neighbor sources, as intuited by [48].

FRAP and FCS both rely on classical fluorophores, constantly emitting light in response to an excitation, until they eventually photobleach. Hence, in these settings, two fluorescent molecules separated by a distance smaller than the PSF diameter can not be distinguished from each other; only the increased total intensity would signal the presence of two distinct fluorophores. Figure I.8 illustrates the overlap between neighboring PSFs, hiding the detailed structure of the molecules' positions. In the following, we introduce super-resolution techniques, including photo-activated localization microscopy (PALM), used to acquire the data analyzed in chapter V.

SMLM overcomes Abbe's limit by temporally separating the emission of light coming from neighbor molecules. This general idea is declined in several variants, of which [49] and [50] provide thorough reviews. In most cases, it takes advantage of fluorophores engineered so that, excited by a laser beam, they enter a bright mode and bleach in a stochastic manner, the reversibility of which depends on the exact technique used. The pace at which fluorescent markers enter this bright phase, as well as the characteristic duration of the emission, can be calibrated according to the density of probes so as to ensure that the probability of two

neighboring molecules to fluoresce simultaneously is weak or negligible. This then allows one to assume that each observed spot is due to a single bright particle, the coordinates of which can be determined with a precision of a few dozens of nanometers by fitting the PSF.

Although the most common PSF-fitting methods only give access to planar coordinates of the emitter, a few techniques allow to recover the axial coordinate, opening the possibility of performing three-dimensional localization – although the axial resolution is often lower than the transversal one. When no such technique is employed, cells are imaged in Total internal reflection fluorescence (TIRF) to avoid background fluorescence and image only a slice of the volume: the excitation laser beam is tilted so as to have a critical incidence angle on the coverslip. Hence, only the evanescent wave is transmitted to the cell, exciting a layer whose depth does not exceed one wavelength (a few hundreds of nanometers). This modality is thus particularly well suited to image membrane proteins, or structures located just underneath the membrane.

c) Photo-activated localization microscopy (PALM)

PALM, introduced by Betzig et al. for fixed samples and adapted in [51] to live imaging, is the specific variant of SMLM used to access the positions of proteins diffusing in living cells analysed in the following chapters of this thesis. Prior to being imaged, cells are transfected with a DNA vector containing the genetic code of the protein of interest fused with a fluorescent protein, associated with a promoter of expression ensuring a certain level of transcription. The fluorescent proteins used are photo-activable: they stochastically become bright in response to a laser excitation and then irreversibly bleach [52]. PALM has been used to track proteins in a wide range of cell types, from bacteria [53] to embryonic stem cells [44]. A downside of PALM is the toxicity of the prolonged exposition to the excitation laser, which might damage the cell as observed in [54]. Furthermore, the promoter of expression potentially light lead to an over-expression of the protein of interest and thus induce a change in the cell's physiological activity, perturbating the state of the observed system. In addition, due mostly to the transfection step and the sensitivity of the acquired movies to laser intensities (excitation and illumination power have to be cautiously set), PALM has the inconvenient of requiring more experimentalist's work and fine-tuning than other SMLM techniques. Nonetheless, bearing in mind these limitations, PALM remains a technique of choice for probing the diffusion properties of proteins in living cells. The main advantages of this technique is that it is compatible with live-cell imaging (contrary to other SMLM modalities requiring the prior fixation of samples) and that the perturbation undergone by the imaged cell can be kept relatively low.

d) Stimulated emission depletion (STED)

Stimulated emission depletion microscopy, introduced by [55], reaches sub-diffraction resolution by separating the light emitted by neighbor fluorophores not by stochastic blinking – as in PALM, where it is unlikely that two blinking events colocalize both in space and time – but by forcing the transition towards the dark state of some fluorophores in a well defined region. The depletion of all excited fluorophores except those located in a central region is forced by a depletion-stimulating beam, and only spontaneous depletions occuring in the untouched region are recorded. Excitation and depletion-stimulating beams, synchronised and concentric, are focused on regions which, due to the diffraction, cannot be smaller than

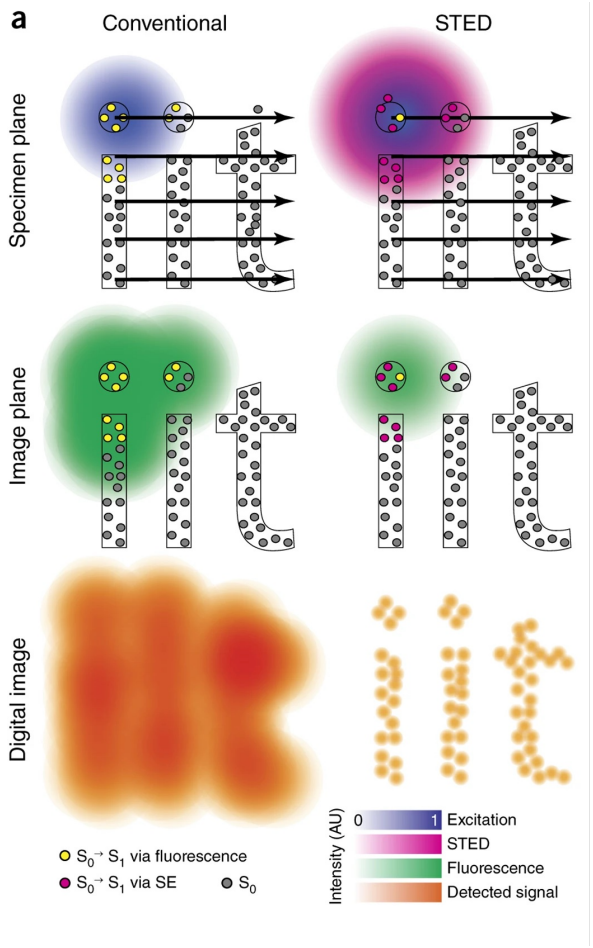

Figure I.9: Principle of STED. Image adapted from [56]

a few hundreds of nanometers. The shapes of these regions however differ, as can be seen on figure I.9: the excitation is a Gaussian spot while the stimulated depletion beam has a torus geometry leaving the central region un-depleted. STED can reach resolutions of the same order of magnitude as PALM or STORM. However, the intensity peaks required by the depletion stimulation might induce photo-toxicity, hampering its application to live-cell imaging. An extensive review of STED methods, capabilities and limitations is provided in [56].

e) An alternative to fluorophores: quantum dots and nanoparticles

If one is more interested in measuring properties of the environment than the dynamics of a given type of biomolecule diffusing in it, it is possible to introduce synthetic particles in the investigated environment, the fluorescence properties of which might be better controlled and more favorable than those of fluorophores used to label biomolecules. Furthermore, they have the advantage of not requiring laser excitation, which limits the damage caused to the cells by the imaging process. Nonetheless, despite some efforts, the size and mass of these probes are often of the same order of magnitude (if not larger) than the molecules they tag, which is

detrimental to the relevance of the insights they provide on the unperturbed dynamics of the walker [57]. Latex spheres (500 nm) and gold nanoparticles (40 nm) have first been used as such probes [58], but quantum dots were argued to offer a better compromise between the size of the bead and its optical properties [45], to the point that they could even be used as labels of proteins without too strongly affecting the diffusion. Very stable emitters, they allow one to track individual diffusing particle for tens of minutes while fluorophore-based movies rarely exceed a few seconds of uninterrupted blinking. This length is very valuable as it enables the observation of long cycles in a protein's motion. This high photostability however comes at the expense of spatial density: if the density of simultaneously emitting particles is too important, it hinders the reconstruction of trajectories form localizations. Hence, space cannot be sampled as regularly as in the case when many fast-blinking fluorophores are used.

Bibliography

[1] Robert Brown. A brief account of microscopical observations made in the months of june, july and august 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies. *The philosophical magazine*, 4(21):161–173, 1828.

[2] Albert Einstein et al. On the motion of small particles suspended in liquids at rest required by the molecular-kinetic theory of heat. *Annalen der physik*, 17(549-560):208, 1905.

[3] Jean Perrin. Mouvement brownien et molécules. *J. Phys. Theor. Appl.*, 9(1):5–39, 1910.

[4] Jean-Philippe Bouchaud and Antoine Georges. Anomalous diffusion in disordered media: statistical mechanisms, models and physical applications. *Physics reports*, 195(4-5):127–293, 1990.

[5] Ralf Metzler, Jae-Hyung Jeon, Andrey G. Cherstvy, and Eli Barkai. Anomalous diffusion models and their properties: non-stationarity, non-ergodicity, and ageing at the centenary of single particle tracking. *Phys. Chem. Chem. Phys.*, 16(44):24128–24164, 2014. doi: 10.1039/c4cp03465a. URL https://doi.org/10.1039/c4cp03465a.

[6] Gardiner Crispin. *Handbook of Stochastic Methods: for Physics, Chemistry and natural sciences*. Springer, 4th edition.

[7] Benoit B. Mandelbrot and John W. Van Ness. Fractional brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, October 1968. doi: 10.1137/1010093. URL https://doi.org/10.1137/1010093.

[8] Weihua Deng and Eli Barkai. Ergodic properties of fractional brownian-langevin motion. *Physical Review E*, 79(1), January 2009. doi: 10.1103/physreve.79.011112. URL https://doi.org/10.1103/physreve.79.011112.

[9] Michael J. Saxton. Anomalous subdiffusion in fluorescence photobleaching recovery: A monte carlo study. *Biophysical Journal*, 81(4):2226–2240, October 2001. doi: 10.1016/s0006-3495(01)75870-5. URL https://doi.org/10.1016/s0006-3495(01)75870-5.

[10] S. C. Lim and S. V. Muniandy. Self-similar gaussian processes for modeling anomalous diffusion. *Physical Review E*, 66(2), August 2002. doi: 10.1103/physreve.66.021114. URL https://doi.org/10.1103/physreve.66.021114.

[11] Vittoria Sposini, Ralf Metzler, and Gleb Oshanin. Single-trajectory spectral analysis of scaled brownian motion. *New Journal of Physics*, 21(7):073043, jul 2019. doi: 10.1088/1367-2630/ab2f52. URL https://doi.org/10.1088%2F1367-2630%2Fab2f52.

[12] Jae-Hyung Jeon, Aleksei V. Chechkin, and Ralf Metzler. Scaled brownian motion: a paradoxical process with a time dependent diffusivity for the description of anomalous diffusion. *Phys. Chem. Chem. Phys.*, 16(30):15811–15817, 2014. doi: 10.1039/c4cp02019g. URL https://doi.org/10.1039/c4cp02019g.

[13] Harvey Scher and Elliott W. Montroll. Anomalous transit-time dispersion in amorphous solids. *Physical Review B*, 12(6):2455–2477, September 1975. doi: 10.1103/physrevb.12.2455. URL https://doi.org/10.1103/physrevb.12.2455.

[14] Jean-Philippe Bouchaud and Antoine Georges. Anomalous diffusion in disordered media: Statistical mechanisms, models and physical applications. *Physics Reports*, 195(4-5):127–293, November 1990. doi: 10.1016/0370-1573(90)90099-n. URL https://doi.org/10.1016/0370-1573(90)90099-n.

[15] Michael A. Lomholt, Irwin M. Zaid, and Ralf Metzler. Subdiffusion and weak ergodicity breaking in the presence of a reactive boundary. *Physical Review Letters*, 98(20), May 2007. doi: 10.1103/physrevlett.98.200603. URL https://doi.org/10.1103/physrevlett.98.200603.

[16] Johannes H. P. Schulz, Eli Barkai, and Ralf Metzler. Aging renewal theory and application to random walks. *Physical Review X*, 4(1), February 2014. doi: 10.1103/physrevx.4.011028. URL https://doi.org/10.1103/physrevx.4.011028.

[17] Henning Krüsemann, Aljaž Godec, and Ralf Metzler. First-passage statistics for aging diffusion in systems with annealed and quenched disorder. *Physical Review E*, 89(4), April 2014. doi: 10.1103/physreve.89.040101. URL https://doi.org/10.1103/physreve.89.040101.

[18] S. Burov, R. Metzler, and E. Barkai. Aging and nonergodicity beyond the khinchin theorem. *Proceedings of the National Academy of Sciences*, 107(30):13228–13233, July 2010. doi: 10.1073/pnas.1003693107. URL https://doi.org/10.1073/pnas.1003693107.

[19] J. Klafter and G. Zumofen. Lévy statistics in a hamiltonian system. *Physical Review E*, 49(6):4873–4877, June 1994. doi: 10.1103/physreve.49.4873. URL https://doi.org/10.1103/physreve.49.4873.

[20] Benoit Mandelbrot. *The fractal geometry of nature*. W.H. Freeman, San Francisco, 1982. ISBN 978-0716711865.

[21] V. Zaburdaev, S. Denisov, and J. Klafter. Lévy walks. *Reviews of Modern Physics*, 87(2):483–530, June 2015. doi: 10.1103/revmodphys.87.483. URL https://doi.org/10.1103/revmodphys.87.483.

[22] P. Massignan, C. Manzo, J.A. Torreno-Pina, M.F. García-Parajo, M. Lewenstein, and G.J. Lapeyre. Nonergodic subdiffusion from brownian motion in an inhomogeneous medium. *Physical Review Letters*, 112 (15), April 2014. doi: 10.1103/physrevlett.112.150603. URL https://doi.org/10.1103/physrevlett.112.150603.

[23] Takuma Akimoto and Eiji Yamamoto. Distributional behavior of diffusion coefficients obtained by single trajectories in annealed transit time model. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(12):123201, December 2016. doi: 10.1088/1742-5468/2016/12/123201. URL https://doi.org/10.1088/1742-5468/2016/12/123201.

[24] Katherine Luby-Phelps, Philip E Castle, D Lansing Taylor, and Frederick Lanni. Hindered diffusion of inert tracer particles in the cytoplasm of mouse 3t3 cells. *Proceedings of the National Academy of Sciences*, 84(14):4910–4913, 1987.

[25] Alice B Fulton. How crowded is the cytoplasm? *Cell*, 30(2):345–347, 1982.

[26] Michael J Saxton. Lateral diffusion in an archipelago. dependence on tracer size. *Biophysical journal*, 64(4):1053–1062, 1993.

[27] Matthias Weiss, Markus Elsner, Fredrik Kartberg, and Tommy Nilsson. Anomalous subdiffusion is a measure for cytoplasmic crowding in living cells. *Biophysical journal*, 87(5):3518–3524, 2004.

[28] Anne-Sophie Coquel, Jean-Pascal Jacob, Mael Primet, Alice Demarez, Mariella Dimiccoli, Thomas Julou, Lionel Moisan, Ariel B Lindner, and Hugues Berry. Localization of protein aggregation in escherichia coli is governed by diffusion and nucleoid macromolecular crowding effect. *PLoS computational biology*, 9(4):e1003038, 2013.

[29] Björn F Lillemeier, Manuel A Mörtelmaier, Martin B Forstner, Johannes B Huppa, Jay T Groves, and Mark M Davis. Tcr and lat are expressed on separate protein islands on t cell membranes and concatenate during activation. *Nature immunology*, 11(1):90–96, 2010.

[30] TR Sosnick, L Mayne, R Hiller, and SW Englander. The barriers in protein folding. *Nature structural biology*, 1(3):149–156, 1994.

[31] Martin Karplus and John Kuriyan. Molecular dynamics and protein function. *Proceedings of the National Academy of Sciences*, 102(19):6679–6685, 2005.

[32] Akihiro Kusumi, Takahiro K Fujiwara, Nobuhiro Morone, Kenta J Yoshida, Rahul Chadda, Min Xie, Rinshi S Kasai, and Kenichi GN Suzuki. Membrane mechanisms for signal transduction: the coupling of the meso-scale raft domains to membrane-skeleton-induced compartments and dynamic protein complexes. In *Seminars in cell & developmental biology*, volume 23, pages 126–144. Elsevier, 2012.

[33] Diego Krapf. Compartmentalization of the plasma membrane. *Current opinion in cell biology*, 53:15–21, 2018.

[34] Carlo Manzo, Juan A Torreno-Pina, Pietro Massignan, Gerald J Lapeyre Jr, Maciej Lewenstein, and Maria F Garcia Parajo. Weak ergodicity breaking of receptor motion in living cells stemming from random diffusivity. *Physical Review X*, 5(1):011021, 2015.

[35] Yann Lanoiselée, Nicolas Moutal, and Denis S Grebenkov. Diffusion-limited reactions in dynamic heterogeneous media. *Nature communications*, 9(1):1–16, 2018.

[36] M Coppey, O Bénichou, R Voituriez, and M Moreau. Kinetics of target site localization of a protein on dna: a stochastic approach. *Biophysical journal*, 87(3):1640–1649, 2004.

[37] Spencer C Knight, Liangqi Xie, Wulan Deng, Benjamin Guglielmi, Lea B Witkowsky, Lana Bosanac, Elisa T Zhang, Mohamed El Beheiry, Jean-Baptiste Masson, Maxime Dahan, et al. Dynamics of crispr-cas9 genome interrogation in living cells. *Science*, 350(6262):823–826, 2015.

[38] Olivier Bénichou, C Chevalier, Joseph Klafter, B Meyer, and Raphael Voituriez. Geometry-controlled kinetics. *Nature chemistry*, 2(6):472–477, 2010.

[39] Nicolas Dorsaz, Cristiano De Michele, Francesco Piazza, Paolo De Los Rios, and Giuseppe Foffi. Diffusion-limited reactions in crowded environments. *Physical Review Letters*, 105(12):120601, 2010.

[40] T Guérin, N Levernier, O Bénichou, and R Voituriez. Mean first-passage times of non-markovian random walkers in confinement. *Nature*, 534(7607):356–359, 2016.

[41] Hugues Berry. Monte carlo simulations of enzyme reactions in two dimensions: fractal kinetics and spatial segregation. *Biophysical journal*, 83(4):1891–1901, 2002.

[42] Kateri J Spinelli, Jonathan K Taylor, Valerie R Osterberg, Madeline J Churchill, Eden Pollock, Cynthia Moore, Charles K Meshul, and Vivek K Unni. Presynaptic alpha-synuclein aggregation in a mouse model of parkinson's disease. *Journal of Neuroscience*, 34(6):2037–2050, 2014.

[43] Eric AJ Reits and Jacques J Neefjes. From fixed to frap: measuring protein mobility and activity in living cells. *Nature cell biology*, 3(6):E145, 2001.

[44] Jiji Chen, Zhengjian Zhang, Li Li, Bi-Chang Chen, Andrey Revyakin, Bassam Hajj, Wesley Legant, Maxime Dahan, Timothée Lionnet, Eric Betzig, et al. Single-molecule dynamics of enhanceosome assembly in embryonic stem cells. *Cell*, 156(6):1274–1285, 2014.

[45] Maxime Dahan, Sabine Levi, Camilla Luccardini, Philippe Rostaing, Beatrice Riveau, and Antoine Triller. Diffusion dynamics of glycine receptors revealed by single-quantum dot tracking. *Science*, 302 (5644):442–445, 2003.

[46] Mohamed Hossam El Beheiry. *Towards whole-cell mapping of single-molecule dynamics*. PhD thesis, Université Pierre et Marie Curie-Paris VI, 2015.

[47] Eric Betzig, George H Patterson, Rachid Sougrat, O Wolf Lindwasser, Scott Olenych, Juan S Bonifacino, Michael W Davidson, Jennifer Lippincott-Schwartz, and Harald F Hess. Imaging intracellular fluorescent proteins at nanometer resolution. *Science*, 313(5793):1642–1645, 2006.

[48] Eric Betzig. Proposed method for molecular optical imaging. *Optics letters*, 20(3):237–239, 1995.

[49] Mickaël Lelek, Melina T Gyparaki, Gerti Beliu, Florian Schueder, Juliette Griffié, Suliana Manley, Ralf Jungmann, Markus Sauer, Melike Lakadamyali, and Christophe Zimmer. Single-molecule localization microscopy. *Nature Reviews Methods Primers*, 1(1):1–27, 2021.

[50] William E Moerner. New directions in single-molecule imaging and analysis. *Proceedings of the National Academy of Sciences*, 104(31):12596–12602, 2007.

[51] Suliana Manley, Jennifer M Gillette, George H Patterson, Hari Shroff, Harald F Hess, Eric Betzig, and Jennifer Lippincott-Schwartz. High-density mapping of single-molecule trajectories with photoactivated localization microscopy. *Nature methods*, 5(2):155–157, 2008.

[52] Robert M Dickson, Andrew B Cubitt, Roger Y Tsien, and William E Moerner. On/off blinking and switching behaviour of single molecules of green fluorescent protein. *Nature*, 388(6640):355–358, 1997.

[53] Achillefs N Kapanidis, Alessia Lepore, and Meriem El Karoui. Rediscovering bacteria through single-molecule imaging in living cells. *Biophysical Journal*, 115(2):190–202, 2018.

[54] Sina Wäldchen, Julian Lehmann, Teresa Klein, Sebastian Van De Linde, and Markus Sauer. Light-induced cell damage in live-cell super-resolution microscopy. *Scientific reports*, 5(1):1–12, 2015.

[55] Stefan W Hell and Jan Wichmann. Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy. *Optics letters*, 19(11):780–782, 1994.

[56] Giuseppe Vicidomini, Paolo Bianchini, and Alberto Diaspro. Sted super-resolved microscopy. *Nature methods*, 15(3):173–182, 2018.

[57] Sripad Ram, Dongyoung Kim, Raimund J Ober, and E Sally Ward. 3d single molecule tracking with multifocal plane microscopy reveals rapid intercellular transferrin transport at epithelial cell barriers. *Biophysical journal*, 103(7):1594–1603, 2012.

[58] Michael J Saxton and Ken Jacobson. Single-particle tracking: applications to membrane dynamics. *Annual review of biophysics and biomolecular structure*, 26(1):373–399, 1997.

II – Inference of biomolecule dynamics

Contents

Inference consists in analysing data to extract quantitative properties from an underlying probability distribution. Challenges in performing inference arise from several factors: the quantity and quality of available data are critical, as much as the relation between the various models available to describe the data as well as the various bias associated to both data acquisition and the models themselves.

Single particle tracking experiments provide nanoscale observations of trajectories of biomolecules diffusing at the membrane or inside the cell. The interpretation of these trajectories is neither straightforward nor unique. While some random walks have analytical likelihood facilitating experimental data analysis, most of them don't. Hence, recent advances in statistical learning, along with increased access to computing power, have opened new possibilities in data analysis. Yet, The growing amount of data generated by biological experiments poses new challenges to the interpretation of observations, but combined with the aforementioned advances, broadens the scope of accessible information.

The following chapter introduces the Bayesian inference framework. Elements on limitations as well as practical difficulties are mentioned, providing the motivation for our developments in chapters III, IV and V. As we introduce inference techniques, we will mention a selection of results in the field of biomolecule diffusion which were obtained using these tools, highlighting the specific interest of one or the other method in each case, showing their strengths and potential pitfalls. The interested reader will find in Lee et al. an interesting discussion of the diverse challenges encountered throughout the analysis of SMLM movies, from the image processing steps to the analysis of trajectories.

II.1 Introduction to Bayesian inference

We introduce notations used throughout the following sections. In this part we will use the example of diffusion inference to exemplify our discussions. We note $\mathbf{x}$ the experimental data, it can be of any type in general although in this precise case it is likely to be the intensity curve resulting from a FRAP experiment, or a set of trajectories observed via SMLM techniques (see section I.3). We assume at least one model $\mathcal{M}$ that can describe our data which is parameterized by a vector $\boldsymbol{\theta}$. In a simple example, diffusion will be Brownian and only characterized by the diffusion D. In more complex examples involving anomalous diffusion, an additional parameter will be the anomalous exponent α (see I.1), or even in cases in which two modes of anomalous diffusion coexist in some proportion μ: then, $\boldsymbol{\theta}$ is a five-dimensional vector $(D_1, D_2, \alpha_1, \alpha_2, \mu)$. Given a model $\mathcal{M}$, the framework of Bayesian inference relates $\mathbf{x}$ and θ, but it does as well provide tools to assess the relevance of one model compared to another, and to aggregate quantities inferred assuming different models. As inference problems are very often encountered when processing experimental observations and measurements, the use of Bayesian methods in physics is concisely reviewed in [2].

II.1.1 Bayes' theorem: Likelihood, prior and evidence

Here, we assume that we have a model $\mathcal{M}$ and a corresponding $\boldsymbol{\theta}$. The aim of Bayesian inference is to estimate the posterior distribution of the parameters given the observations: $P(\boldsymbol{\theta}|\mathbf{x})$. For a fixed set of observations, this is a probability defined over the parameters' space. Bayes' theorem relates the posterior to three other quantities. First, the likelihood of the parameters given the data, noted $\mathcal{L}(\mathbf{x}|\boldsymbol{\theta})$. It encapsulates the assumed generative model of the data: given a value of $\boldsymbol{\theta}$, the hypothesis is that observations are sampled according to this probability law: $\mathbf{x} \sim \mathcal{L}(\mathbf{x}|\boldsymbol{\theta})$. The second quantity is the prior probability of parameters, $P(\boldsymbol{\theta})$. It is meant to incorporate any prior assumptions on the parameters or constrains that are not expressed in the likelihood. The last quantity involved is the evidence, that is, the probability $P(\mathbf{x}|\mathcal{M})$ of the observations given the model. It acts as a normalizing constant and is instrumental to compare different models.

Using these quantities, Bayes' theorem reads as follows:

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}$$

or

$$P(\boldsymbol{\theta}|\mathbf{x}) = \frac{\mathcal{L}(\mathbf{x}|\boldsymbol{\theta})\, P(\boldsymbol{\theta})}{P(\mathbf{x}|\mathcal{M})}$$

As argued in [3], the fact that Bayesian inference relies so clearly on assumptions made about the studied system (notably via the choice of prior distribution), far from introducing undesired subjectivity, strengthens the interpretability of inference results: as assumptions are made clear, they are easy to discuss and modify.

II.1.2 Comparing models: the evidence

Bayesian analysis provides a rigorous way of addressing the question of model comparison and aggregation.

In the above formulation of Bayes' theorem, we omitted to mention the dependency to the model $\mathcal{M}$ of the prior, likelihood and posterior, because it was assumed beforehand that they were expressed for a fixed $\mathcal{M}$. Suppose now that we have two different models $\mathcal{M}_1$ and $\mathcal{M}_2$ and that we would like to know whether one is more plausible than the other given the data. The posterior probability of one model given the data is

$$P(\mathcal{M}_i|\mathbf{x}) = \frac{P(\mathbf{x}|\mathcal{M}_i)P(\mathcal{M}_i)}{P(\mathbf{x})}$$

Assuming that the two models have the same probability *a priori*, we have $P(\mathcal{M}_1) = P(\mathcal{M}_2)$. The main quantities of interest are thus the evidences of models, expressed as follows:

$$P(\mathbf{x}|\mathcal{M}_i) = \int \mathcal{L}(\mathbf{x}|\boldsymbol{\theta}, \mathcal{M}_i)P(\boldsymbol{\theta}|\mathcal{M}_i)d\boldsymbol{\theta}.$$

The evidence encapsulates both the ability of the model to explain the data and the volume of the parameters' space occupied by parameter values yielding a good likelihood: if this is a narrow region, and small changes in the parameters make the likelihood vanish, the evidence will be low ; on the contrary, it will be high if a wide range of parameter values explain the data sufficiently well. Intuitively, this favors models with few parameters, as there are more ways of being wrong when many parameters influence the likelihood: this is known as the *Occam's Razor* [4].

When the number of parameters is low, the derivation might be tractable: Serov et al. provide a good example of a such case, where models of diffusion with and without external force are compared using the ratio of their respective evidences. However, quite often, there exist no formula to simply integrate a likelihood over the space of its parameters. In such cases, it is possible to resort to nested sampling, as in [6] or to Monte Carlo methods, but the computational cost of these usually scales exponentially with the number of parameters, preventing them from being applied to complex problems involving a large numbers of parameters.

Another way of approximating the evidence is to fit a Gaussian to the posterior around its maximum (this notably implies assuming the posterior to have a single mode and no heavy tails). One first has to find the MAP parameters $\boldsymbol{\theta}_{MAP}$, and then to estimate the Hessian of the log-posterior, $\mathbf{H} = -\nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}} \ln(P(\boldsymbol{\theta}_{MAP}|\mathcal{M}_i, \mathbf{x}))$. Then, the evidence can be approximated as follows [7]:

$$P(\mathbf{x}|\mathcal{M}_i) \approx P(\mathbf{x}|\boldsymbol{\theta}_{MAP}, \mathcal{M}_i)P(\boldsymbol{\theta}_{MAP}|\mathcal{M}_i)/\sqrt{\det(\mathbf{H}/2\pi)}.$$

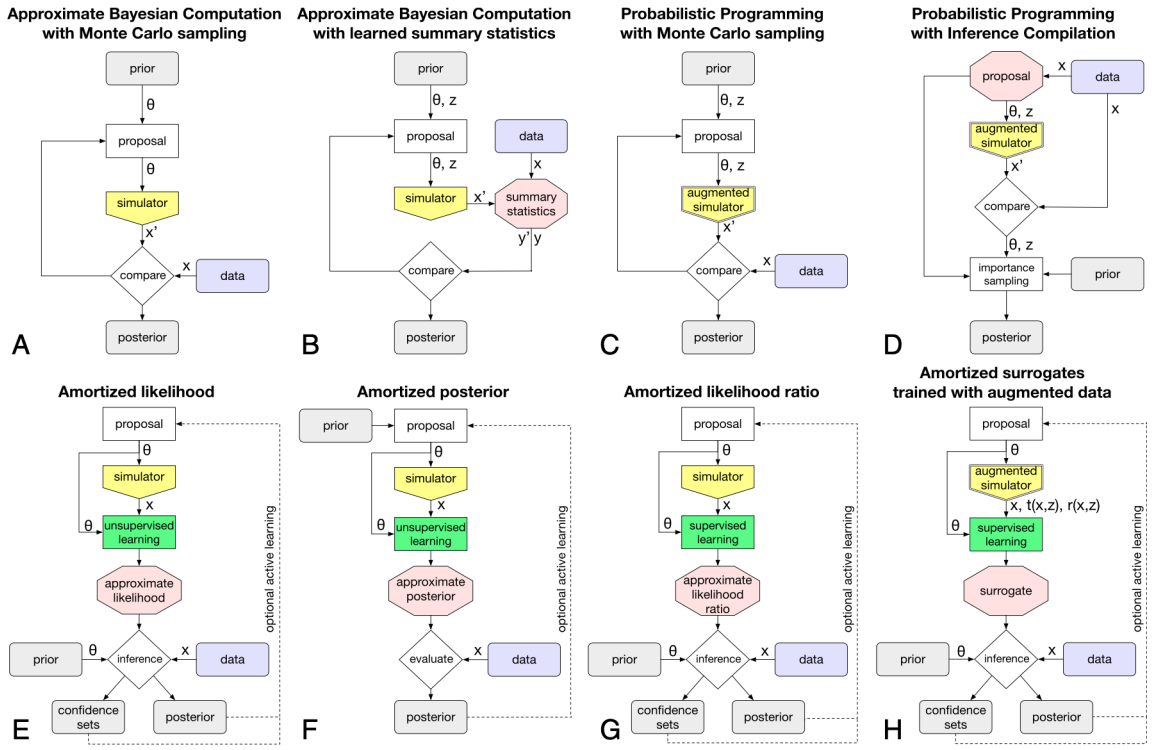

Figure II.1: Overview of different approaches from simulation-based inference. Taken from [9]

II.1.3 Simulation-based inference (SBI) for models without likelihood

Many challenges can be associated to the evaluation of Bayes' formula. Not all generative models admit an analytically tractable likelihood (*i.e.* a closed-form, calculable likelihood). This is for instance the case of continuous time random walks: the generative process is known and well understood, but there are no analytical expressions allowing one to estimate the likelihood of an anomalous exponent given a trajectory: this would require integration over all possible paths from a point to another.

Similar examples are encountered in other scientific fields, notably in epidemiology [8]. The difficulties encountered in the study of such processes, for which it is possible to simulate data but impossible to compute a likelihood, motivated the development of likelihood-free inference, or simulation-based inference, of which [9] provides an overview.

Nonetheless, the relevance of simulation-based methods is not limited to models whose inference is not tractable: they are also helpful when the likelihood is tractable but requires a long computation. We will give an in-depth example of such problem in chapter IV.

A taxonomy of this family of models is shown in figure II.1. Here, we will focus on a few variants of SBI which relate the most to methods used in the later chapters of this work. In SBI, the simulator can be seen as a function f of the parameters and of random hidden variables $\boldsymbol{z}$, from which it generates data in a deterministic way: $\mathbf{x} = f(\boldsymbol{\theta}, \boldsymbol{z})$. Accessing the likelihood of the parameters would require integrating over the whole space of $\boldsymbol{z}$, which is infeasible in real cases when the dimension of $\boldsymbol{z}$ is high.

Simulation-based approaches can be split in two main categories: approximate Bayesian computation and approximate frequentist computation.

a) Approximate Bayesian computation (ABC)

ABC, schematically described in figure II.2 consists in sampling the posterior of parameters $\boldsymbol{\theta}$ by generating data using these parameters and measuring the difference between simulations and actual observations using some informed metric [10, 11]. Sampled parameters are accepted if the difference is lower than a given threshold. In the limit of an infinitely small threshold, ABC becomes exact, but as the probability of accepting a set of parameters vanishes, the required sampling is prohibitively expensive. On the contrary, using a softer criterion will harm the inference's quality, at the benefit of speed. The metric on which the acceptance criterion relies can either be based either directly on observed and simulated data, as in figure II.1A, or on intermediate summary statistics. Resorting to summary statistics is especially relevant, for instance, when doing inference on random walks with ABC: given the importance of randomness in the generative process, two random walks generated using the same model and parameters might have drastically different coordinates, while statistics such as their MSD curves will certainly have more in common. One might indeed think of the MSD as a relevant statistics for a such task, but restricting solely to such hand-picked summaries might harm the quality of inference by discarding some information. Machine learning provides tools to learn summary statistics which are optimal in some sense and can be used for ABC, as depicted in figure II.1B. ABC scales poorly to high-dimensional problems, both because of the rejection process and of the difficulty of finding good proposal distributions (advanced ABC methods include a feedback from the rejection process on the sampling, which helps sampling the most relevant regions of the parameters space). Moreover, the sampling has to be re-done each time a new observation is considered, which makes ABC computationally quite costly.

b) Approximate frequentist computation

Approximate frequentist computation, illustrated in figure II.1E, consists in estimating the likelihood from generated data using histograms or density kernel methods (corresponding to "unsupervised learning" in the figure). Recently, the development of neural conditional density estimators [13, 14] has allowed invertible transformations to be integrated in neural architectures. Parameterized by a summary vector describing the observation on which the inference is to be performed, these mappings, applied to samples drawn from multivariate normal distributions, offer a mean to sample and evaluate complex posterior distributions. An upfront simulation phase is first performed in order to learn the likelihood of the studied system (*i.e.* a function of the parameters and the simulated data, approximated by a neural network). The computational cost of this step scales exponentially with the dimension of the data (or of the summary statistics), just as for ABC. However, contrarily to ABC, approximate frequentist computation handles efficiently the inference of properties for new observations, as the estimation of the likelihood on a new data point can be based on the previously run simulations, there is no need to re-run them. We say that this inference method is *amortized*, as the cost of the initial simulation step is amortized over all subsequent inferences.

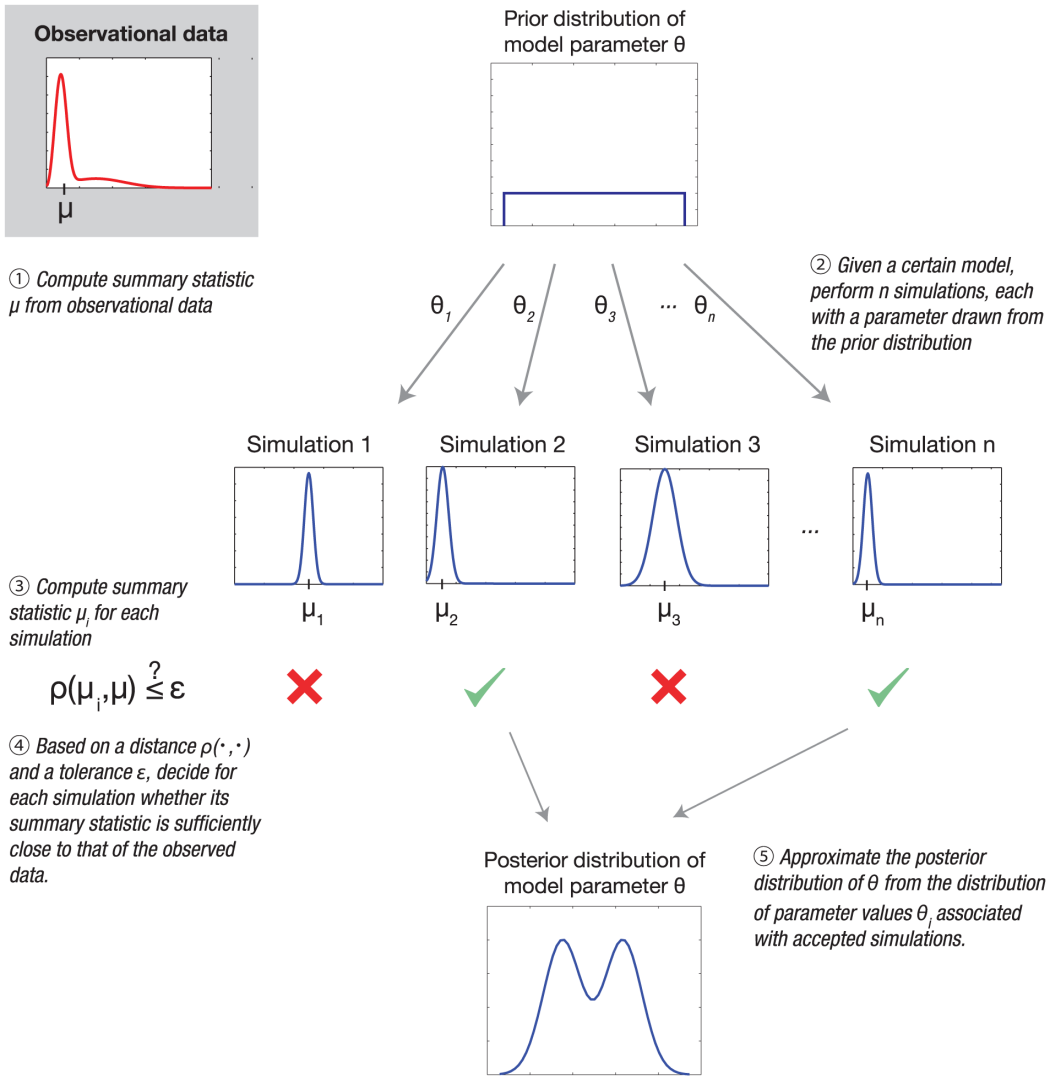

Figure II.2: The principle of approximate Bayesian computation, taken from [12]

II.2 Inferring landscapes of diffusion coefficients and forces

Let us here focus on one approach to analyzing biomolecule dynamics: the mapping of forces and diffusion coefficients based on particle trajectories. In this framework, random walk properties are associated to spatial properties and summarized by local diffusion and forces.

All methods presented in this section model the diffusion dynamics using the overdamped Langevin equation, assuming that the fluctuation-dissipation relation locally holds. Considering for simplicity that the diffusion tensor is isotropic and with $\boldsymbol{\xi}(t)$ being a continuous-time Gaussian white noise process, this equation reads:

$$\frac{d\mathbf{r}(t)}{dt} = \frac{D(r,t)}{k_B T}\mathbf{a}(\mathbf{r},t) + \sqrt{2D(r,t)}\boldsymbol{\xi}(t)$$

, where $D(\mathbf{r},t)$ is the diffusion coefficient and $\mathbf{a}(\mathbf{r},t)$ the drift vector, accounting for both the

force $\mathbf{F}(\mathbf{r},t)$ and effects due to eventual gradients of diffusivity (see [5]). If the additional assumption is made that the force field is conservative, the potential from which it derives can be inferred instead of the force vectors. It follows from the Langevin equation that the likelihood of a jump is

$$p\left(\Delta\mathbf{r}|\mathbf{F}(\mathbf{r},t),D(\mathbf{r},t)\right)=\frac{1}{4\pi D(\mathbf{r},t)\Delta t}\exp\left(-\frac{\left\|\Delta\mathbf{r}-\frac{D(\mathbf{r},t)\Delta t}{k_BT}\mathbf{a}(\mathbf{r},t)\right\|_2^2}{4D(\mathbf{r},t)\Delta t}\right)$$

and, considering that jumps are independent, the likelihood of the set of observations is the product of the likelihoods of all jumps.

Bayesian inference usually handles finite sets of parameters: applying this framework to continuous landscapes requires some adaptation. The three approaches mentioned below have different ways of dealing with this issue.

II.2.1 InferenceMAP & The random walk analyzer (TRamWAy)

Let us first mention the approach proposed by El Beheiry et al., Laurent et al.. There, in order to recover a finite set of parameters, spatio-temporal bins are defined, inside which diffusion and effective potential are assumed constant. The size of these domains needs to be large enough, so that a sufficient number of observations (*i.e.* particle jumps) are present in each bin for the inference to be precise, but small enough to be able to probe the local variations of these values, whose characteristic size might be small. Finally, assumptions about the expected shape of the effective potential and diffusivity landscapes are translated into priors, defined in terms of the difference of potential between neighbor bins (both temporally and spatially). Hence, we have all the elements mentioned in the previous paragraph: a model yielding a likelihood linking the observed jumps to a set of parameters (resulting from a tessellation of the space), as well as priors on these parameters. Assuming that we are not interested in comparing different models, this is sufficient to obtain values of the posterior distribution, up to a normalization factor.

The number of parameters to infer (two per spatio-temporal bin) can be of the order of several thousands if large and dense regions are considered. Getting an estimate of the full posterior distribution is thus very hard, and a more reasonable objective is to focus on estimating the parameters maximizing the posterior: this is referred to as inferring the maximum a posteriori (MAP), and amounts to optimizing the posterior with regard to the parameters, something which can be done using stochastic gradient descent (SGD) methods.

II.2.2 Gaussian processes

Another approach introduced in [17] handles the issue of spatial continuity by resorting to Gaussian processes priors for the force field, with mean zero and covariance $K(x,x')=f(|x-x'|)$. The diffusion coefficient is assumed constant in space, and has a Beta prior. The advantage of this method is that, given that its priors handle under-sampled regions naturally, it overcomes the trade-off between observations density and spatial resolution as well as the artifacts introduced by the discrete binning. It also provides a way of sampling the full posterior, using a Gibbs sampling for the diffusion coefficient. In the publication, values are assumed constant in time, but adding another dimension to allow for time-variations

would not profoundly alter the principle. The main drawback of the method however is its inability to leverage large numbers of observations, as its algorithmic complexity and memory footprint grows proportionally to the cubed number of observations. Thus, its is mainly interesting in the regime where little data is available. The TRamWAy method is more appropriate in cases where data is abundant, firstly because it is faster, and because the artifacts introduced by the discrete binning tend to vanish in the infinite data limit.

II.2.3 A non-Bayesian method: projection on a family of functions

Stepping out of the Bayesian inference framework, Frishman and Ronceray introduces another mean of inferring force fields. Building on an analogy with information transmission via a noisy channel, the authors introduce a way of inferring the force and diffusion fields which also provides a lower bound of the creation of entropy along the process. Fields are inferred by fitting the coefficients of a decomposition of the fields on a finite orthonormal family of functions (such as Fourier or a polynomial functions). This is another mean of restricting the inference to a finite number of parameters. Each jump provides a maximum-likelihood estimate of the fields at its position, and the projections of these estimates on each component of the chosen family of function are averaged in order to obtain a global estimate of the force fields. The advantage of this method is that it comes with interesting relations about the thermodynamics, that it generalizes well to high-dimensional trajectories, and that its complexity scales linearly with the number of observed jumps, not requiring any optimization step. However, results depend on the family of functions chosen to decompose the force field, and it is unclear why one should favor a family over the other – this highlights the interest of the Bayesian framework when it comes to clearly stating assumptions. Moreover, although estimates of the errors are provided, there are no guarantees about the effect of highly irregular landscapes on the estimated fields.

II.2.4 Remarks

Comparing these three inference schemes aimed at addressing relatively similar questions showed important criteria for the choice of a method: the computational complexity, the amount of data required for the method to provide reliable results as well as the possibility to link assumptions to inference results and to control the error (via posterior sampling or theoretical bounds).

In the remaining of this thesis, the spatial criteria is left aside as trajectories are considered irrespective of their localization. Nonetheless, methods introduced in chapter V could well be used to highlight particular properties of given spatial regions and spatial binning tools such as those implemented in TRamWAy could help closing the loop.

II.3 Statistical methods for analysing random walks

II.3.1 MSD-based methods

a) Single trajectories

The most commonly encountered quantity when measuring the diffusion coefficient of a diffusing particle is naturally the mean squared displacement (MSD). For a single Brownian

trajectory of duration $N\Delta t$ whose positions r_i along a single dimension axis are measured at intervals Δt, the slope of the curve of $\rho_n = \frac{1}{N-n+1}\sum_i (r_{i+n} - r_i)^2$ is indicative of the diffusion coefficient D. ρ_n is known in the literature as the time-averaged MSD (TAMSD). However, due to the correlation between displacements of length n measured from overlapping segments of the trajectory, its variance grows rapidly with n as demonstrated in [19]:

$$\mathrm{Var}(\rho_n) = \frac{(2Dn\Delta t)^2(2n^2+1)}{3n(N-n+1)}$$

The estimator $\hat{D}_n = \rho_n/2n\Delta t$ thus has a squared relative error of the order of

$$\mathrm{Var}(\hat{D}_n/D) \approx \frac{2n}{3(N-n+1)}.$$

For short trajectories, the relative error is thus very high, even when $n \ll N$: about 20% when $n = 1, N = 10$ and 10% when $N = 50$. Depending on the time scale of the correlations between jumps, it might be needed to base the estimator on larger time intervals, increasing the level of relative error. Moreover, if one is not only interested in the diffusion coefficient but also in the anomalous diffusion exponent, scaling properties of ρ_n at large n must be estimated.

More generally, [20] study the biases of the estimation of D_α and α from fits of log-TAMSD curves, *i.e.* when they are determined by a least square fit of the curve of $\log(\rho_n) = \log(D_\alpha) + \alpha \log(n)$. They demonstrate that the resulting estimation of α depends on the range of considered values of n, as exemplified in figure II.3. They confirm the effect of localization error on estimations of α and highlight the importance of taking it into account (and of estimating it beforehand). An estimation of the negative bias induced by localization errors is introduced by Kepten et al.. Finally, regarding short and noisy trajectories (shorter than 100 localizations), one might quote the conclusion of Lanoiselée et al.: "for short trajectories the TAMSD is not appropriate".

Altogether, this confirms the intuition that if one's aim is to extract information about diffusion from a single trajectory, it should rather be from a long one. If no sufficiently long trajectories are available, then a good option might be to resort to ensemble methods.

b) Ensemble methods

Beyond individual trajectory analysis, information can be gained from the aggregation of several trajectories, if such observations are available. In general, this however implies that additional assumptions are made that all particles share the same physical properties (and hence, that estimates deriving from single trajectories can be considered as independent samples from a same probability distribution). Although it is possible, if assumptions are made about the underlying distributions of α and D_α, to gain insight on the average α from TAMSDs based on ensembles of trajectories (see [21]), this is in general not the case. Indeed, if particles have different properties, superdiffusive dynamics will tend have a predominant effect on the TAMSD at long time lags while subdiffusive will dominate the short time lags. In the case where walkers are assumed to share the same α and D_α, [22] introduce several estimators based on various fitting procedures of the MSD, some being tractable and others requiring a numerical optimization step. Such fits are used in [23] to measure with precision the exponent α. It is interesting to remark that this work mentions the slight

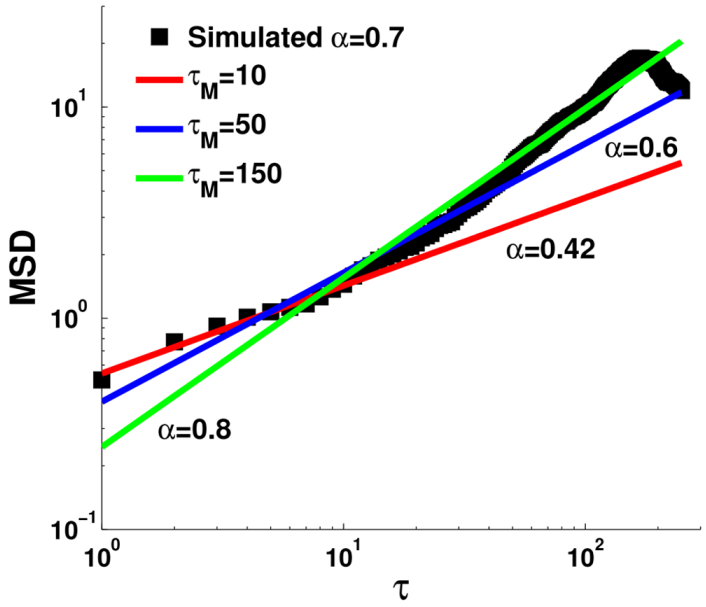

Figure II.3: Estimations of the anomalous exponent α from a single trajectory of length 300, based on different ranges of TAMSD, taken from [20]

(but significant) difference between the anomalous exponent measured at long and short time scales. In addition, comparing TAMSD and ensemble-averaged TAMSD might be interesting to pinpoint the existence of weak ergodicity breaking, as in [24].

II.3.2 Distinguish diffusion states with hidden Markov models (HMMs)

Up to this point, we have only discussed inference methods assuming constant dynamics for the entire trajectories. Yet, long trajectories can sample portions of the cells with different physical properties or the biomolecule can undergo change in conformation changing the natures of its interaction with its environment. This motivated development of inference schemes aimed at quantifying the transition dynamics between these modes, as well as the diffusion properties of each of these modes. Many of these approaches resort to hidden Markov models (HMMs), which we will introduce briefly prior to presenting a selection of multi-state inference methods.

a) Hidden Markov models

Let us consider a system evolving over N time steps between K possible states. The sequence of its states is $s_1, s_2, \ldots, s_N$, with $s_i \in \{s^{(1)}, s^{(2)}, \ldots, s^{(K)}\}$. This system is said to be Markovian if the probability of transitioning from one state to another only depends on the current state, and not of the previous ones. A such process is thus fully characterized by the vector $(\pi_1, \ldots, \pi_K)$ the probabilities of the initial states, and the transition matrix whose coefficients $p_{i\to j}$, are the probabilities of transitioning from state i to state j. In a hidden Markov model, states are not observed directly, they are *hidden* and one only accesses observations $(y^{(1)}, y^{(2)}, \ldots, y^{(N)})$ conditioned by the hidden state: $y^{(n)} \sim p(\cdot|s^{(n)})$. In the context of SPT experiments, the states are the particles' diffusion modes, while the observations are

the actual recorded trajectory jumps. Fitting such models to trajectories requires determining the actual number of states, the transition probabilities from one state to another, and the physical properties driving the diffusion in each such state. The initial probabilities can also be inferred if it is not assumed that the system is observed at equilibrium. Once these parameters are known, determining the most plausible states visited by a particle from a single trajectory, *i.e.* to segment the trajectory according to the hidden states, can be done using the Viterbi algorithm [25].

b) A two-state model

Here, we present the model introduced in [26]. It assumes that a particle randomly switches between two modes of Brownian diffusion, defined each by a diffusion coefficient. The model thus has four parameters: two diffusion coefficient and two transition probabilities $(D_1, D_2, p_{1\to2}, p_{2\to1})$. As it is considered to be at equilibrium, and thus initial probabilities can be confounded with equilibrium ones, determined by the transition rates. The log-likelihood of the parameters given an observed trajectory can be expressed using the probabilities of being at state i at step n given the data, which are computed recursively. Then, the posterior is estimated using a Monte Carlo Markov Chain, with an initialization step allowing the parameters to converge towards the MAP. Posteriors of different trajectory can be multiplied to aggregate information stemming from different observations. Compared to simpler methods applied to a such two-mode diffusion [27, 28, 29], based on fits of a bimodal distribution on the MSD histogram (as illustrated in figure II.4), this approach allows one to access the rates of transition between the two modes, and to estimates more precisely the characteristics of each mode.

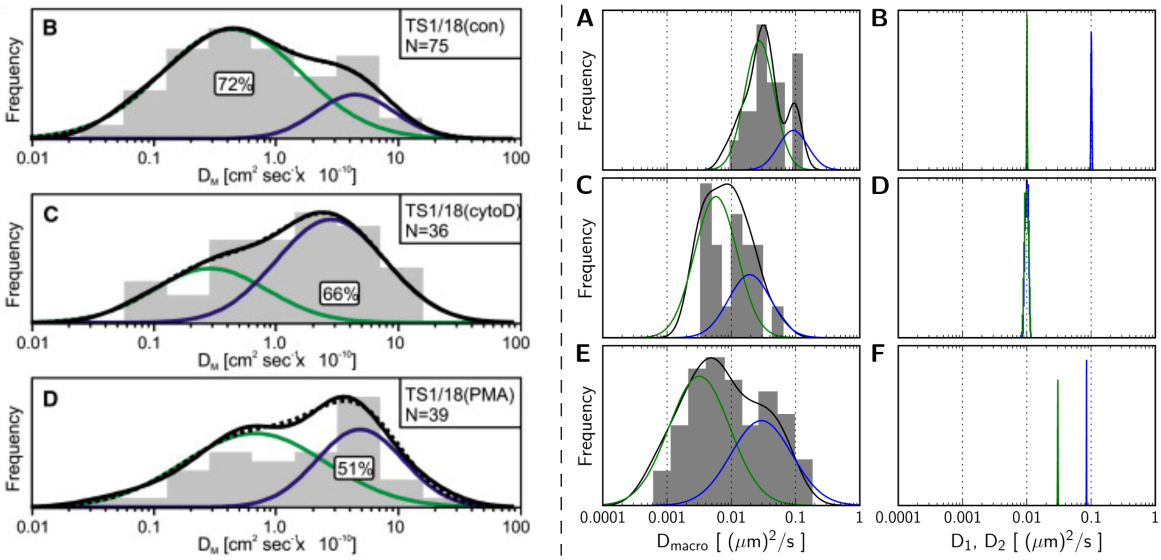

Figure II.4: Illustration of results obtained with Gaussian fitting of the apparent diffusion histogram (left) and with the two-state HMM method (right). The HMM yields posterior distributions of the two diffusion coefficients as well as transition rates. Adapted from [27] and [26]

c) Unknown number of unknown states

Although the previously introduced two-state model could be extended to an arbitrary number of diffusion states, one may have no reason to make the assumption of one precise number of hidden states. A more complete model is introduced in [30] to address this issue. It encompasses diffusion modes which can be either pure Brownian motion or directed Brownian motion, in which case they are parameterized not only by a diffusion coefficient D_i but also by a drift vector $\mathbf{v}_i$. The number K of different states is not fixed beforehand, only an upper limit $K_{\max}$ is set. As the fraction of diffusion modes with non-zero drift is not known either, there are $K+1$ possible models with K steps. Hence, there are $(K_{\max}^2 + 3K_{\max})/2$ alternative models with $K \leq K_{\max}$. These models are compared *via* their evidences, which are computed using a Metropolis Markov chain Monte Carlo sampling algorithm. This inference scheme has the advantage over the previously cited one that it does not force the detection of several modes. Moreover, it accounts for a wider diversity of modes (directed motion and pure diffusion). The authors demonstrate its relevance on several examples of biomolecules diffusing in cells. We remark that on the examples they provide, the number of states selected by the model rarely exceeds two or three.

Another approach, similar to this one because it also compares models based on a range of numbers of states using their evidence, was proposed earlier by Persson et al.. It was restricted to normal diffusion however, but had the interest of using a mean-field approximation to circumvent the computational burden of Monte Carlo samplings. The approximation amounts to considering the diffusion parameters and the series of hidden states as independent, facilitating the optimization of the likelihood.

d) Unparametric Gibbs sampling

Finally, we mention the model recently presented in [32], as it notably has the conceptual advantage of not requiring from the user to input a maximal number of considered states. This method is inspired by the approach introduced in [33] for the analysis of time-series coming from single molecule observations. Instead of sampling the posteriors of a collection of models defined by various sets of parameters, a single iterative sampling scheme is performed, varying the number of considered states across iteration. Thus, computing resources might be more efficiently allocated, as extensive samplings of irrelevant models are avoided. Each step of the iteration can be summarized as follows: first, the number of states is set using a Dirichlet Process, reduced to a finite number of states by discarding those with the smallest weights. Then, a Gibbs sampling method similar to the one used in the HMM model is used to get posteriors of the modes' parameters. When new states are inserted, their properties are drawn from a prior distribution, those of the remaining states are sampled from the last iteration's posterior distribution. This runs until a convergence criterion is reached.

e) Remarks

HMMs have the advantage of providing an intuitive interpretation of the dynamics, as transition probabilities between discrete diffusion states are rather telling notions. However, most HMM-based methods have the inconvenient of requiring costly sampling steps, which have to be re-run every time a new data point is added. Moreover, these models require that a tractable likelihood of the diffusion mode is available and are thus restricted to a limited set

of random walk models. Furthermore, there might exist cases where the diffusion properties vary continuously and not discretely. In such cases, HMM-based schemes might suggest a high number of diffusion modes, or a very broad mode.

II.3.3 Leveraging hand-picked features

In the following, we introduce a selection of random walks characterization techniques focused on specific mathematical or physical properties, along with their application and the type of dynamics that they allow one to distinguish.

a) Directional changes

In Burov et al., the distribution of directional changes is shown to be a signature of different types of diffusion dynamics, complementary of the MSD curve. It notably provides insights on the eventual anisotropy of the diffusion, or the existence of a drift, as such phenomenon would favor either small changes of the direction or U-turns. At each recorded position of the trajectory, the angle between consecutive jumps is accessed via the sine and cosine of the jumps. Hence, this metric can be computed regardless of the dimension (as long as it is greater or equal to two)

b) Mean maximal excursion (MME)

Studies of the mean maximum excursion (MME) of random walks on fractals and of CTRWs have motivated [35] to demonstrate this quantity's relevance for quantifying the anomalous diffusion exponent of an ensemble of trajectories assumed to have the same diffusion dynamics. For a trajectory, the maximum excursion at a given time t is the maximum distance $r_{max}(t)$ from the origin reached by the particle up to this point. It was shown that for CTRWs and diffusion on fractals, the second moment of this quantity, $\langle r_{max}(t)^2 \rangle$, scales with t^{α}. Authors show that fits of the MME curve yield more precise estimations of α than those of the MSD. Besides, the authors study the property of the ratios, both for the MSD and the MME, of the fourth moment and the squared second moment. Along with the probability of being in a sphere centered at the first position of the trajectory and whose radius grows like $t^{\alpha/2}$, these constitute a set of quantities which can allow one to distinguish between various random dynamics (CTRW, fBM, diffusion on fractals).

c) First passage observables

Interested in studying the kinetics of diffusion-limited chemical reactions, [36] compared the properties of several first passage observables resulting from CTRW and diffusion on fractals. Among other quantities, they study the distribution and first moment of the first passage time (FPT, time taken to reach a distance r from the origin), and analytically show that the mean FPT scales differently for CTRWs, which explore the space non-compactly, than for diffusion on fractals which is much slower. These observables thus allow to discriminate between these two concurrent models of diffusion.

d) Remarks

The approaches mentioned here, while relevant for the insights they provide about defined features of the trajectories, share several flaws which hinder, for now, their applicability to a range of experimentally recorded trajectories. Indeed, FPT and mean maximal excursions rely on asymptotic properties of the walks and are thus not suited to the analysis of short recordings such as those provided by SPT-PALM experiments. Moreover, they assume a single mode of diffusion for the analyzed particles while biomolecule often do not diffuse for very long in cells before transitioning between regions with different physical properties, or undergoing a state change of some sort (oligomerization, change of three-dimensional conformation...). Hence, both technical and biological constraints limit the set of problems to which these methods can be applied. Their interest in fact mostly lies in the understanding that they provide of how the motion dynamics of a walker govern its biological function. Besides, the distribution of directional changes does not require long trajectories, but is hard to interpret on its own and will mostly serve to compare sets of trajectories observed under different conditions, or to recover relaxation times. All these methods lack a quantification of likelihood of observing a given level of discrepancy between two measurements: to use them to quantify the statistical significance of these discrepancies, one has to resort to common sense, or (better) to simulations in order to interpret non-overlapping curves. The issue of probing the statistical significance of measured differences between observables derived from trajectories of random walks is addressed by the method we introduce in chapter V.

II.4 Methods based on random forests and neural networks

A last class of inference methods rely on supervised learning methods to characterize parameters of the diffusion. Prior to introducing a few examples of application of these techniques to the study of diffusion, we rapidly summarize some concepts associated to machine learning.

II.4.1 A very short introduction to supervised machine learning

The task addressed by supervised machine learning is that of *learning* a mapping between a space of observations (images, text, sound, ...) and a space of predictions [37]. The mapping belongs to a given class of functions, which depends on the type of algorithm used. Denoted $f_{\boldsymbol{\theta}}$, it is parameterized by a vector $\boldsymbol{\theta}$: simple models have only a handful of parameters, while most advanced neural architectures can have billions. In a supervised setting, a collection of samples $\mathbf{x}$ is available, along with the corresponding expected predictions $\mathbf{y}$.

During the learning (or "training") phase, $\boldsymbol{\theta}$ is set so as to minimize a loss term, which measures the discrepancy between $\hat{\mathbf{y}} = f_{\boldsymbol{\theta}}(\mathbf{x})$ and $\mathbf{y}$. In some cases, the loss might as well depend on $\boldsymbol{\theta}$ itself, so as to penalize undesired types of mappings. For instance, when learning a mapping between two Euclidian spaces, one might want to impose regularity constraints and therefore prevent too steep variations by penalizing exceedingly large values of coefficients of $\boldsymbol{\theta}$. Although some learning algorithms provide a closed form expression of $\boldsymbol{\theta}$ given the data, most learning procedures are iterative optimizations.

The fact that, given a sufficient amount of data, such predictors can be automatically fitted, is interesting for many problems where defining explicit rules on which a prediction should be based is a difficult task. Image recognition, which has driven lots of the recent

advances in neural networks, is one such class of problems: expressing in mathematical terms the difference between a picture of a dog and one of a cat can indeed be quite challenging.

Nonetheless, for a such mapping to be relevant, it does not suffice that it makes accurate predictions on the training data: it should also provide a sensible interpolation between the training examples, so as to be used on novel samples. This is especially complex when the dimension of the data is high, because the manifold on which the data lies is never extensively sampled in the training set. Hence, the difficulty of machine learning mostly lies in finding algorithms capable of *generalizing* outside of the training data. Several factors influence this property, notably the architecture of the mapping (the class of function to which it belongs), the pre-processing of the data and the optimization procedure [3].

Before the improvements of hardware, which made possible the rapid training of models counting millions of parameters, the rise of performance of machine learning (and more specifically deep learning [38]) on complex tasks was first driven by the development of predictors architectures able to capture symmetries of the data, hence reducing the complexity of the interpolation task [39].

There exist numerous families of predictors, whose complexity vary between the simple linear regressions and the large transformer neural networks counting billions of parameters. As important as the structure of the predictor itself is the input that it is given: does the predictor access the raw observations, are they pre-processed, or does it only deal with a determined set of features computed from the observations ? Resorting to features can help the model to better generalize, both because it often reduces the dimensionality of the input and because informed features wipe out irrelevant degrees of freedom (for instance, they might respect some symmetries). On the opposite, the risk is to discard information, present in the data but not captured by the hand-picked features. The advantage of neural network is that they are able, to some extent, to learn features relevant to the prediction process directly from the raw data, overcoming this pitfall.

II.4.2 Machine learning for random walk analysis

Predictors obtained via machine learning methods have the advantage over analytical estimators (such as those introduced in section II.3) of being more versatile. Indeed, they are not restricted to a subset of model random walks: as long as they have been trained on the right data, they should be able to output a relevant measure. Moreover, they can provide estimators for which no analytical formula has been developed, such as changes of anomalous diffusion exponent. Predictors are trained on trajectories simulated using a variety of random walk model and parameter values, depending on their purpose. Indeed, we can only know with certainty the *ground truth* values of underlying parameters for numerically simulated trajectories: such labels do not exist for recorded trajectories.

Relying on simulations for training lifts the constraints induced by the limited amount of available data which most machine learning use cases have, but simulations imply assuming the generative model used within these simulations to be close enough from the observed dynamics. If an artifact in the experimental protocol perturbs the observations, (localization noise, for instance), it might fool the predictors. But if this artifact is anticipated and included in the simulation scheme used during training, the predictors can learn to account for it. This ability to handle a variety of known artifact is another advantage of machine learning methods over conventional analytical estimators. Finally, methods relying on ma-

chine learning belong to the class of amortized inference: after an expensive training phase, the inference of new observations is almost instantaneous. This is a great advantage over all methods requiring long sampling procedures for each new data point.

II.4.3 Methods based on random forests

We will not introduce random forests in details here, as we will mostly state the properties of these predictors which matter to their use for random walk analysis. We refer the interested reader to [40] for an introduction to random forests and a presentation of some recent developments.

Random forests are reckoned as predictors of choice when dealing with tabular data, *i.e.* data lacking geometric structure or symmetries. They accept both continuous and discrete inputs, and provide more direct insights than neural networks about the relative of importance that they assign to each input variable. Random forest thus must resort to trajectory-level features to handle trajectories of various sizes as they only handle a fixed size of input vectors.

One of the first studies of this type, [41] proposes a trajectory classifier based on nine features, chosen each for their ability to capture a given aspect of random dynamics. The classifier is able to distinguish between anomalous diffusion, Brownian motion, directed motion and confined diffusion. The drop in performance caused by the removal of each of the features from the model, allows one to probe their relative importance in the prediction. Recently, [42] explored in more detail the contribution of each feature and the effect of various parameters of the random forest on its performance. Taking another approach, [43] directly inputs trajectory positions into random forests, to predict both the anomalous exponent and the type of random walk. It reaches a good performance but has the inconvenient of requiring a different predictor for each trajectory length.

II.4.4 Methods based on neural networks

Over the last decades, complex architectures of neural networks were developed, of which many share a same building block: the multi-layer perceptron (MLP). It is a succession of steps (called "layers") sequentially performing the following operation on an input vector $\mathbf{x}$: $l_{\mathbf{W},\mathbf{b}}(\mathbf{x}) = g(\mathbf{W}\mathbf{x} + \mathbf{b})$, where g is a non-linear function and the parameters of l are a weight matrix $\mathbf{W}$ and a bias vector $\mathbf{b}$. MLPs, like random forest, are restricted to fixed-size input vectors. They were used by [44] on MSD curves computed on windows of the trajectories, in order to extract the diffusion mode (among three options) of the corresponding portion of the trajectory.

Recurrent neural networks (RNN) were developed to analyze data with a sequential structure (they are very much used in natural language processing). In [45], they were used to measure the anomalous diffusion exponent of fBM trajectories, as well as to detect changes of anomalous diffusion exponent. The authors leveraged the flexibility offered by the use of synthetic training data to handle trajectories with irregularly sampled observations (that is, with a varying time interval between locations, or with missing observations). They observe, as one could expect, that their approach only outperforms conventional MSD-based methods on short trajectories. Most of the best-performing models of the AnDi challenge [46], a competition aiming to objectively compare random walk analysis methods, were based

on RNNs (see for example [47]). Good results were notably obtained by using convolution layers such as those used for image and sound analysis yield in conjunction with a RNN architecture [48].

Interestingly, deep learning has sometimes been coupled with analytical methods of random walk characterization, such as in [49], where an RNN first segments tracks in tracklets assumed to have a constant dynamics, which are then analysed using the moment scaling method, an analytically computed metric providing a hint about whether trajectories are confined or diffusing freely.

II.4.5 Limitations of machine learning

As machine and deep learning methods almost all rely (linear regression being one of the few exceptions) on intractable learning rules, they are notably difficult to dissect. Indeed, the resulting predictor depends simultaneously on the data, the structure of the network and the learning rule, in a very intricate manner. Hence, although results obtained on numerically generated data often overcome those of conventional estimators, one is often left with conjectures and intuitions when it comes to ensuring the capability of a machine-learning predictor to generalize on experimental data. Statistical methods such as the maximum mean discrepancy presented in chapter V are interesting in this regard, as they allow one to take advantage of representations of trajectories learnt by neural networks with more flexibility and way for interpretation than when only values predicted by the networks are considered.

Bibliography

[1] Antony Lee, Konstantinos Tsekouras, Christopher Calderon, Carlos Bustamante, and Steve Pressé. Unraveling the thousand word picture: an introduction to super-resolution data analysis. *Chemical reviews*, 117(11):7276–7330, 2017.

[2] Udo Von Toussaint. Bayesian inference in physics. *Reviews of Modern Physics*, 83(3):943, 2011.

[3] David JC MacKay et al. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[4] Thomas M Cover and Joy A Thomas. Information theory and statistics. *Elements of information theory*, 1(1):279–335, 1991.

[5] Alexander S Serov, François Laurent, Charlotte Floderer, Karen Perronet, Cyril Favard, Delphine Muriaux, Nathalie Westbrook, Christian L Vestergaard, and Jean-Baptiste Masson. Statistical tests for force inference in heterogeneous environments. *Scientific Reports*, 10(1):1–12, 2020.

[6] Jens Krog, Lars H Jacobsen, Frederik W Lund, Daniel Wustner, and Michael A Lomholt. Bayesian model selection with fractional brownian motion. *Journal of Statistical Mechanics: Theory and Experiment*, 2018(9), September 2018. doi: 10.1088/1742-5468/aadb0e. URL https://doi.org/10.1088/1742-5468/aadb0e.

[7] Christopher Bishop. *Pattern recognition and machine learning*. Springer, New York, 2006. ISBN 978-0387310732.

[8] Steven Riley, Christophe Fraser, Christl A Donnelly, Azra C Ghani, Laith J Abu-Raddad, Anthony J Hedley, Gabriel M Leung, Lai-Ming Ho, Tai-Hing Lam, Thuan Q Thach, et al. Transmission dynamics of the etiological agent of sars in hong kong: impact of public health interventions. *Science*, 300(5627): 1961–1966, 2003.

[9] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

[10] Donald B Rubin. Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, pages 1151–1172, 1984.

[11] Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

[12] Mikael Sunnraker, Alberto Giovanni Busetto, Elina Numminen, Jukka Corander, Matthieu Foll, and Christophe Dessimoz. Approximate bayesian computation. *PLoS computational biology*, 9(1):e1002803, 2013.

[13] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[14] Stefan T Radev, Ulf K Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Köthe. Bayesflow: Learning complex stochastic models with invertible neural networks. *IEEE transactions on neural networks and learning systems*, 2020.

[15] Mohamed El Beheiry, Maxime Dahan, and Jean-Baptiste Masson. InferenceMAP: Mapping of single-molecule dynamics with Bayesian inference. *Nature Methods*, 12(7):594–595. ISSN 1548-7091. doi: 10.1038/nmeth.3441.

[16] François Laurent, Charlotte Floderer, Cyril Favard, Delphine Muriaux, Jean-Baptiste Masson, and Christian L Vestergaard. Mapping spatio-temporal dynamics of single biomolecules in living cells. *Physical Biology*, 17(1):015003, November 2019. doi: 10.1088/1478-3975/ab5167. URL https://doi.org/10.1088/1478-3975/ab5167.

[17] J Shepard Bryan IV, Ioannis Sgouralis, and Steve Pressé. Inferring effective forces for langevin dynamics using gaussian processes. *The Journal of Chemical Physics*, 152(12):124106, 2020.

[18] Anna Frishman and Pierre Ronceray. Learning force fields from stochastic trajectories. *Physical Review X*, 10(2):021009, 2020.

[19] 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):910–921, October 1991. doi: 10.1016/s0006-3495(91)82125-7. URL https://doi.org/10.1016/s0006-3495(91)82125-7.

[20] Eldad Kepten, Aleksander Weron, Grzegorz Sikora, Krzysztof Burnecki, and Yuval Garini. Guidelines for the fitting of anomalous diffusion mean square displacement graphs from single particle tracking experiments. *PLOS ONE*, 10(2):e0117722, February 2015. doi: 10.1371/journal.pone.0117722. URL https://doi.org/10.1371/journal.pone.0117722.

[21] Eldad Kepten, Irena Bronshtein, and Yuval Garini. Improved estimation of anomalous diffusion exponents in single-particle tracking experiments. *Physical Review E*, 87(5):052713, 2013.

[22] Yann Lanoiselée, Grzegorz Sikora, Aleksandra Grzesiek, Denis S Grebenkov, and Agnieszka Wyłomańska. Optimal parameters for anomalous-diffusion-exponent estimation from noisy data. *Physical Review E*, 98(6):062139, 2018.

[23] Iva Marija Tolić-Nørrelykke, Emilia-Laura Munteanu, Genevieve Thon, Lene Oddershede, and Kirstine Berg-Sørensen. Anomalous diffusion in living yeast cells. *Physical Review Letters*, 93(7):078102, 2004.

[24] Carlo Manzo, Juan A Torreno-Pina, Pietro Massignan, Gerald J Lapeyre Jr, Maciej Lewenstein, and Maria F Garcia Parajo. Weak ergodicity breaking of receptor motion in living cells stemming from random diffusivity. *Physical Review X*, 5(1):011021, 2015.

[25] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269, 1967.

[26] Raigraphk Das, Christopher W. Cairo, and Daniel Coombs. A hidden markov model for single particle tracks quantifies dynamic interactions between LFA-1 and the actin cytoskeleton. *PLoS Computational Biology*, 5(11):e1000556, November 2009. doi: 10.1371/journal.pcbi.1000556. URL https://doi.org/10.1371/journal.pcbi.1000556.

[27] Christopher W Cairo, Rossen Mirchev, and David E Golan. Cytoskeletal regulation couples lfa-1 conformational changes to receptor lateral mobility and clustering. *Immunity*, 25(2):297–308, 2006.

[28] Gerhard J Schütz, Hansgeorg Schindler, and Thomas Schmidt. Single-molecule microscopy on model membranes reveals anomalous diffusion. *Biophysical journal*, 73(2):1073–1080, 1997.

[29] Jiji Chen, Zhengjian Zhang, Li Li, Bi-Chang Chen, Andrey Revyakin, Bassam Hajj, Wesley Legant, Maxime Dahan, Timothée Lionnet, Eric Betzig, et al. Single-molecule dynamics of enhanceosome assembly in embryonic stem cells. *Cell*, 156(6):1274–1285, 2014.

[30] Nilah Monnier, Zachary Barry, Hye Yoon Park, Kuan-Chung Su, Zachary Katz, Brian P English, Arkajit Dey, Keyao Pan, Iain M Cheeseman, Robert H Singer, et al. Inferring transient particle transport dynamics in live cells. *Nature methods*, 12(9):838–840, 2015.

[31] Fredrik Persson, Martin Lindén, Cecilia Unoson, and Johan Elf. Extracting intracellular diffusive states and transition rates from single-molecule tracking data. *Nature methods*, 10(3):265–269, 2013.

[32] Joshua D Karslake, Eric D Donarski, Sarah A Shelby, Lucas M Demey, Victor J DiRita, Sarah L Veatch, and Julie S Biteen. Smaug: Analyzing single-molecule tracks with nonparametric bayesian statistics. *Methods*, 193:16–26, 2021.

[33] Ioannis Sgouralis and Steve Pressé. An introduction to infinite HMMs for single-molecule data analysis. *Biophysical Journal*, 112(10):2021–2029, May 2017. doi: 10.1016/j.bpj.2017.04.027. URL https://doi.org/10.1016/j.bpj.2017.04.027.

[34] S. Burov, S. M. A. Tabei, T. Huynh, M. P. Murrell, L. H. Philipson, S. A. Rice, M. L. Gardel, N. F. Scherer, and A. R. Dinner. Distribution of directional change as a signature of complex dynamics. *Proceedings of the National Academy of Sciences*, 110(49):19689–19694, November 2013. doi: 10.1073/pnas.1319473110. URL https://doi.org/10.1073/pnas.1319473110.

[35] Vincent Tejedor, Olivier Bénichou, Raphael Voituriez, Ralf Jungmann, Friedrich Simmel, Christine Selhuber-Unkel, Lene B Oddershede, and Ralf Metzler. Quantitative analysis of single particle trajectories: mean maximal excursion method. *Biophysical journal*, 98(7):1364–1372, 2010.

[36] S Condamin, Vincent Tejedor, Raphaël Voituriez, Olivier Bénichou, and Joseph Klafter. Probing microscopic origins of confined subdiffusion by first-passage observables. *Proceedings of the National Academy of Sciences*, 105(15):5675–5680, 2008.

[37] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[38] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[39] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[40] Adele Cutler, D Richard Cutler, and John R Stevens. Random forests. In *Ensemble machine learning*, pages 157–175. Springer, 2012.

[41] Thorsten Wagner, Alexandra Kroll, Chandrashekara R. Haramagatti, Hans-Gerd Lipinski, and Martin Wiemann. Classification and segmentation of nanoparticle diffusion trajectories in cellular micro environments. *PLOS ONE*, 12(1):e0170165, January 2017. doi: 10.1371/journal.pone.0170165. URL https://doi.org/10.1371/journal.pone.0170165.

[42] Patrycja Kowalek, Hanna Loch-Olszewska, Janusz Szwabi, et al. Boosting the performance of anomalous diffusion classifiers with the proper choice of features. *Journal of Physics A: Mathematical and Theoretical*, 2022.

[43] Gorka Muñoz-Gil, Miguel Angel Garcia-March, Carlo Manzo, José D Martín-Guerrero, and Maciej Lewenstein. Single trajectory characterization via machine learning. *New Journal of Physics*, 22(1): 013010, 2020.

[44] Patrice Dosset, Patrice Rassam, Laurent Fernandez, Cedric Espenel, Eric Rubinstein, Emmanuel Margeat, and Pierre-Emmanuel Milhiet. Automatic detection of diffusion modes within biological membranes using back-propagation neural network. *BMC Bioinformatics*, 17(1), May 2016. doi: 10.1186/s12859-016-1064-z. URL https://doi.org/10.1186/s12859-016-1064-z.

[45] Stefano Bo, Falko Schmidt, Ralf Eichhorn, and Giovanni Volpe. Measurement of anomalous diffusion using recurrent neural networks. *Physical Review E*, 100(1):010102, 2019.

[46] Gorka Muñoz-Gil, Giovanni Volpe, Miguel Angel Garcia-March, Erez Aghion, Aykut Argun, Chang Beom Hong, Tom Bland, Stefano Bo, J Alberto Conejero, Nicolás Firbas, et al. Objective comparison of methods to decode anomalous diffusion. *Nature communications*, 12(1):1–16, 2021.

[47] Aykut Argun, Giovanni Volpe, and Stefano Bo. Classification, inference and segmentation of anomalous diffusion with recurrent neural networks. *Journal of Physics A: Mathematical and Theoretical*, 54(29): 294003, 2021.

[48] Dezhong Li, Qiujin Yao, and Zihan Huang. Wavenet-based deep neural networks for the characterization of anomalous diffusion (wadnet). *Journal of Physics A: Mathematical and Theoretical*, 54(40):404003, 2021.

[49] Marloes Arts, Ihor Smal, Maarten W. Paul, Claire Wyman, and Erik Meijering. Particle mobility analysis using deep learning and the moment scaling spectrum. *Scientific Reports*, 9(1), November 2019. doi: 10.1038/s41598-019-53663-8. URL https://doi.org/10.1038/s41598-019-53663-8.

III – Amortized inference of random walk properties with graph neural networks

This chapter's content has been adapted from the following publication. Results are displayed as published although some improvements of the methods have been developed since then. These improvements are discussed in section III.6

Verdier Hippolyte, Duval Maxime, Laurent François, Cassé Alhassan, Vestergaard Christian L, and Masson Jean-Baptiste. Learning physical properties of anomalous random walks using graph neural networks. *Journal of Physics A: Mathematical and Theoretical*, 54(23): 234001, 2021

Contents

A variety of archetypal random walk processes were introduced to cover the diversity of dynamics observed in physical or biological systems: jumps can be continuous or discrete, exhibit ageing or obey a stationary distribution, etc... For a large class of such processes, the anomalous diffusion exponent α is defined asymptotically (*i.e.* for long trajectories). While this exponent is easily estimated when one disposes of a large number of long trajectories (using previously introduced TA-MSD curves), inferring the value of α of a such process from a single one of its realisation can be challenging, all the more so when the type of random walk model is not known in advance. In this chapter, we introduce a method addressing the inverse problem consisting in inferring both the anomalous diffusion exponent and the type of random walk (among a dictionary of five).

Most analytical estimators of α do not work well for short trajectories, or when positions are measured with uncertainty. Furthermore, ad-hoc features such as those mentioned in II.3.3 do not allow to simultaneously distinguish more than two or three types of walks. Thus, in order to cover a broad diversity of random walk models and to account for positioning noise, we chose to develop a simulation-based inference approach. One originality of the method lies in the use of graph neural networks (GNNs), which we adapted to handle trajectories. We start by introducing GNNs and motivating or choice of architecture. Then, we demonstrate the performance of the method on simulated trajectories. We highlight the structure of the latent space learnt by the network, which is informative of the distinctions that it is able to make. We end the chapter by exploring a few perspectives of improvement over the published version of the model, some of which are used in chapters IV and V.

III.1 Graph neural networks on random walks

We detail below the methodology developed to leverage GNN's flexibility and their capacity for representation learning [2] in order to infer properties of random walks, in particular their anomalous exponent and model class.

First, a graph has to be associated with each trajectory for it to be processed by a GNN (see Figure III.1 and paragraph III.1.2). The actual inference is performed by a GNN, capable of processing graphs of variable size, and which outputs an estimate of the anomalous exponent and a probability of belonging to each random walk model class. The weights of this neural network are set during a training phase (on numerically generated trajectories) similarly to conventional supervised learning schemes. The network can later be used for inference on other trajectories, not seen during training.

III.1.1 Rationale for learning random walks with graph neural networks

The use of a graphical representation has long been a method of choice to model and perform inferences of complex systems [3]. For example, factor graphs associated to mean field, belief propagation [4] and cavity methods [5] have been used to model spin glass dynamics and perform complex optimisation problems. Hidden Markov models [6] have been developed to

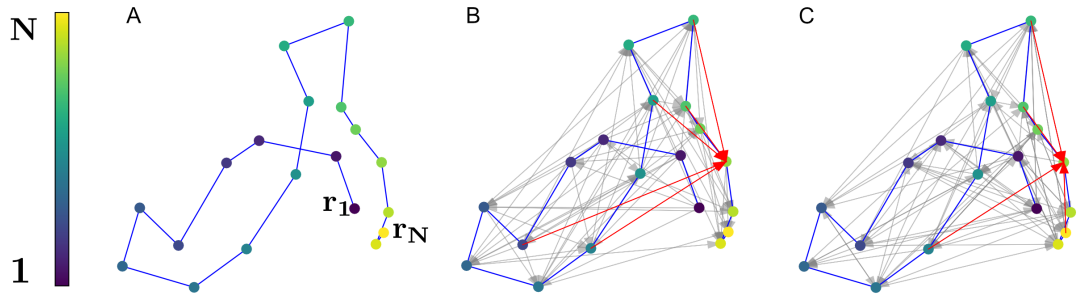

Figure III.1: Examples of graphs associated to a single trajectory. A) An example of a short, subdiffusive fBM trajectory. B) Graph associated to the trajectory following a causal geometric edge wiring scheme (see Section III.1.2). C) Graph obtained by wiring edges at random. Edges linking to a single selected node are shown in red in B and C.

model changes in random walker dynamics, and mixture models [7, 8] have been applied to approximate the point spread function in fluorescence microscopy and to model and analyse complex networks [9].

Over the past 4 years, extensions of deep learning approaches to graph data in the form of graph neural networks have attracted significant attention [2] and have demonstrated great efficiency for representation learning on point clouds, graphs and manifolds [10, 11]. GNNs meet several criteria that make them well suited for analyzing trajectories, and which motivated the design of our learning procedure. (i) They can be applied directly, using a shared architecture, to trajectories of different lengths [12, 11]. (ii) The choice of graph structure allows taking into account different time scales while retaining a sparse architecture. (iii) Numerous known features associated to random walks, such as the convex hull [13, 14, 15], first passage times [16, 17, 18], or the distributions of different features' extreme values [19, 20, 21], are linked to geometric properties which can be learned efficiently using GNNs. (iv) Finally, while advances in machine learning are associated to impressive achievements [22, 23] they are often obtained with large scale models (several millions to billions of parameters) that are prone to over-fitting and are challenging to interpret [24]. Hence, a model with a limited number of parameters and a means to quantify the acquired information during training would be beneficial to understand the requirements for random walk inference using machine learning. All these criteria point towards using graph neural networks for individual random walks analysis.

III.1.2 Graph representation of trajectories

We associate to each trajectory $\mathbf{R} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$ a directed graph $G = (V, E, \mathbf{X})$, with $V = \{1, 2, \ldots, N\}$ the set of nodes corresponding to the positions in the trajectory, $E \subseteq \{(i,j)|(i,j) \in V^2\}$ the set of edges connecting pairs of nodes, and $\mathbf{X} = (\mathbf{x}_1^{(0)}, \mathbf{x}_2^{(0)}, \ldots, \mathbf{x}_N^{(0)})$ a sequence of local feature vectors $\mathbf{x}_i^{(0)}$ associated to each node i in V (fig. III.2A). Each node features vector $\mathbf{x}_i^{(0)}$, of size n_x, may contain any feature of the trajectory or of the graph and may depend only on i or on arbitrary neighborhoods of i. The size of each node's feature vector depends on the dimensionality of the trajectory, so a model trained on trajectories of a given dimensionality can only be applied to trajectories with the same

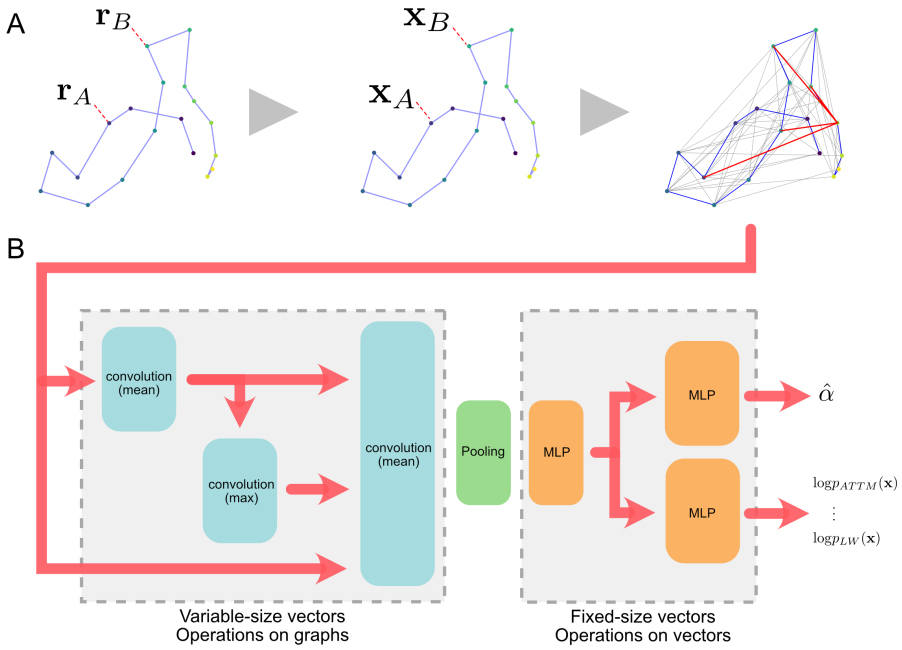

Figure III.2: Graphical representation of the GNN model. A) Construction of a graph from a raw trajectory: (left) raw trajectory, (middle) positions in the trajectory are represented by nodes and a vector of local features is associated to each node; (right) nodes are connected to each other using a wiring scheme as described in section III.1.2 to form a graph on which the GNN is applied. B) Overview of the GNN's architecture. Red arrows symbolise outputs of layers passed as inputs to other layers. When several arrows point to a same layer, their features vectors are concatenated before being passed to the layer. Graph convolution layers are shown in blue, they can process inputs of variable length (for graphs of variable size representing trajectories of variable length) using neural message passing (see Section III.1.4). They use two different aggregation methods for messages: mean (averages over all the messages) and maximum (takes the maximal value of messages, for each feature). In green, the pooling operation embeds the variable-sized vector output from the graph convolution layers into a fixed-size vector representing the trajectory. Orange boxes downstream are multi-layer perceptrons, acting on vectors of fixed size as in conventional neural networks architectures.

number of dimensions. However, the approach can be applied to trajectories with any number of dimensions using the exact same procedure, only the size of the node feature vectors will differ. This graph-based encoding of trajectories was inspired by applications of GNNs to point cloud data [12, 11], but where time is an additional feature associated to each point here. We here consider only trajectories recorded at regularly spaced points in time. However, since time is an explicit feature of each node, the method is straightforward to extend to trajectories with missing points or recorded at irregular sampling rates.

We attach to each node i the time t_i and three differently normalised versions of the i-th position $\mathbf{r}_i$: (1) normalised using the standard deviation of step sizes, (2) normalised using the standard deviation of positions, and (3) normalised using the mean step size. We also include in each node features the values of the cumulative sums of step sizes and of squared step sizes up to their time-point, computed using each of the three normalised positions.

In order to prevent Levy walks from having a disproportionate influence in the learning process, due to the extreme distance values that the walk can induce, we clipped extreme jump lengths before normalisation. We noted during initial training that these rare events induced significant bias in the batch normalisation layers [25].

Thus, two matrices initially represent the graph associated to the random walk: the (sparse) adjacency matrix A, of size (N, N), and the node feature matrix X, of size (N, n_x) where n_x is the number of features initially attached t. Note that we may also add features to edges in the graph [26], represented by an edge feature matrix U, of size $(|E|, n_e)$.

Remark: At the time of writing, the implementation of graph convolution involving edge features was much slower than those relying strictly on nodes features, which is one of the reason why we did not use them. Moreover, they require a higher number of model parameters. We nonetheless chose to use edge features in later developments of the method, as explained in section III.6.

A known limitation of message passing GNNs has motivated us to choose particular wiring schemes for the graph. The mechanism of information propagation in a GNN involves iteratively passing messages between neighboring nodes, aggregating them in each step. The latter creates an information bottleneck [27], leading to a limitation of information encoding in finite sized vectors. A GNN may fail to faithfully propagate local information stemming from nodes separated by long paths in the graph. It can hence perform poorly if the properties to be predicted depend on long-range information, which is generally the case for the task of classifying various random walks and inferring their anomalous exponents. Our approach overcomes these limitations by using structured wiring schemes to ensure more direct message passing from distant nodes. We discuss here two different wiring schemes, (i) hierarchical causal and (ii) regular random, but many options are possible. In the hierarchical causal scheme (i), the incoming edges of each node connect only to nodes in the past (respecting causality): node i is connected to nodes $i - \Delta_1, \ldots, i - \Delta_{max}$, where $(\Delta_i)_{i\geq 1}$ is a geometric series (see details in III.5.1). In the regular random scheme (ii), edges are drawn at random, with the only constraint that all nodes have the same *in*-degree, generating a type of random regular graph. Example graphs can be seen in Figure III.1. In both schemes, the graph structure ensures that distant time points of the random walk are connected by short paths.

III.1.3 Neural network architecture

We used a two-part architecture for the graph neural network, starting with an encoder followed by task-specific multi-layer perceptrons [24], each estimating a property of interest from the latent representation built by the encoder – here the anomalous exponent and the random walk class. This architecture, shown in figure III.2, enables multi-task training (i.e. simultaneous inference of a random walk's class and anomalous exponent).

The encoder is the entry-point to the model. It embeds the graph representation of the trajectory into a latent space whose dimension is independent of the trajectory length. To do so, it performs several graph convolution operations [28, 29, 30, 31] (described in Section III.1.4 below) which propagate learnt features through the graph. It is terminated by a pooling layer, i.e. an operator that combines an aggregation of features across nodes with a multi-layer perceptron that outputs a fixed-size vector. We used convolution layers using both “mean" and “max" operations to aggregate messages they receive from their neighbors, thus enabling the network to compute a broader variety of features. We also chose to wire

convolution layers so that the last one receives both the output of its predecessor and the initial features to prevent the information from vanishing through bottlenecks created by successive graph convolutions.

III.1.4 Graph convolution layers and neural message passing

The core of GNN operations is formulated in terms of neural message passing, which gathers and transmits information from nodes to nodes through connecting edges (Fig. III.3) and aggregates it using basic operations such as convolution and pooling [32, 29, 30, 31, 33].

Graph convolution layers implement operations on node feature vectors following a message passing scheme [33] (Fig. III.3):

$$\vec{x}_i^{(k)} \leftarrow \gamma_k \left(\vec{x}_i^{(k-1)}, \mathcal{X}_{j \in \mathcal{N}(i)} \vec{x}_i^{(k-1)} \right) \ .$$

Here, the exponent $(k-1)$ denotes a vector's value before the k-th convolution, and (k) its updated value after the convolution. The functions γ_k are neural networks (in our case, multi-layer perceptrons) whose weights are learned during training. The size of $\vec{x}_i^{(k)}$ is not constrained to be the same as that of $\vec{x}_j^{(k-1)}$: drawing analogy with convolution layers in classic architectures used for image processing, each output size of γ corresponds to a convolution kernel. χ is a permutation-invariant aggregation operator which reduces the set of input vectors to a single vector of fixed size. In our case, it is either a feature-wise mean or maximum across nodes. We illustrate both the graph building process and the learning in Figure III.2.

We point out here that, while initial motivations for using GNNs stemmed from the known efficiency (see Section III.1.1) of graph models in physics, GNNs differ strongly from physically motivated graph models and message passing techniques. The neural messages do not represent beliefs about features or variables of interest, they are not normalised by conservation of probability and optimisation is not performed by sampling. The graph serves as a means to link features to sets of neighbor features and allow "classical" learning to be performed by optimisation.

III.2 Results

We test our model's performance on classification (model selection) and regression (parameter estimation) tasks for simulated trajectories of varying lengths and with a range of localisation noise amplitudes and characteristic scales of motion. We use the same convention as in the AnDi Challenge [34] regarding the localisation noise, i.e. we apply an independent Gaussian noise to all positions in a trajectory with a standard deviation equal to a constant factor of the expected standard deviation of the jump sizes of the random walk. We refer to this proportionality factor as the “noise amplitude", and we consider noise amplitudes in the range $[0, 1]$.

The lengths and anomalous exponents of trajectories used for both training and evaluation were sampled uniformly between their respective extreme values (unless otherwise specified, $N = 10$–1000 and $\alpha = 0.05$–1.95). For each evaluation, performance metrics were computed using one million trajectories (200 000 from each model), generated using the AnDi package [34]. We added localisation errors with the same amplitude to all dimensions of a

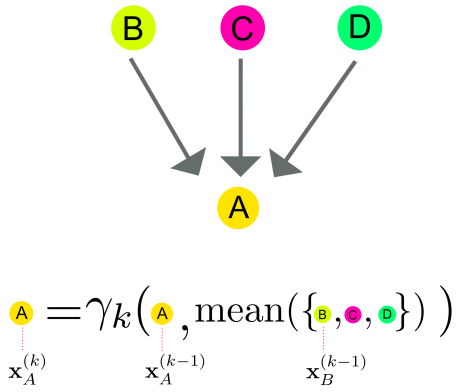

Figure III.3: Illustration of a graph convolution. At each iteration k, each node (A) aggregates messages sent by nodes that are connected to it. The color code illustrates how the information is propagated between nodes. The weight parameters of the multi-layer perceptron γ_k, corresponding to the kth layer of the GNN, are learnt during the training. Here, the aggregation scheme for feature vectors shown is a mean over neighbor nodes.

trajectory. Here, we will show only results for 3D trajectories, but the dimension of the trajectories does not qualitatively change results.

Unless otherwise specified, we trained the GNN to perform both classification and regression simultaneously by using a training objective given by a simple sum of the mean squared error (MSE) of the estimated anomalous exponent and the cross-entropy between the true and predicted class labels. We use the mean absolute error (MAE) between the estimated and true values of the anomalous exponent α to quantify regression accuracy, the F_1 score to quantify overall classification accuracy and provide confusion matrices to estimate class-by-class evaluation. Section III.5.1 gives detailed definitions of each of the measures. The code of the model described in the paper is available at https://github.com/DecBayComp/gratin. This model is dubbed GRATIN (Graphs on trajectories for inference).

III.2.1 Performance in absence of localisation noise

We show in Figures III.4 and III.5 the performance of a GNN trained on trajectories of lengths between 10 and 1000 and in the absence of localisation noise. Figure III.4A shows that the accuracy of the inference of the anomalous exponent α from a single trajectory depends on trajectory length and the class of random walk considered: the anomalous exponent of ATTM is harder to infer than that of CTRWs or fBMs. Looking at how the estimation error on α depends on its true value, we see as can be expected a conservative bias shifting estimates away from extreme α values (i.e. 0, 1 and 2). The bias is pronounced for short trajectories and decreases with trajectory length (Fig. III.4B,C). The bias stems in majority from the poor performance on ATTM trajectories.

Looking at the confusion matrices to assess classification accuracy (Fig. III.5), we see that even for short trajectories, most walks are accurately classified, and misclassifications mainly confuse sBM and fBM. For trajectories longer than 200 points, the classification exhibits

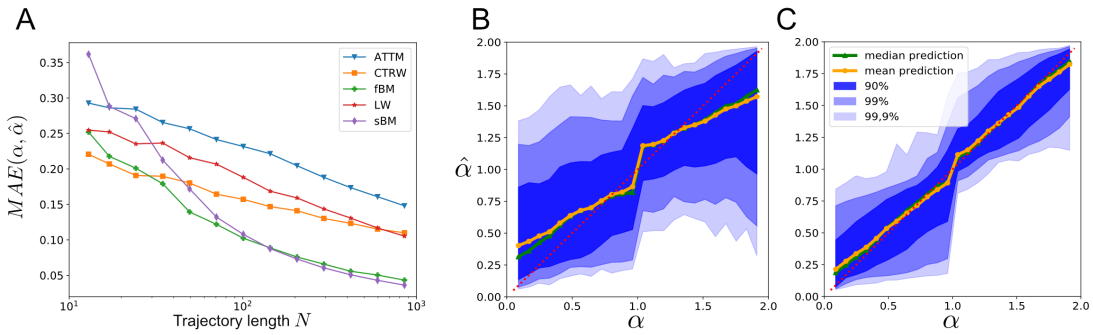

Figure III.4: Performance of the inference of the anomalous exponent α on 3D trajectories. A) Mean absolute error (MAE) of estimate of the anomalous exponent, $\hat{\alpha}$, as a function of trajectory length N. B) & C) Distribution of $\hat{\alpha}$ values as a function of true value of α for B) trajectories of 10 to 100 points and C) 100 to 1000 points. x-axis: true exponent α, y-axis: inferred exponent $\hat{\alpha}$.

high performance. The GNNs classification performance for short trajectories illustrates that it relies not only on the asymptotic properties but also on finite-scale features of the random walks. Hence, even for short trajectories identification is possible. Furthermore, the confusion between sBM and fBM can be explained by the similarity of the processes for short trajectories, with a more pronounced effect for α close to one (in the range of approximately 0.7 to 1.4). sBM and fBM indeed share the same marginal probability density for the time-evolution of the walker's position [35] and they both approach Brownian motion as α approaches one.

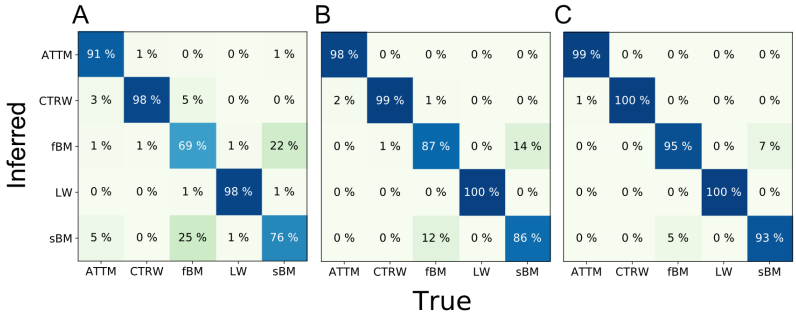

Figure III.5: Confusion matrix for model classification, i.e. probability to identify a trajectory as having been generated by each random walk model (inferred class) given its true generating model (true class). For trajectory lengths in the range: A) 10-50, B) 50-200, and C) 200-1000.

III.2.2 Robustness to noise

Experimentally recorded trajectories are subject to various sources of noise. We here focus only on localisation noise, modelled as an uncorrelated Gaussian noise, but correlated noise

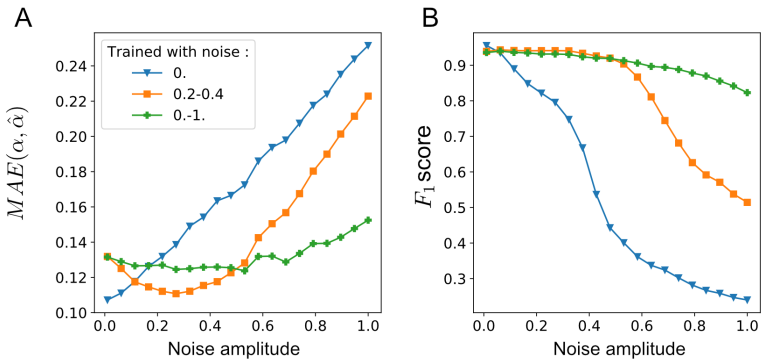

Figure III.6: Inference performance as a function of noise amplitude and for different ranges of noise amplitudes included in the training data. A) Mean absolute error of the anomalous exponent inference, MAE $(\hat{\alpha})$. B) F_1 score for model identification. The F_1 score is the harmonic mean of precision and recall (see details in Supplementary Section III.5.1).

sources may also be present [36, 37]. We investigated the performance of the inference when both training and inferring on trajectories observed with a broad range of noise amplitudes.

As shown in Figure III.4, high localisation noise may significantly impair the accuracy of inferences when it has not been taken into account in the training data. We tested a global approach to induce noise robustness by assuming that no information on noise was accessible safe for a range of possible amplitudes. Hence, we trained on trajectories with added localisation error whose noise amplitude was randomly drawn from $[0, 1]$. Remarkably, training on such a wide distribution of noise leads to a nearly flat performance in of the anomalous exponent inference over the full range of noise amplitudes (green curve in Fig. III.6). Similarly, the performance in classifying the trajectories, i.e. identifying their generative model, exhibited only a limited decrease with increasing noise amplitude. Conversely, when inference is performed on trajectories with localisation noise outside the range that the GNN was trained on, the performance of both regression and classification may degrade significantly.

In most experimental settings, there are means by which a range of possible values for positioning noise can be either deduced *a priori* or directly measured. Taking this in account by simulating with the same noise range to generate the training set increases the accuracy (curve corresponding to 0.2-0.4 in Fig. III.6A), even if the GNN trained on the whole range of noise amplitudes already performs well. We refer the reader to [38] for an example of a procedure to estimate positioning noise within the context of single molecule experiments.

III.2.3 Improving performance on specific cases

We tested the performance of the architecture on data with parameters corresponding to typical experimental conditions, i.e. short trajectories (between 10 and 50 points) corrupted by a significant but bounded amount of noise (here, noise amplitudes between 0.2 and 0.4, as defined at the beginning of this section). The model was trained on this same range of trajectory lengths and noise amplitudes. We illustrate the model's performance in Figure III.7. We show that despite the inherent difficulty of inferring properties from such short observations, the model is precise enough to extract relevant properties from these trajectories, e.g., it can reliably separate subdiffusive and superdiffusive trajectories and distinguish CTRWs

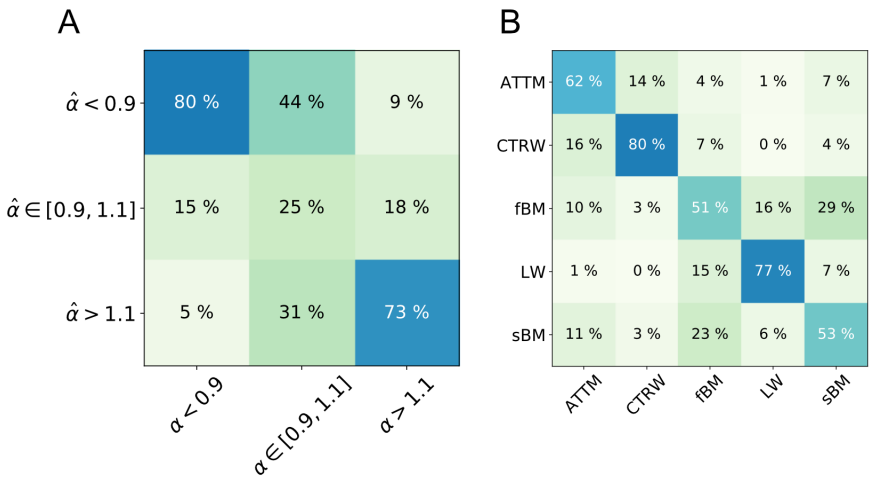

Figure III.7: Performance on short trajectories ($10 \geq N \geq 50$) for a GNN trained with a known range of localisation noise amplitudes. A) Confusion matrix for identification of subdiffusive, normally diffusive and superdiffusive anomalous exponent values. B) Confusion matrix for model classification. The color code is identical to the one in figure III.5.

from fBM. These results suggest that if we consider the random walk models to provide good approximations of the dynamics we may encounter in an experimental system, the approach may be efficiently leveraged to analyse single molecule experiments.

We next assessed more generally the effect of specialisation of the inference task on performance. First, we compared the performance of a model trained specifically on short trajectories ($10 \leq N \leq 100$) to that of a model trained on short and long trajectories ($10 \leq N \leq 1,000$) (Fig. III.8A,B). We only considered model performance on the range of lengths that was common to both training sets. Then, we compared the performance of a model trained solely to infer the anomalous exponent with one trained both for regression and classification (Fig. III.8C,D). We see that the performance increases with specialized training, although here only to a limited extent. This capacity of GNN models trained on numerical simulation to provide generalisable inference will be instrumental for their use in experimental data analysis.

III.2.4 Influence of the number of neural network parameters

GNNs generally do not require as many parameters as other recent deep neural architectures [10]. Here, we investigated the effect of the GNN's size (as measured in total number of parameters) on its performance. To do so, we modified the base architecture, first by making layers thinner and then by removing some layers (reducing by up to two orders of magnitude the number of parameters). Details are available in Table b).

Results are shown in Figure III.9. It is noteworthy that even with only about 1,600 parameters, the GNN maintains a good performance. This suggests that increased model tractability may be possible for GNNs.

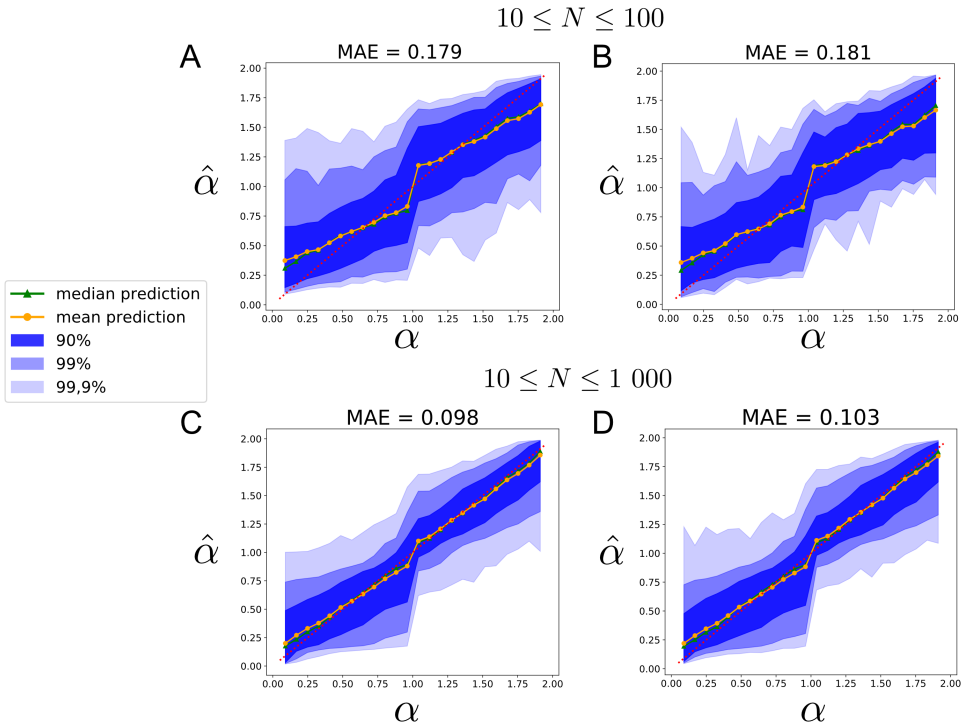

Figure III.8: Improvement in model performance with specialized training: Distribution of estimated anomalous exponent values, $\hat{\alpha}$ as a function of the true value α for GNNs trained on only short or both short and long trajectories when applied to analyse short trajectories (A,B) and for GNNs trained only for regression or for both regression and classification (C,D). A) GNN trained only on short trajectories, $10 \leq N \leq 100$, and B) GNN trained on both short and long trajectories, $10 \leq N \leq 1000$, both applied to short trajectories of lengths $10 \leq N \leq 100$. C) GNN trained only on the anomalous exponent estimation task and D) GNN trained on both tasks.

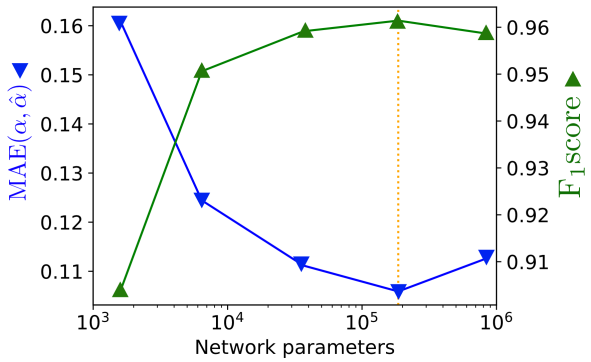

Figure III.9: Inference performance (MAE and F_1 score) as a function of the number of neural network parameters. The vertical yellow line indicates the number of parameters used in the rest of the paper.

III.3 Discussion

III.3.1 Latent space encoding of physical properties and generalisation

Our work lies in the framework of simulation-based inference [39]. This allowed large scale data generation at low computational cost and ensured that features of one dataset could not impair the learning. This contributed to reducing possible overfitting bias [40]. However, since some processes are non-ergodic, it is still a concern that the learning procedure might have failed to learn relevant properties on a finite dataset. Moreover, the statistics of experimentally recorded trajectories are unlikely to exactly match those of any of the models that we trained our machine learning model on. Here, we show that the learned latent space of the GNN encodes physically relevant properties of random walks, and we explore how it can be used to evaluate the robustness and generalisation performance of the approach.

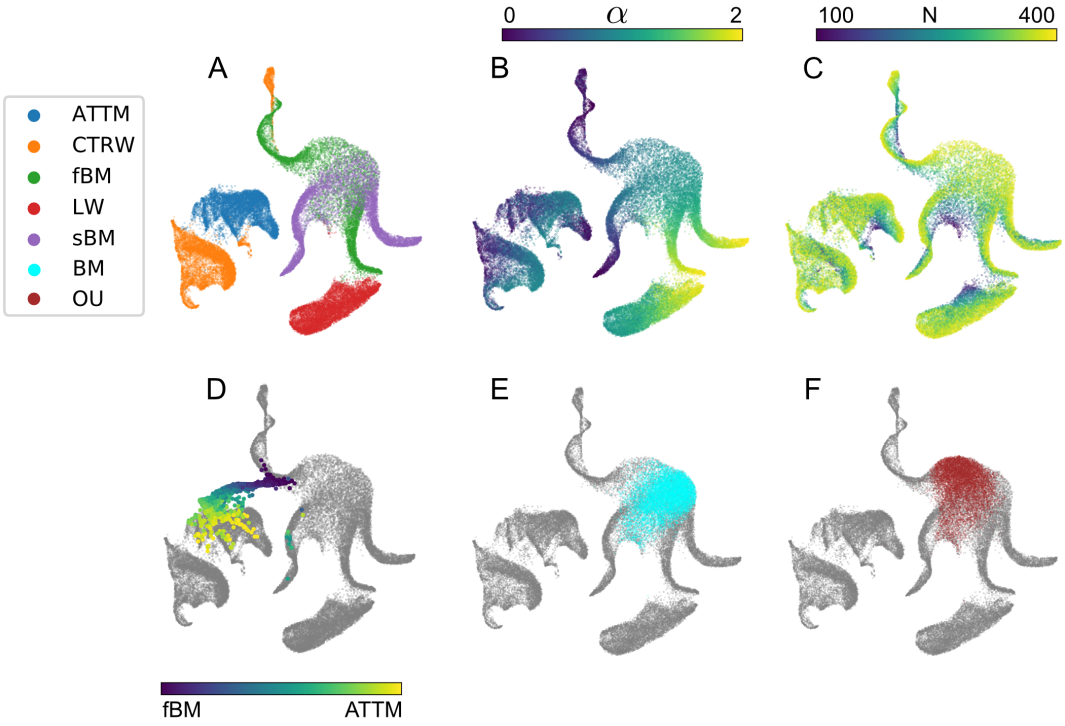

Figure III.10: UMAP [41] representation of the latent space of the GNN. UMAP is a generalist manifold learning and dimension reduction algorithm relying on Riemannian geometry and algebraic topology to perform the embedding. Each point represents the encoded latent position of a single trajectory. A) Positions of each trajectory colored by the random walk's true model class. B) Mapping of the anomalous exponent α in the latent space. The color is associated with the value of α. C) Mapping of the trajectory length in the latent space. The color is associated with trajectory length. D) Positions of trajectories mixing fBM and ATTM in different proportions. Colors indicate the percentage of the trajectories generated by each of the two random walks. Note that trajectories continuously occupy the empty space between the ATTM cluster and the fBM domain. E) Projection of Brownian trajectories in the latent space. F) Projection of Ornstein-Uhlenbeck trajectories in the latent space.

In Figure III.10 we provide a representation of the latent space (output of the penultimate MLP module). Each point corresponds to the latent representation of a single trajectory. We relied on UMAP [41], a dimensionality reduction algorithm in the family of manifold learning techniques, for 2D visualisation of the high dimensional latent space while preserving its local topology.

First, the structure reveals how well the model is able to distinguish trajectories of different types and anomalous exponents. Levy Walks, CTRWs and ATTM form three well separated clusters while fBMs and sBMs form a more complex shape with an extended overlapping region corresponding to $\alpha \approx 1$, i.e. the regime of normal (Brownian) diffusion (Fig. III.10A). Within each cluster and continuous region, the anomalous exponent is mapped in a stable fashion (Fig. III.10B). In concordance with the lower performance on regression for the ATTM model, we see that the value of α is less clearly mapped than for the other random walk models. The length of the trajectories, which directly relates to the available quantity of information, is also encoded in an ordered manner within the latent space, along a direction that is roughly orthogonal to the direction encoding the exponent.

In order to investigate the capacity of the GNN to encode physical properties of the random walks and its ability to generalize to trajectories with unseen properties, we applied the GNN to trajectories of walkers starting with a given type of motion and ending with another, and to random walks generated by unseen models on which the GNN was not trained.

To evaluate how the GNN encodes random walks that change of motion class over time, we generated trajectories of fixed length ($L = 200$) where the first part was generated by a given model of subdiffusive walks and the second by another, and varied the relative importance of the two parts. Both segments have $\alpha = 0.5$. In Figure III.10D we can see the trajectories' encoded positions in the latent space draw a transition from the fBM domain to the ATTM cluster as the percentage of the ATTM part in the trajectory increases. Some trajectories fall within a region of the latent space that was originally not occupied. The model is thus able to continuously encode random walk properties and interpolate between the properties of the two models based on previously unseen behaviour. It is an indication of its ability to generalize.

To investigate how the GNN behaves when used on trajectories generated by models not included in the training phase, we use the GNN to encode the properties of trajectories generated by pure Brownian motion (BM, with $\alpha = 1$), and by the Ornstein-Uhlenbeck (OU) process which models Brownian motion confined in a harmonic potential [42]. In Figure III.10E we show that BM trajectories all fall within the portion of the latent space where fBM and sBM overlap, corresponding to the region where $\alpha = 1$ and their dynamics approach Brownian motion. The OU trajectories cover a region where subdiffusive fBMs with α values close to, or slightly below one are encoded. This is also a physically sensible encoding as the OU process shows anticorrelated dynamics, similar to the fBM but with a much faster, exponentially decaying kernel. In this respect, the latent space does encode relevant physical properties, suggesting that the GNN will generalize well to experimental data with statistical properties that may not be exactly equal to any of the random walk models it was trained on.

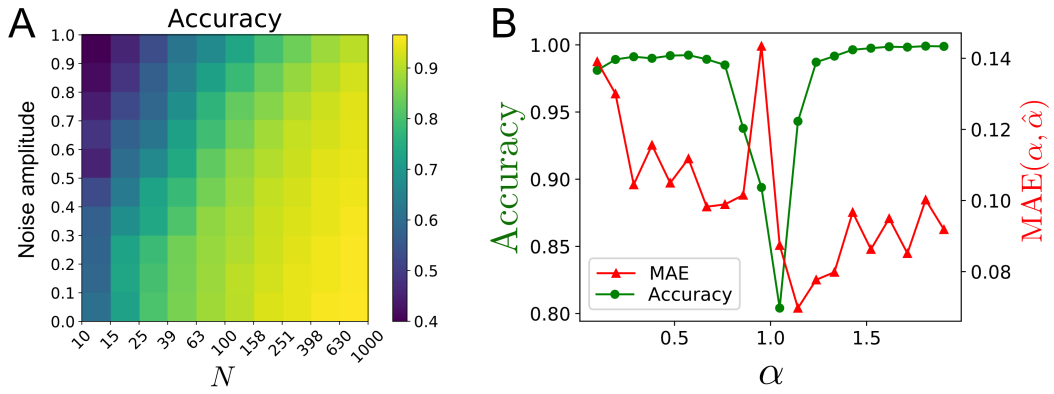

Figure III.11: A) Accuracy of random walk classification as a function of trajectory length and noise amplitude. B) Classification accuracy and MAE of the estimated anomalous exponent $\hat{\alpha}$ as a function of the true value of the anomalous exponent. Trajectories are of length $10 \leq N \leq 1\,000$, and without positioning noise.

III.3.2 Misclassified random walks

We investigated the sources of misclassification by the GNN. In Figure III.11A we show the classification accuracy as a function of both trajectory length and noise factor. Error in model identification is concentrated in high noise and low length regions. Figure III.11 helps to pinpoint the hardest samples to characterise. Intuitively, they are short and noisy. Furthermore, as could be expected from the latent space's structure, properties of trajectories whose anomalous exponent is close to one tend to be harder to infer (Fig. III.11B).

III.3.3 Influence of graph structure

Here, we investigated GNNs applied to graphs generated from recorded trajectories using a specific wiring scheme. As discussed in Section III.1.2, we designed this scheme to ensure that the graph would connect distant portions of the trajectories, for efficient information exchange, while retaining a causal and hierarchical dependency structure. To test the influence of graph structure, we compare its performance to that of a random wiring scheme in Supplementary Section III.5.3.

Recent work [28] has highlighted a strong link between the how a GNN's depth affects its performance and the connectivity of the graphs it is applied to. Following this direction, it has been proposed to learn the graph structure itself from an input point cloud in Euclidean space [43]. This could very well be applied to this setting in the future.

III.3.4 Computational Complexity

The inference-time algorithmic complexity of the GNN model depends on the three main time-consuming parts when applying it to infer the properties of a random walk:

1. initial features evaluation;

2. forward pass through graph convolutions;

3. forward pass through subsequent layers.

1. The set of features we use to initialize each node's feature vectors can be computed in $O(N)$ time (N is the number of nodes, equal to the trajectory length).

2. From Eq. (III.1.4) it follows that the complexity of a convolution operation for all nodes of the graph is of $O(|E|)$ time complexity in the general case. In our case we restrict node's degrees to a maximum of k, so we have $E \leq k\,N$ and the convolution operation thus has $O(N)$ time complexity too.

3. Finally, as the dimension of the latent representation is independent of the graph's size, the forward pass through subsequent layers has $O(1)$ time complexity.

Thus, provided that the number of edges scales linearly with the number of nodes, this architecture can scale well to long trajectories, inferring their model class and anomalous exponent in $O(N)$ time.

III.4 Conclusion

In this paper we have shown that we could learn a physical representation of anomalous random walks using GNNs. We leveraged this representation to infer both the anomalous exponent and the model of a random walk from single trajectories. We relied on simulations to train our procedure. The scheme was found to be efficient in performing regression and classification tasks as well as being robust to positional noise. We showed that the latent space learned by the GNN is linked to the physical properties of the random walks.

While GNNs provide a general and expressive framework, they are still a new approach. Future developments are likely to improve their computational efficiency (e.g., improving their scalability to large graphs [44] and the efficiency of GPU acceleration [44]), their statistical power (e.g., by incorporating higher-order geometric features [33, 45] or appropriately relaxing the permutation-invariance of the aggregation operator), as well as our theoretical understanding of their capacities [26].

Representation learning paves the way to new approaches to explore biomolecule random walks whose dynamics cannot be purely described by a unique canonical random walk model. We foresee two directions for developments: First, neural networks and feature learning may be used to accelerate likelihood-free inference [39], allowing to fit more complex and realistic simulation-based models for experimentally recorded random walks. Second, the ability to learn relevant representations from random walk realisations may be exploited to develop unsupervised approaches to analyse experimentally recorded random walks.

Future work may also involve imposing constrains during learning to reinforce known symmetries in the random walk (e.g. directional symmetries) to increase training efficiency and to ensure that the neural network does not learn spurious features.

III.5 Supplementary Material

III.5.1 Numerical simulations, Graph neural network training and hyper parameters

a) Simulating trajectories

Random walks were generated using the python package provided during the AnDi challenge [34]. We slightly modified it to generate the same noise amplitude along all dimensions, which corresponds more closely to experimental conditions.

We additionally considered pure Brownian motion and the Ornstein-Uhlenbeck process:

- We simulated Brownian motion by directly sampling its displacements according to $\mathbf{X}_{n+1} = \mathbf{X}_n + \boldsymbol{\Delta x_n}$ with $\boldsymbol{\Delta x_n} \sim \mathcal{N}(\mathbf{0}, \mathbb{I})$.
- We simulated the Ornstein–Uhlenbeck process using the Euler method according to the following update formula: $\mathbf{X}_{n+1} = \mathbf{X}_n(1 - \delta t) + \sqrt{\delta t}\boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, 0.1\mathbb{I})$ and $\delta t = 0.01$.

b) Neural network hyper-parameters

The detailed architectures of the model's components are summarized in table b), ordered from the smallest to the largest tested network in terms of the number of parameters. Within this paper we have mostly discussed results associated to the architecture showing the best performance (see vertical line in III.9), which correspond to the fourth row in the table. The first convolution layer receives the initial nodes features (there are 28) as well as, optionally, some of their powers. In the first two architectures presented in table b), it receives only the first power. In the third and fourth, it additionally receives the squared features. In the last architecture it also receives the cubed features (hence the initial width being a multiple of 28). This is meant to allow the network to compute moments of the features distributions. When building the graphs, we used a maximal $in-$degree value of 20.

parameters	γ layers	projector	α module	classifier
1 588	(56,8) (16,8) (32,8)	(24,6)	(6,16,1)	(6,5)
6 420	(56,16,16) (32,16,16) (64,16,16)	(48,8)	(8,64,16,1)	(8,16,5)
36 660	(84,32,32,32) (64,32,32,32) (128,32,32,32)	(96,64,16)	(16,128,64,16,1)	(16,16,16,5)
185 879	(84,128,64,64) (128,128,64,64) (256,128,64,64)	(192,128,64,32)	(32,128,128, 64,16,1)	(32,64,32,5)
871 596	(112,256,128,128,128) (256,256,128,128,128) (512,256,128,128,128)	(384,512, 256,128,64)	(64,128,128, 128,64,1)	(64,128,64, 32,5)

We used batches of 128 trajectories for training, with a learning rate of 10^{-3}, exponentially decaying until it reaches $2 \cdot 10^{-4}$ after the network has seen $3 \cdot 10^6$ trajectories. Training lasts about 5 to 10 hours. We relied on the "PyTorch Geometric" package [2] to implement the graph convolutions and perform the learning.

c) Metrics

The mean square error (MSE) is computed as follows: $\mathrm{MSE}(\hat{\alpha}) = \langle(\hat{\alpha} - \alpha)^2\rangle$.

The mean absolute error (MAE) reads: $\mathrm{MAE}(\hat{\alpha}) = \langle|\hat{\alpha} - \alpha|\rangle$

The cross-entropy loss, used for the classification task, reads: $\mathrm{CE} = -\langle\sum_{i=1}^{5} \delta_{m,i} \log \hat{p}_i\rangle$. Where $m \in \{1, 2, 3, 4, 5\}$ denotes the index of the true model of a trajectory.

To quantify overall classification performance, we used the F_1 score, which is the harmonic mean of the *precision* and the *recall* of the model. Using TP, FP & FN to denote the number of true positives, false positives and false negatives, respectively, we have:

- Precision: $P = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$
- Recall: $R = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$
- F_1 score: $F_1 = 2\frac{\mathrm{PR}}{\mathrm{P+R}}$

The confusion matrices, used to illustrate the ability of the GNN to infer random walk models, are defined as follows: $C_{i,j} = \langle\delta_{\hat{m}(\mathbf{R}),i}\rangle_{\mathbf{R} \text{ of type j}}$, where $\hat{m}$ is the index of the inferred model, i.e. the one which has been assigned the highest probability, and δ is the Kronecker symbol. That is, $C_{i,j}$ is the probability that a trajectory generated by the model class j (column) is classified as belonging to model class i (row). Defined this way, the diagonal elements $C_{i,i}$ are the per-class recall.

III.5.2 More complex Graph Neural Networks

In this paper, we focused on an approach where the GNN learns to build relevant features of random walks. We focused on a setting where features were only assigned to nodes, but the general framework of graph neural networks allows a more complex structure, by attaching features to edges, and by having two multi-layer perceptrons involved in the convolution operations: one before and one after the aggregation. In this more general setting, the neural message passing equations [33] read:

$$\vec{x}_i^{(k)} \leftarrow \gamma_k \left(\vec{x}_i^{(k-1)}, \mathcal{X}_{j \in \mathcal{N}(i)} \phi_k \left(\vec{x}_i^{(k-1)}, \vec{x}_j^{(k-1)}, \vec{e}_{j,i}^{(k-1)}\right)\right)$$

We show in Figure III.12 that edge features allow better performance on noise-free trajectories but is less robust to localisation noise.

III.5.3 Random versus structured connection patterns

As illustrated in Figure III.13, connecting nodes according to a regular geometric pattern yields better performance than linking them at random.

In the geometric causal wiring scheme, node i receives edges from nodes $i - \lfloor\beta_1\rfloor, i - \lfloor\beta_2\rfloor, \ldots, \min(0, i - \lfloor\beta_k\rfloor)$, where k is the maximal *in*-degree, independent of trajectory length

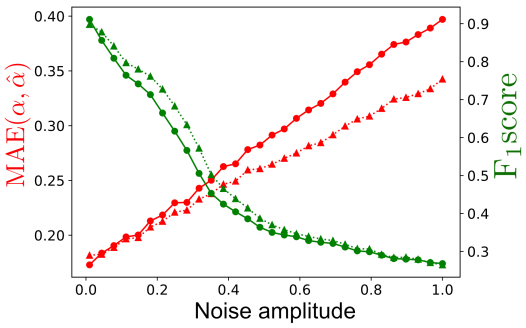

Figure III.12: Performance of GNNs trained with (plain lines, circles) or without (dashed lines, triangles) features attached to edges as a function of the noise amplitude in the trajectories. MAE is shown in red, F_1 score in green.

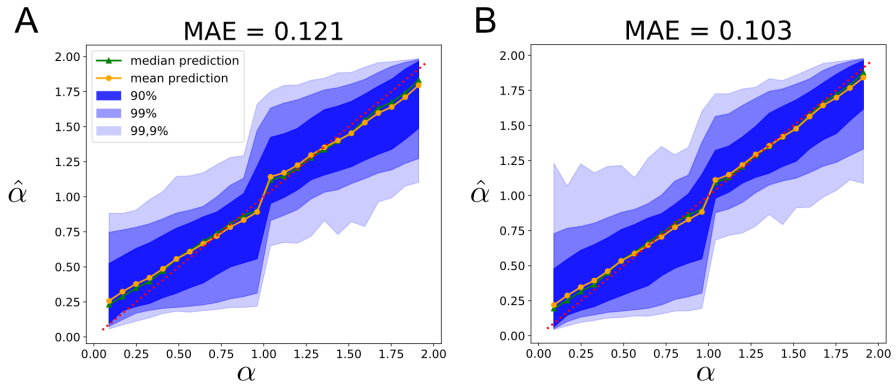

Figure III.13: Regression performance of GNNs applied to different graph structures. A) Random regular graph structure. B) Causal hierarchical graph structure. Shaded regions represent probability intervals of the estimators.

and which enforces sparsity, and where $\beta_1, \beta_2, \ldots, \beta_k$ is a geometric progression, parametrized such that the last node receives a connection from the first node, and $\lfloor \beta \rfloor$ denotes the integer part of β. We ensure that no edge is doubled. We used a value of $k = 20$ throughout this paper. Hence, nodes close to the start of the trajectory receive less connections than those located at the end.

III.6 Recent Improvements

Here, we present a few improvements to the neural network and the procedure used to extract two-dimensional embeddings of latent vectors.

III.6.1 Parametric UMAP for reproducible embeddings

UMAP such as those presented in figure III.10 have the inconvenient of requiring a new optimization phase each time a new data point is added to the set of points to embed.

Besides, it is sensible to the relative importance of the types of walks present in the dataset to visualize, and might thus not render well the most scarcely populated regions. Therefore, in the following chapters, we resort to a variant of the UMAP called "Parametric UMAP". In this variant, instead of optimizing the low-dimensional coordinates of each point, one learns an "encoder" neural network using as a loss the function minimized when fitting the UMAP. Once done the training phase, on a well-balanced dataset, this method has the advantge of being faster and more stable than the classic UMAP.

III.6.2 Rotational invariance

The problem of inferring the nature of a trajectory – provided that, if it is directed, the direction of the drift is not an object of attention – should not be affected by a rotation of the considered trajectory. Enforcing such symmetries into the architecture of neural networks is a mean to guide their convergence towards relevant functions [46, 47]. Here, the rotational invariance can be guaranteed by removing all direct dependence of the input graph to the coordinates of the trajectory, resorting only to distances: instead of using the point's coordinates as node features, the geometry of the trajectory can be encoded using inter-point distances as edge features. Plausible coordinates might be unambiguously retrieved from a distance matrix if all pairwise distances are provided [48]. However, sparse distance matrices such as the one we have when considering only a limited number of neighbors per node greatly constrain the trajectory's geometry, especially given the low dimension of in which points lie ($D \leq 3$). Thus, we leveraged the possibility of graph convolutions to process edge features to remove all non-rotation invariant node features. Of course, we considered only rotation-invariant edge features.

III.6.3 Other improvements of the GNN architecture

In more recent applications of the method, we relied on more modern convolution layers: the GIN (Graph isomorphism network) and GINE convolutions [26, 49]. GIN implements the following operation

$$x_i^{'} = h_\theta \left((1+\epsilon) x_i + \sum_{j \in \mathcal{N}(j)} x_j \right)$$

with $\mathbf{h}_t heta$ a MLP and ϵ a learnt parameter. GINE uses the same inspiration but involves edge features. If edge and node features do not have the same dimension, a projection matrix is learnt. Its operation reads as follows:

$$x_i^{'} = h_\theta \left((1+\epsilon) x_i + \sum_{j \in \mathcal{N}(j)} ReLu(x_j + e_{i,j}) \right)$$

Besides, we have observed that using the InstanceNorm layer introduced in [50], and which computes normalization statistics at the graph level, yielded much more stable convergence than classic batch normalization layers. Classic batch normalization of nodes-features actually often prevented the learning from converging. Eventually, we observed that the attention pooling operation tends to better propagate gradients to the first convolution layers than mere average pooling.

III.6.4 Varying the exposure and handling irregular sampling

Another improvement of the network, although not yet implemented, would be to consider a range of exposure times. This could be done in two ways: either by adding the exposure as a global feature of the graph, after the pooling layer, or by specifying the time coordinate of each node. The second option is more flexible as it would also allow the network to process irregularly sampled trajectories. Balancing the various modes of temporal sampling in the training data is nonetheless delicate, as the relation between the difficulty of inferring a trajectory's parameters dependance to the ratio $\sqrt{D\Delta t}/\sigma$ and the length N is unknown in most cases.

Bibliography

[1] **Verdier Hippolyte**, Duval Maxime, Laurent François, Cassé Alhassan, Vestergaard Christian L, and Masson Jean-Baptiste. Learning physical properties of anomalous random walks using graph neural networks. *Journal of Physics A: Mathematical and Theoretical*, 54(23):234001, 2021.

[2] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

[3] Daphne Koller. *Probabilistic Graphical Models: Principles and Techniques (Adaptive Computation and Machine Learning series)*. The MIT Press, jul 2009.

[4] Jonathan S. Yedidia. Message-passing algorithms for inference and optimization. *Journal of Statistical Physics*, 145(4):860–890, October 2011. doi: 10.1007/s10955-011-0384-7. URL https://doi.org/10.1007/s10955-011-0384-7.

[5] Marc Mezard. *Information, physics, and computation*. Oxford University Press, Oxford England New York, 2009. ISBN 978-0198570837.

[6] Christopher Bishop. *Pattern recognition and machine learning*. Springer, New York, 2006. ISBN 978-0387310732.

[7] Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2007. doi: 10.1561/2200000001. URL https://doi.org/10.1561/2200000001.

[8] Denis K. Samuylov, Prateek Purwar, Gabor Szekely, and Gregory Paul. Modeling point spread function in fluorescence microscopy with a sparse gaussian mixture: Tradeoff between accuracy and efficiency. *IEEE Transactions on Image Processing*, 28(8):3688–3702, August 2019. doi: 10.1109/tip.2019.2898843. URL https://doi.org/10.1109/tip.2019.2898843.

[9] Tiago P. Peixoto. Nonparametric bayesian inference of the microcanonical stochastic block model. *Physical Review E*, 95(1), January 2017. doi: 10.1103/physreve.95.012317. URL https://doi.org/10.1103/physreve.95.012317.

[10] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[11] C. R. Qi, L. Yi, H. Su, and L. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *ArXiv*, abs/1706.02413, 2017.

[12] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, July 2017. doi: 10.1109/cvpr.2017.16. URL https://doi.org/10.1109/cvpr.2017.16.

[13] Mirko Luković, Theo Geisel, and Stephan Eule. Area and perimeter covered by anomalous diffusion processes. *New Journal of Physics*, 15(6):063034, jun 2013. doi: 10.1088/1367-2630/15/6/063034. URL https://doi.org/10.1088%2F1367-2630%2F15%2F6%2F063034.

[14] Yann Lanoiselée and Denis S. Grebenkov. Unraveling intermittent features in single-particle trajectories by a local convex hull method. *Physical Review E*, 96(2), August 2017. doi: 10.1103/physreve.96.022144. URL https://doi.org/10.1103/physreve.96.022144.

[15] Denis S Grebenkov, Yann Lanoiselée, and Satya N Majumdar. Mean perimeter and mean area of the convex hull over planar random walks. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(10): 103203, October 2017. doi: 10.1088/1742-5468/aa8c11. URL https://doi.org/10.1088/1742-5468/aa8c11.

[16] Sidney Redner. *A Guide to First-Passage Processes*. Cambridge University Press, August 2001. doi: 10.1017/cbo9780511606014. URL https://doi.org/10.1017/cbo9780511606014.

[17] T. Guérin, N. Levernier, O. Bénichou, and R. Voituriez. Mean first-passage times of non-markovian random walkers in confinement. *Nature*, 534(7607):356–359, June 2016. doi: 10.1038/nature18272. URL https://doi.org/10.1038/nature18272.

[18] Marie Chupeau, Olivier Bénichou, and Satya N. Majumdar. Survival probability of a brownian motion in a planar wedge of arbitrary angle. *Physical Review E*, 91(3), March 2015. doi: 10.1103/physreve.91.032106. URL https://doi.org/10.1103/physreve.91.032106.

[19] Francesco Mori, Satya N. Majumdar, and Grégory Schehr. Distribution of the time between maximum and minimum of random walks. *Physical Review E*, 101(5), May 2020. doi: 10.1103/physreve.101.052111. URL https://doi.org/10.1103/physreve.101.052111.

[20] Claude Godrèche, Satya N Majumdar, and Grégory Schehr. Record statistics for random walk bridges. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(7):P07026, July 2015. doi: 10.1088/1742-5468/2015/07/p07026. URL https://doi.org/10.1088/1742-5468/2015/07/p07026.

[21] Wei Wang, Andrey G Cherstvy, Aleksei V Chechkin, Samudrajit Thapa, Flavio Seno, Xianbin Liu, and Ralf Metzler. Fractional brownian motion with random diffusivity: emerging residual nonergodicity below the correlation time. *Journal of Physics A: Mathematical and Theoretical*, 53(47):474001, November 2020. doi: 10.1088/1751-8121/aba467. URL https://doi.org/10.1088/1751-8121/aba467.

[22] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015. doi: 10.1038/nature14539. URL https://doi.org/10.1038/nature14539.

[23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[24] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[25] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[26] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[27] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. *arXiv preprint arXiv:2006.05205*, 2020.

[28] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications, 2020.

[29] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[30] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems*, 31, 2018.

[31] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds, 2018.

[32] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[33] William L Hamilton. Graph representation learning. *Synthesis Lectures on Artifical Intelligence and Machine Learning*, 14(3):1–159, 2020.

[34] Gorka Muñoz-Gil, Giovanni Volpe, Miguel Angel García-March, Ralf Metzler, Maciej Lewenstein, and Carlo Manzo. The anomalous diffusion challenge: single trajectory characterisation as a competition. *Emerging Topics in Artificial Intelligence 2020*, Aug 2020. doi: 10.1117/12.2567914. URL http://dx.doi.org/10.1117/12.2567914.

[35] S. C. Lim and S. V. Muniandy. Self-similar gaussian processes for modeling anomalous diffusion. *Physical Review E*, 66(2), August 2002. doi: 10.1103/physreve.66.021114. URL https://doi.org/10.1103/physreve.66.021114.

[36] Andrew J. Berglund. Statistics of camera-based single-particle tracking. 82(1):1–8. ISSN 15393755. doi: 10.1103/PhysRevE.82.011917.

[37] Christian L Vestergaard, Paul C Blainey, and Henrik Flyvbjerg. Single-particle trajectories reveal two-state diffusion-kinetics of hOGG1 proteins on DNA. 46(5):2446–2458. ISSN 1362-4962. doi: 10.1093/nar/gky004.

[38] Martin Lindén, Vladimir Ćurić, Elias Amselem, and Johan Elf. Pointwise error estimates in localization microscopy. *Nature Communications*, 8(1), May 2017. doi: 10.1038/ncomms15115. URL https://doi.org/10.1038/ncomms15115.

[39] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

[40] Yaser Mostafa. *Learning from data : a short course*. AMLBook.com, United States, 2012. ISBN 978-1-60049-006-4.

[41] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[42] Gardiner Crispin. *Handbook of Stochastic Methods: for Physics, Chemistry and natural sciences*. Springer, 4th edition.

[43] Anees Kazi, Luca Cosmo, Seyed-Ahmad Ahmadi, Nassir Navab, and Michael Bronstein. Differentiable graph module (dgm) for graph convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[44] Chuangyi Gui, Long Zheng, Bingsheng He, Cheng Liu, Xinyu Chen, Xiaofei Liao, and Hai Jin. A survey on graph processing accelerators: Challenges and opportunities, 2019.

[45] P. Gainza, F. Sverrisson, F. Monti, E. Rodolà, D. Boscaini, M. M. Bronstein, and B. E. Correia. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. *Nature Methods*, 17(2):184–192, December 2019. doi: 10.1038/s41592-019-0666-6. URL https://doi.org/10.1038/s41592-019-0666-6.

[46] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[47] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[48] Gale Young and Aiston S Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3(1):19–22, 1938.

[49] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks, 2019.

[50] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

IV – Variational inference of fractional Brownian motion

This chapter's content has been adapted from the following pre-publication.

Verdier Hippolyte, François Laurent, Alhassan Cassé, Christian Vestergaard, and Jean-Baptiste Masson. Amortised inference of fractional brownian motion with linear computational complexity. 2022

Contents

IV.1 Introduction

Fractional Brownian motion (fBm) [2, 3] is a paradigmatic model of anomalous transport. It is a non-Markovian Gaussian process characterized by stationary increments and long temporal correlations in the noise driving the process. It allows capturing long-range temporal correlations in the dynamics of a walker or its environment, and it is a model of choice to describe a multitude of dynamic processes in numerous scientific fields [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22]. Following the classification given in [23] of the three main sources of anomalous diffusion, the anomalous dynamics of fBm stems from the statistical dependency of the displacements at all time scales. Since fBm is a Gaussian process, it admits an analytical expression of the joint likelihood of the recorded signal. It is

thus an ideal model to investigate the performance of approximate schemes to infer anomalous diffusion, such as variational inference or machine-learning-based approaches, since it allows direct comparison to statistically optimal exact inference.

The position of a random walker undergoing fBm is described by a Langevin equation [2] of the form

$$\frac{d\mathbf{r}(t)}{dt} = \sqrt{K_\alpha}\,\eta(t),$$

where η is a zero-mean Gaussian noise process with covariance $\langle \eta(t_1)\,\eta(t_2)\rangle = \alpha(\alpha-1)\,|t_1 - t_2|^{\alpha-2}$ and K_α is a generalized diffusion constant that sets the scale of the process. fBm is self-similar and ergodic [24, 25]. However, it has been shown to exhibit transient non-ergodic behaviour when confined [25, 26] and it is worth noting that the ergodic regime is witnessed only after a long transient passage exhibiting non-ergodic properties [27, 25, 28, 29]. The noise η is negatively correlated in the subdiffusion regime ($\alpha < 1$), while it is positively correlated in the super-diffusion regime ($\alpha > 1$).

Methods for estimating a random walk's parameters can roughly be divided into two types: heuristic approaches using features extracted from the trajectories [30, 31, 18, 32, 33], and likelihood-based (e.g., Bayesian) approaches [34, 35, 36, 37]. Each has its strengths and weaknesses. Likelihood-based approaches are provably asymptotically optimal, but they are often computationally intensive and are only applicable to random walk models that have a tractable likelihood. Feature-based approaches are typically computationally cheaper, and they can be applied to a much larger range of models since they do not require a tractable likelihood. However, they are generally not statistically efficient, are prone to bias when used on experimental data and their precision can be difficult to evaluate.

We developped an amortized Bayesian inference approach to estimate the parameters of a fBm from a single recorded trajectory. More precisely, this paper focuses on two tasks: (i) amortizing the inference of the anomalous exponent to reduce the computational cost of inference and test how much information about temporal correlation can be inferred by a computational scheme of linear complexity, and (ii) exploring the possibility of retrieving information about finite decorrelation times of the walker's dynamics. We used a graph neural network (GNN) to encode a set of summary features of the trajectory. Trained on simulated trajectories , the GNN allows to capture long-range interactions while retaining a linear scaling of the computational complexity with the length of the trajectories. We trained an invertible network to generate the posterior distribution from the summary features using a variational objective. Focusing on the fBm model allowed us to compare the performance of the amortized approach to maximum likelihood estimation and to the Cramér-Rao bound which provides a lower bound on the variance of any unbiased estimator. We show that our amortized inference attains near-optimal performance as compared to exact likelihood-based inference and to the Cramér-Rao bound. We furthermore discuss the latent space structure learned by the summary network and its ability to encode physical properties. We tested the applicability of the approach to trajectories corrupted by positional noise and its potential to generalize to trajectories that are longer than those seen during training. Finally, we extended the inference procedure to capture a finite decorrelation time in the dynamics which may typically arise in physical environments.

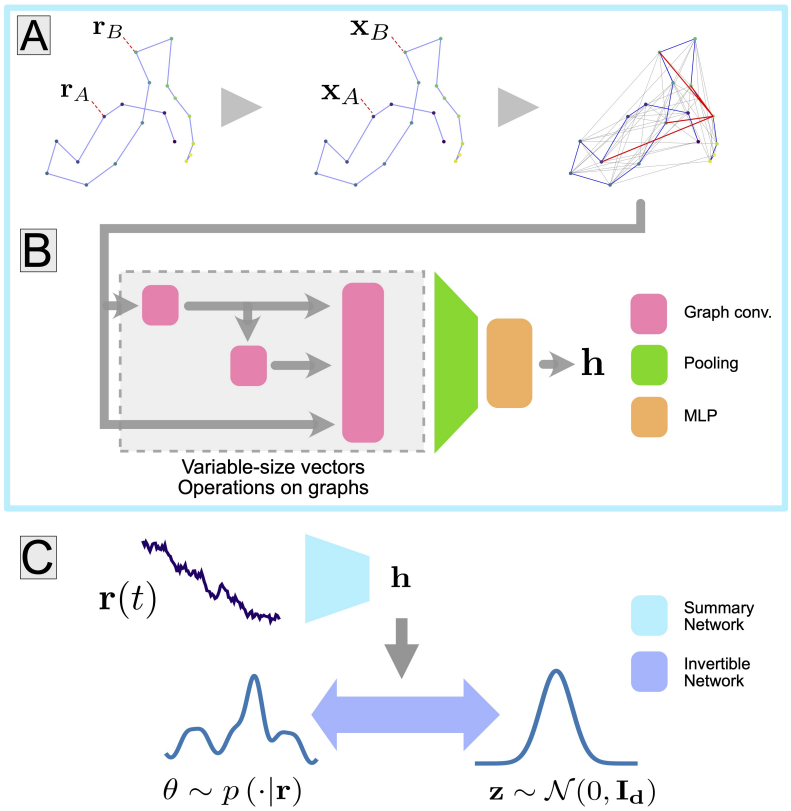

Figure IV.1: **Model Architecture.** A: Construction of a graph from a single trajectory, on which graph learning is performed by the summary network shown in B, see [38] for details. B: Summary network consisting of graph convolution layers, a pooling layer and a multi-layer perceptron. The vector of statistics is indicated by $\mathbf{h}$. C: General structure of the model, with the summary network parametrizing the invertible network. In training mode, the invertible network is used from left to right, and in inference mode it is used from right to left.

IV.2 Amortised Bayesian inference

In the context of parameter estimation, Bayesian inference uses Bayes' theorem to compute the posterior probability distribution of the parameters $\boldsymbol{\theta}$ given recorded data $\mathbf{R}$ (here a trajectory) and a probabilistic model of these data,

$$p(\boldsymbol{\theta}|\mathbf{R}) = \frac{p(\mathbf{R}|\boldsymbol{\theta})\, p(\boldsymbol{\theta})}{p(\mathbf{R})}. \tag{IV.1}$$

Equation (IV.1) relates the posterior distribution, $p(\boldsymbol{\theta}|\mathbf{R})$ to the likelihood $p(\mathbf{R}|\boldsymbol{\theta})$, the prior $p(\boldsymbol{\theta})$ and the evidence $p(\mathbf{R})$. Here, we only consider one single model, i.e., the fBm, and thus do not explicitly refer to it. The principle of amortized inference [39] is to split the estimation of the posterior $p(\boldsymbol{\theta}|\mathbf{R})$ into two independent steps. The first is computationally costly and involves learning an approximate posterior density $\hat{p}(\boldsymbol{\theta}|\mathbf{R})$ from numerically generated

data. Then, the second step consists in running the pre-trained approximate system on the experimental data to infer the posterior density, assuming that they are similar to the training data.

A tractable likelihood can be computed for fBm. We consider a trajectory $\mathbf{R} = (\mathbf{r}_0, \mathbf{r}_2, \dots, \mathbf{r}_N)$ to be a 1-dimensional time-series of positions $\mathbf{r}_i$ recorded at equidistant points in time $t_i \in \{0, \Delta t, 2\Delta t, \dots, N\Delta t\}$. The likelihood of a trajectory reads

$$p\left(\mathbf{R}|\,\boldsymbol{\theta}\right) = \frac{1}{(2\pi)^{N/2}\sqrt{\det\Sigma(\boldsymbol{\theta})}}\exp\left(-\frac{1}{2}\left(\Delta\mathbf{r}\right)^{\top}\Sigma(\boldsymbol{\theta})^{-1}\Delta\mathbf{r}\right), \tag{IV.2}$$

where $\Delta\mathbf{r} = (\Delta\mathbf{r}_1, .., \Delta\mathbf{r}_N)^{\top}$, with $\Delta\mathbf{r}_i = \mathbf{r}_i - \mathbf{r}_{i-1}$ the individual displacements. Then, $\boldsymbol{\theta} = (K_\alpha, \alpha)$ are the fBm's parameters to infer, and Σ is the displacements' covariance matrix whose coefficients are given by

$$[\Sigma(\boldsymbol{\theta})]_{ij} = K_\alpha \Delta t^\alpha \left(|i-j+1|^\alpha + |i-j-1|^\alpha - 2|i-j|^\alpha\right).$$

We choose to rely on a likelihood-free approach to amortize our inference procedure. This may seem counter-intuitive for the precise case of fBm because the likelihood is analytically tractable, but this method has the advantage of relying solely on computations of linear complexity. Furthermore, the approach is also directly portable to more complex problems for which a tractable likelihood may not be available or may be too computationally costly. Indeed, likelihood-free inference is a method of choice to address such problems. As more and more complex models are encountered in numerous fields of science, the field of simulation-based inference [39] is growing very rapidly to address the associated challenging inverse problems. The shift towards amortization of the likelihood is notably driven by new tools and conceptual approaches derived from machine learning [40, 41].

The architecture of the amortized model of the posterior distribution is shown in Figure IV.1. It is based on the recently introduced Bayes Flow (BF) [42] procedure. In this framework, a first neural network, working as an encoder, creates a fixed-dimension vector of summary statistics from a set of observations. In our case, the encoder takes the form of a GNN (Fig. IV.1A). The encoder's output, the summary statistics vector, parametrizes an invertible transformation between easily sampled distributions (Gaussian) and the posterior distribution of the parameters (Fig. IV.1B). The full procedure generates the posterior distribution of the parameters. Such flow-based approaches, derived from normalizing flows [43], have the advantage that they provide an estimation of the posterior without requiring extensive sampling. The whole module is trained on numerically generated data and can then be used for inference. In the two following subsections, we first present the GNN (Section IV.2.1) and then the invertible network (Section IV.2.2).

IV.2.1 Graph neural network for learning summary statistics

The architecture of the summary network is similar to the one introduced in III. Here, we emphasize on the relevance of given choices for addressing this specific inference problem.

The features vector of a given edge e, $\mathbf{y}_e^{(0)}$, of size n_y, encapsulates information about the trajectory's course between the two nodes i and j it connects, such as the normalized time difference $(j-i)/N$ and various distance measures. The wiring scheme of a trajectory's graph, with exponentially spaced time intervals covering the full length of a trajectory, is

particularly suited to the fBM as its correlations are scale-free. Distance-based features are normalized so that they do not depend on the trajectory length. This is done in order to facilitate the generalization capability of the network. Although the training of the GNN is dimension-specific, its architecture can be adapted to trajectories of any dimension by adapting the features' initialization. A key point about the graph construction procedure is that it has linear algorithmic complexity (see details in IV.3).

Following the graph initialization step, the summary network performs several graph convolution operations [44, 45, 46]. It then passes the learnt node feature vectors as inputs to a pooling layer that aggregates features across all nodes of a trajectory graph into a fixed-length vector. The vector is finally passed through a multi-layer perceptron to obtain the summary statistics vector $\mathbf{h} = g_\psi(\mathbf{R})$, where ψ denotes the neural network coefficients. We refer the interested reader to [38] for details about the graph generation and GNN implementation.

IV.2.2 Invertible network for generating a variational posterior density

The Bayes Flow approach provides an invertible transformation, $f_\phi(\cdot;\mathbf{h})$, between the parameter space (in $\mathbb{R}^D$, with $D \geq 2$) and the prior space (in $\mathbb{R}^D$), on which a D-dimensional standard Gaussian density is assumed. The transformation $f_\phi(\cdot;\mathbf{h})$ is parametrized by a conditional invertible neural network (cINN) [47] made of a succession of affine coupling blocks [48] (multiple blocks sequentially applied) and maps $\boldsymbol{\theta}$ to the prior conditioned on $\mathbf{h}$, the summary statistics of the trajectory.

By design, these blocks can be inverted and the determinant of the Jacobian matrix $\mathbf{J}_{f_\phi}$ of the transformation is retrieved from the forward pass. During training we seek to approximate the true posterior $p(\theta|\mathbf{R})$ by the learnt posterior $p_\phi(\theta|\mathbf{R}) = \exp\left(-\frac{\|f_\phi(\boldsymbol{\theta};\mathbf{h})\|_2^2}{2}\right)$. The loss function is the Kullback-Leibler divergence between $p(\theta|\mathbf{R})$ and $p_\phi(\theta|\mathbf{R})$ which reads as

$$\mathcal{L}(\mathbf{R}) = \frac{1}{2}\|f_\phi(\boldsymbol{\theta};\mathbf{h})\|_2^2 - \log|\det \mathbf{J}_{f_\phi}|,$$

where $\mathbf{h} = g_\psi(\mathbf{R})$. Sampling the posterior distribution consists in computing $\mathbf{h}$ from the trajectory $\mathbf{R}$, and generating the required number of sample as $\boldsymbol{\theta} = f_\phi^{-1}(\mathbf{z};\mathbf{h})$ with $\mathbf{z}$ generated from a standard D-dimensional Gaussian distribution.

IV.3 Estimation of the anomalous exponent

We evaluate the performance of our amortized inference procedure on numerically generated trajectories. Estimating the anomalous exponent α is the most challenging part of the inference, and we thus focus on this here, but our approach infers a joint posterior density for $\boldsymbol{\theta} = (K_\alpha, \alpha)$. Figure IV.2A shows the inferred posteriors of α on portions of increasing length of two example trajectories. The amortized posterior is consistent with the exact one (See Supplementary Material 2.). Both become increasingly peaked around the true value of α as the length of the trajectory increases. The inferred posterior distributions do not exhibit broad tails or divergences, and are thus proper distributions, i.e., they are normalizable.

We show the precision of the inference on trajectories with lengths varying across two orders of magnitudes. Both the variance of the approximated posterior and the square error of the estimator follow a power-law decrease, as can be seen Fig. IV.2B. Using the

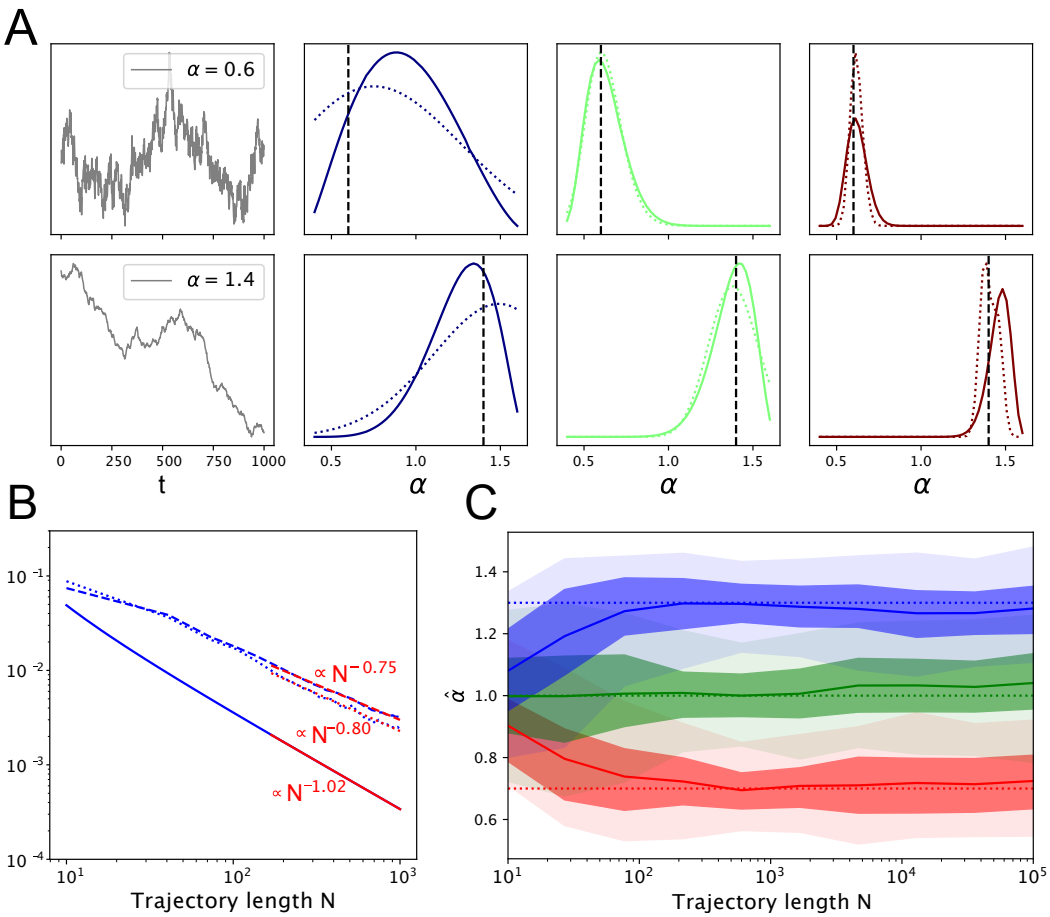

Figure IV.2: **Model performance.** A: Evolution of the posterior density of α inferred by the model (plain lines) versus the true posterior (dotted lines) from two example trajectories with $\alpha = 0.6$ (top) and $\alpha = 1.4$ (bottom), respectively. The length of the portion of the trajectory used for inference is increased ten-fold between each panel, i.e., from left to right: $N = 10$ (blue), 100 (green), 1000 (red). B: Evolution of the variance of the estimator $\hat{\alpha}$ (dashed line), the mean square error of the mean posterior $\hat{\alpha}$ (dotted line) and the Cramér-Rao bound for an unbiased estimator of α, for increasing values of trajectory length N. In red: power-law fits on large values of N C: Convergence towards the true value of α as function of trajectory length. The model was trained on trajectories of length $10 \leq N \leq 1,000$. Darker zones correspond to the first and third quartiles, while lighter ones correspond to 5% and 95% quantiles We modified the normalization procedure so that it is able to generalize to trajectories longer than those seen during training.

exact likelihood shown in Eq. IV.2 we can evaluate the Cramér-Rao bound and compare the amortised inference's performance to it. The amortised inference is suboptimal (as expected from a variational inference), but its variance shows a fast decreasing trend similar to the Cramér-Rao bound, i.e., close to $\propto 1/N$.

We looked at the learnt summary statistics h, that after training constitutes a low-dimensional representation of the random walks which is use as features to compute the posterior distribution. The latent representation can be interpreted for its own sake as the way the encoder represents information about the trajectories. An assumption in representation learning [49] is that interpretable representation lead to better generalisation. We projected h onto a 2D plane using UMAP [50] (a non-linear dimenstionality reduction algorithm) and mapped α on it (see Supplementary Fig. IV.4). We see that the latent space is organised according the value of α, a good indication that the learning process properly captured the underlying physical properties. We tested the robustness of the inference procedure when applied to trajectories corrupted by positional noise. We show in Supplementary Fig. IV.5 the evolution of the mean square error of the amortised inference of α and compare it with the corresponding Cramér-Rao bound. The precision of the amortised inference procedure closely follows the lower bound set by the Cramér-Rao inequality. This was obtained by training models specifically on trajectories corrupted with increasing amounts of noise.

The summary network's architecture, with normalized initial features, leads to an approximately "length-invariant" inference, i.e., the vector of summary statistics captures relevant information regardless the length of the trajectories. Hence, the approach is not limited to trajectories seen during training. We show in Figure IV.2C an example of application for trajectories a hundred times longer than the maximal ones the inference procedure was trained on. The inferred $\hat{\alpha}$ for long trajectories were still well ordered, but suffered from a slight bias, which we corrected using a simple polynomial of degree 3.

An important attribute of the amortized approach is that is has linear computational complexity at inference time, i.e., when applied to infer the parameters of a random walk. To show this, we subdivide the amortised inference procedure into three steps: (i) initial feature evaluation, (ii) forward pass through graph convolutions and pooling, and (iii) operations on summary statistics to generate the posterior. (i) The initial evaluation of node and edge features requires $O(N + |E|)$ time and memory, where N is the number of nodes (for a trajectory of $N+1$ points) and $|E|$ is the number of edges. Here $|E| \propto N$ by design (the in-degree of nodes is bounded), so this step has $O(N)$ complexity. (ii) The forward pass through the graph convolutions and the following pooling of node features requires $O(|E|)$ operations and memory slots, and hence this step also has $O(N)$ complexity. (iii) The latent space is of fixed dimensions, and hence all operations after the pooling layer have $O(1)$ complexity. The global complexity of the amortised architecture is thus linear with respect to the number of points in the trajectory.

In comparison, calculating the exact likelihood [Eq. (IV.2)] requires evaluating the determinant $\det\Sigma(\boldsymbol{\theta})$ and the quadratic form $(\Delta\mathbf{r})^\top\Sigma(\boldsymbol{\theta})^{-1}\Delta\mathbf{r}$, which can be done in $O(N^2)$ time [51]. This makes exact inference prohibitively expensive for very long trajectories, where our amortized inference scheme may instead be used (Fig. IV.2C). Note furthermore that for many models the exact likelihood cannot be calculated at all, in which case approximate inference is the only route possible. In all of the above cases, our amortized inference scheme retains its linear computational complexity.

IV.4 Estimation of a finite decorrelation time

When considering fBm as a model of biomolecule random walks, we have to keep in mind that many physical environments might exhibit a finite decorrelation time τ_c possibly stemming from motion occurring outside a polymer-dominated environment [52] or from changes of conformations of the bimolecule altering the nature of its interactions The characteristic time bears information on the local environment's physical properties, and it might be spatially dependent or specific to interactions with local partners. In practice, inferring τ_c from individual trajectories is challenging. Autocorrelation-based approaches for example give incomplete results on individual trajectories as the limited number of points prevents proper averaging [53, 54].

We adapted our amortised inference procedure to infer (α, τ_c) instead of (K_α, α). We left out K_α here since it is simply a scale factor and can be removed by rescaling the trajectories. We used the same node and edge features as above, and we thus conserve the procedure's linear computational complexity. A finite decorrelation time was modeled by multiplying the autocovariance of the fBm by an exponential factor, $\min(1, e^{\tau_c - \tau})$, where τ is the time difference. Examples of the autocorrelation function for several values of α and τ_c are given in Supplementary Fig. IV.6. The modified covariance matrix thus reads

$$\begin{aligned}[\Sigma(\alpha, \tau_c)]_{i,j} \quad &= \min\left(1, e^{\tau_c - |i-j|\Delta t}\right) \times \\ &\left(|i-j+1|^\alpha + |i-j-1|^\alpha - 2|i-j|^\alpha\right),\end{aligned}$$

where we have ignored the scale factor $K_\alpha \Delta t^\alpha$.

Here, we provide example we performed this inference solely on trajectories of length 1,000 with τ_c integer-valued and ranging from 5 to 50.

We compared our estimator with the maximum likelihood estimator, obtained by choosing the value of (α,τ_c) that maximizes the likelihood of the observed trajectory.

To optimize the likelihood in practice, we computed the log-likelihood on a grid of values, of α and τ_c, with α taking 30 regularly spaced values between 0.4 and 1.6, and τ_c taking all possible values in its range. As shown in Figure IV.3B (upper panel), our amortised inference yields a slightly more biased estimate than the maximum-likelihood estimator (when taking the mean of the posterior distribution) but has a smaller variance. When $\alpha = 1$, successive increments are completely independent of each other and there is thus no information to retrieve regarding τ_c. This is observable on the lower panel of Figure IV.3, both by looking at the Cramér-Rao bound, which diverges, and at the variance of our estimator, which is maximal at $\alpha = 1$.

IV.5 Discussion

Simulation-based inference coupled with machine learning are a promising avenue to address challenging inverse problems. When applied to intractable systems, this combination allows splitting the inference task into two steps. In the first, computationally intensive simulations produce artificial data. These data are used to train neural networks to approximate the posterior distribution of the parameters using a variational objective. In the second step, which is computationally fast, inference is performed on experimental data and the posterior distribution is evaluated by a direct forward pass through the trained neural networks. The

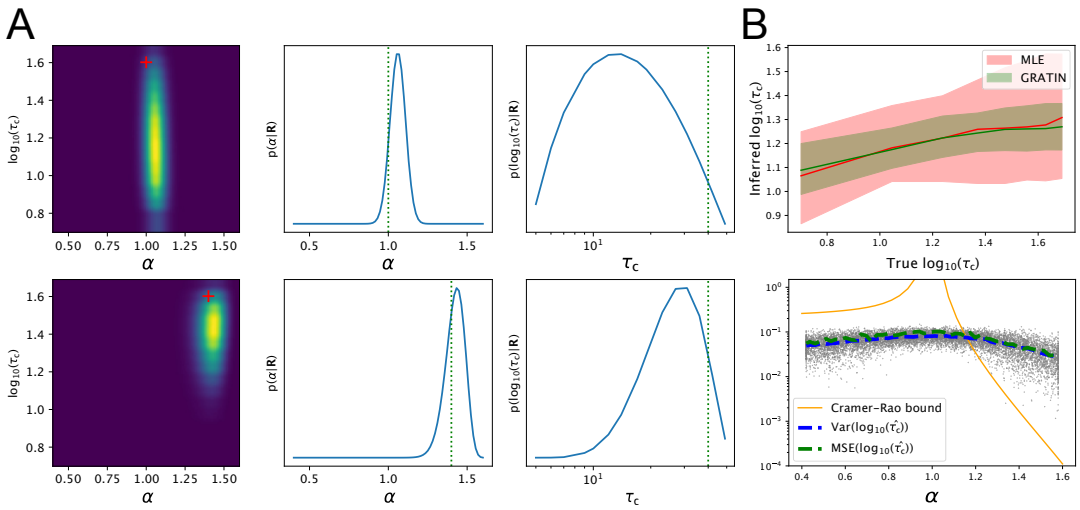

Figure IV.3: **Performance of the anomalous exponent and decorrelation time estimation** A: Posteriors of α and $\log_{10}(\tau_c)$ for two trajectories with different α (plain lines). Dashed green lines indicate true parameter values. B: Top: Comparison of the values of $\log(\tau_c)$ inferred by our method (in green) and by a maximum-likelihood estimator (in red). The thick line represents the mean across all trajectories, while the filled regions correspond to the first and last quartiles. Bottom: Variance and mean square error of our inference of $\log_{10}(\tau_c)$ plotted as a function of α, compared with the Cramér-Rao bound for an unbiased estimator.

procedure is statistically efficient if the numerical data match the properties of experimental one and if the variational inference is able to capture the complex relations that might exist between the variables to be inferred.

There are two main challenges associated with amortised approaches. First, training variational inference often consists in minimising a Kullback-Liebler distance between the approximate distribution and the real (unknown) one [55]. Optimising such a non-convex function is challenging and is not generally guaranteed to converge towards a global optimum. The second challenge is linked to interpretability. Both the models used to learn the summary statistics and the variational posterior distribution are generally intractable. There is thus limited insurance that the process does not misbehave, especially when applied to real experimental data. Evaluation of the exact posterior distribution using sampling, such as in approximate Bayesian computation, may however lead to similar problems due to the difficulty of properly sampling complex likelihood landscapes.

We here used fBm to quantify the performance of our amortised inference approach. We chose to focus on fBm both due to its paradigmatic status as an anomalous random walk model and because it has a tractable likelihood, allowing us to compare our amortised method to exact likelihood-based inference and to the Cramér-Rao bound on estimator precision. We advocate more generally for the use of exactly solvable random walk models, such as the fBm, as benchmarks to evaluate the performance of machine-learning based inference methods.

We showed that our amortized inference can successfully be applied to infer the parameters of fBm, with a precision that is lower than the Cramér-Rao bound but which increases

with a scaling that is similar to it. Our algorithm has a linear complexity in the length of the trajectory and can be applied to trajectories of any length at inference time, even if the algorithm has not been specifically trained on trajectories of the same length. We furthermore showed that our amortised approach could be used to efficiently infer the parameters of a more realistic fBm-type model with a finite decorrelation time.

Our amortised inference framework can be used for any random walk model, even for models that do not have a tractable likelihood, provided that they can be simulated efficiently enough to provide a large number of trajectories for training. In all cases, our approach retains its linear computational complexity at inference time. For random walk models with intractable likelihoods, only empirical evaluation of the performance will in general be possible. Thus, it is not possible to make absolute statements about the statistical efficiency of the approach in these cases.

Beyond random walks, amortized inference can more generally be instrumental in providing posterior distributions for models of complex systems with fractional noise and/or long memory. Numerous challenges have to be addressed to standardise the optimisation of the variational inference, especially in cases where some parameters are not sufficiently constrained by data or when there are sloppy directions in the parameter space [56]. Furthermore, variational inference does not necessarily lead to physically realistic parameters. Ensuring the physics-informed [57, 58] nature of the inference may require imposing constraints on the network generating the summary statistics. Though our results show that the network is able to learn physically meaningful features without inductive bias. Finally, the statistical efficiency of amortised approaches will depend on the ability of numerically generated data to match experimental observations.

IV.6 Supplementary material

IV.6.1 Amortized inference model architecture and training

a) Node and edge features

The features associated to each node $i \in \{1, \dots, N\}$ in the graph of a trajectory $(\mathbf{r}_0, \dots, \mathbf{r}_N)$ of length N are:

1. the normalized time: $t = i/N$;
2. denoting $S_i^d = \sum_{k \leq i} \|\Delta \mathbf{r}_k\|_2^d$ the the sum of step lengths to the power d, we add as node features the following ratios for $d \in \{1, 2, 4\}$: $t^{-1} S_i^d / S_N^d$
3. the distance to origin, normalized (up to a constant factor) by the square root of the expected square displacement for a Brownian motion: $\|\mathbf{r}_i\| / \sqrt{N \operatorname{Var}(\Delta \mathbf{r}) t}$
4. the maximal distance to origin up to i, normalized by the same factor: $\max_{k \leq i} \|\mathbf{r}_k\| / \sqrt{N \operatorname{Var}(\Delta \mathbf{r}) t}$

The features associated to an edge $e_{i,j}$ with $i < j$ are:

1. the normalized time difference: $\Delta t = (j - i)/N$;
2. the distance, normalized (up to a constant factor) by the square root of the expected square displacement for a Brownian motion: $\|\mathbf{r}_j - \mathbf{r}_i\| / \sqrt{|i - j| \Delta t \operatorname{Var}(\Delta r)}$

3. for $d \in \{1, 2, 4\}$: $(\Delta t)^{-1}(S_i^d - S_j^d)/S_N^d$;

4. the dot product of jumps, normalized by the variance of step sizes: $\Delta\mathbf{r}_i^\top \Delta\mathbf{r}_j / \operatorname{Var}(\Delta\mathbf{r})$;

The computation of each features is done in linear or constant time. In particular, the last two features are calculated in linear complexity by leveraging the fact that they equal to the differences of two quantities which each solely depends on i or j.

b) GNN Architecture

The architecture of the GNN used in the summary network is similar to the encoder network proposed in [38], with the difference that we here additionally apply edge features. Node and edge features are first passed to perceptrons, which embeds them in a 32-dimensional space. The network is then composed of three successive convolution layers (one taken from [59] and two edge-conditioned layers taken from [60]) outputting node features matrices $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$, each of 32 dimensions, which are summed to form $\mathbf{x}^{(f)}$. The rows of this matrix of nodes features are then averaged during the pooling step, to keep just one row per graph, i.e., per trajectory. This vector is subsequently passed to a three-layer perceptron, the output of which is the summary statistics vector.

c) Invertible network

The invertible network is a succession of three affine coupling blocks. These blocks, introduced in [48], transform an input vector $\mathbf{u}$ into $\mathbf{v}$ in an invertible manner parametrized by the summary statistics vector $\mathbf{h}$. They do so by splitting $\mathbf{u}$ into two halves $\mathbf{u}_1$ and $\mathbf{u}_2$, used to compute the two halves $\mathbf{v}_1$ and $\mathbf{v}_2$ of $\mathbf{v}$ by consecutively performing the two following operations:

$$\begin{aligned}\mathbf{v}_1 &= \mathbf{u}_1 \odot \exp\left(s_1\left(\mathbf{u}_2; \mathbf{h}\right)\right) + t_1(\mathbf{u}_2; \mathbf{h}) \\ \mathbf{v}_2 &= \mathbf{u}_2 \odot \exp\left(s_2\left(\mathbf{v}_1; \mathbf{h}\right)\right) + t_2(\mathbf{v}_1; \mathbf{h})\end{aligned}$$

where $\odot$ denotes the element-wise multiplication (the Hadamart product) and where s_1, s_2, t_1 and t_2 are multi-layer perceptrons, which do not need to be invertible. In our case, they have five hidden layers and their activation function is an exponential linear unit. This procedure can be inverted to efficiently retrieve $\mathbf{u}$ from $\mathbf{v}$.

d) Training the networks

The networks are trained for 50 epochs during which they are presented 10^5 trajectories (generated on the fly using ten CPUs), grouped by batches of 1 024, so as to ensure a good estimation of the gradients and of the normalization statistics. We use a learning rate of 10^{-2} which we reduce by a factor of 5 when the mean error on the prediction of α (estimated on a fixed validation dataset at the end of each epoch) reaches a plateau, with a patience of ten epochs and a "cooldown" of five epochs. The training lasts for approximately five hours with a P6000 GPU.

IV.6.2 Exact posterior inference

To compute exact posteriors, likelihood values were computed on grids of points in parameter space. We picked a uniform prior on $(0.1, 1.9)$ for α and a log-uniform one for τ_c and K_α, which spanned 8 orders of magnitude. There was thus no coupling between parameters in the priors. The parameters used to generate trajectories during training were sampled from these same priors.

IV.6.3 Cramér-Rao bound

Formally, we consider any estimator of the parameters $\boldsymbol{\theta}$ to be a (possibly implicit) function of the recorded trajectory, $\mathbf{R}$, i.e., $\hat{\boldsymbol{\theta}} = \mathbf{T}(\mathbf{R})$. We denote by $\boldsymbol{\psi}(\boldsymbol{\theta}) = E[\mathbf{T}(\mathbf{R})]$ its expectation, and by $\Gamma(\boldsymbol{\theta}) = E[(\mathbf{T}(\mathbf{R}) - \boldsymbol{\psi}(\boldsymbol{\theta}))(\mathbf{T}(\mathbf{R}) - \boldsymbol{\psi}(\boldsymbol{\theta}))^\top]$ its covariance matrix. Finally, $\mathbf{I}(\boldsymbol{\theta})$ is the Fischer information matrix, whose elements are given by $\mathbf{I}_{n,m}(\boldsymbol{\theta}) = E\left[\frac{\partial}{\partial\theta_n}\log p(\mathbf{R}|\,\boldsymbol{\theta})\frac{\partial}{\partial\theta_m}\log p(\mathbf{R}|\,\boldsymbol{\theta})\right]$.

The Cramér-Rao bound states that, for any unbiased estimator $\mathbf{T}$,

$$\Gamma(\boldsymbol{\theta}) \geq \nabla\boldsymbol{\psi}(\boldsymbol{\theta})\left[\mathbf{I}(\boldsymbol{\theta})\right]^{-1}\left[\nabla\boldsymbol{\psi}(\boldsymbol{\theta})\right]^\top,$$

where $\nabla\boldsymbol{\psi}$ is the Jacobian of $\boldsymbol{\psi}$. In particular, this matrix inequality implies the following lower bound on the variance of any unbiased estimator of a single parameter:

$$\mathrm{Var}_{\boldsymbol{\theta}}(T_n(\mathbf{R})) \geq \left[\mathbf{I}(\boldsymbol{\theta})^{-1}\right]_{n,n}$$

IV.6.4 Supplementary figures

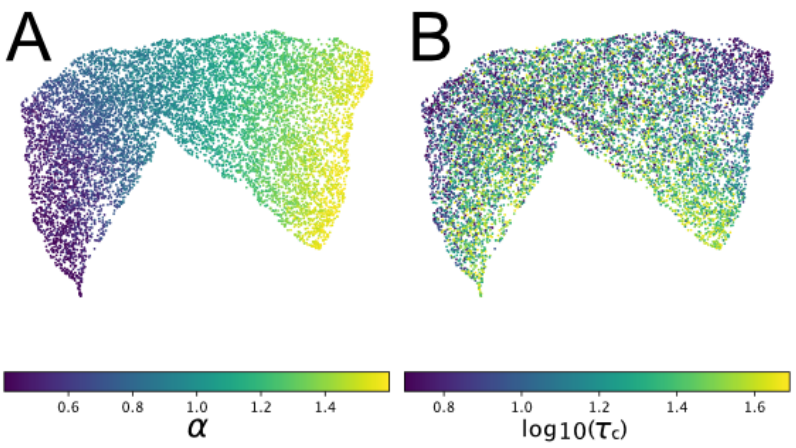

Figure IV.4: **Latent space representations of individual trajectories.** 2D visualisation of summary vectors (one point per trajectory), obtained by UMAP and colored according to A: their anomalous diffusion exponent α, and B: their correlation time τ_c. Trajectories are of length $N = 1,000$.

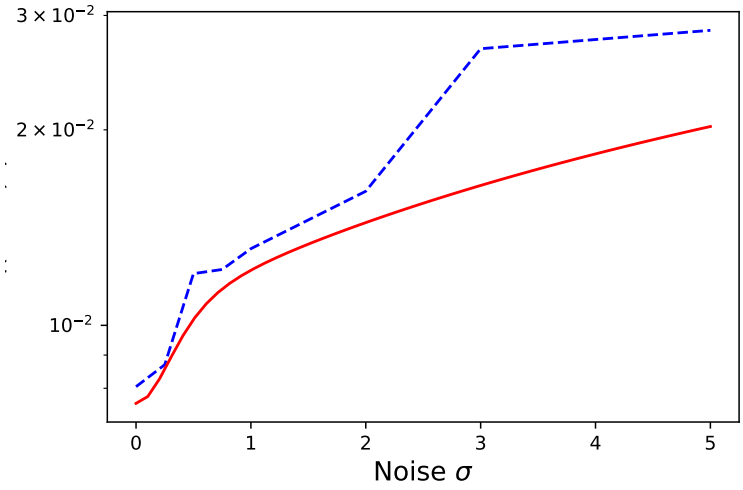

Figure IV.5: **Robustness to noise.** Mean square error on α estimated with amortised inference compared to the Cramér-Rao bound as a function of positioning noise σ. Trajectories are of length $N = 200$ and generalised diffusivity 1. Positions are independently corrupted with Gaussian noise of variance σ^2.

Bibliography

[1] **Verdier Hippolyte**, François Laurent, Alhassan Cassé, Christian Vestergaard, and Jean-Baptiste Masson. Amortised inference of fractional brownian motion with linear computational complexity. 2022.

[2] Gardiner Crispin. *Handbook of Stochastic Methods: for Physics, Chemistry and natural sciences*. Springer, 4th edition, .

[3] Benoit B. Mandelbrot and John W. Van Ness. Fractional brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, October 1968. doi: 10.1137/1010093. URL https://doi.org/10.1137/1010093.

[4] Jean-Philippe Bouchaud and Antoine Georges. Anomalous diffusion in disordered media: Statistical mechanisms, models and physical applications. *Physics Reports*, 195(4-5):127–293, November 1990. doi: 10.1016/0370-1573(90)90099-n. URL https://doi.org/10.1016/0370-1573(90)90099-n.

[5] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, 1994. doi: 10.1109/90.282603. URL https://doi.org/10.1109/90.282603.

[6] Nigel J. Cutland, P. Ekkehard Kopp, and Walter Willinger. Stock price returns and the joseph effect: A fractional version of the black-scholes model. In *Seminar on Stochastic Analysis, Random Fields and Applications*, pages 327–351. Birkhäuser Basel, 1995. doi: 10.1007/978-3-0348-7026-9_23. URL https://doi.org/10.1007/978-3-0348-7026-9_23.

[7] V. Kukla, J. Kornatowski, D. Demuth, I. Girnus, H. Pfeifer, L. V. C. Rees, S. Schunk, K. K. Unger, and J. Karger. NMR studies of single-file diffusion in unidimensional channel zeolites. *Science*, 272(5262): 702–704, May 1996. doi: 10.1126/science.272.5262.702. URL https://doi.org/10.1126/science.272.5262.702.

[8] Laurent Decreusefond and Ali Suleyman Üstünel. Fractional brownian motion: theory and applications. *ESAIM: Proceedings*, 5:75–86, 1998. doi: 10.1051/proc:1998014. URL https://doi.org/10.1051/proc:1998014.

[9] JP Bouchaud. *Theory of financial risk and derivative pricing : from statistical physics to risk management*. Cambridge University Press, Cambridge, 2003. ISBN 978-0521819169.

[10] Stephanie C. Weber, Andrew J. Spakowitz, and Julie A. Theriot. Bacterial chromosomal loci move subdiffusively through a viscoelastic cytoplasm. *Physical Review Letters*, 104(23), June 2010. doi: 10.1103/physrevlett.104.238102. URL https://doi.org/10.1103/physrevlett.104.238102.

[11] J. L. A. Dubbeldam, V. G. Rostiashvili, A. Milchev, and T. A. Vilgis. Fractional brownian motion approach to polymer translocation: The governing equation of motion. *Physical Review E*, 83(1), January 2011. doi: 10.1103/physreve.83.011802. URL https://doi.org/10.1103/physreve.83.011802.

[12] Jae-Hyung Jeon, Vincent Tejedor, Stas Burov, Eli Barkai, Christine Selhuber-Unkel, Kirstine Berg-Sørensen, Lene Oddershede, and Ralf Metzler. In VivoAnomalous diffusion and weak ergodicity breaking of lipid granules. *Physical Review Letters*, 106(4), January 2011. doi: 10.1103/physrevlett.106.048103. URL https://doi.org/10.1103/physrevlett.106.048103.

[13] J.-C. Walter, A. Ferrantini, E. Carlon, and C. Vanderzande. Fractional brownian motion and the critical dynamics of zipping polymers. *Physical Review E*, 85(3), March 2012. doi: 10.1103/physreve.85.031120. URL https://doi.org/10.1103/physreve.85.031120.

[14] Dominique Ernst, Marcel Hellmann, Jürgen Köhler, and Matthias Weiss. Fractional brownian motion in crowded fluids. *Soft Matter*, 8(18):4886, 2012. doi: 10.1039/c2sm25220a. URL https://doi.org/10.1039/c2sm25220a.

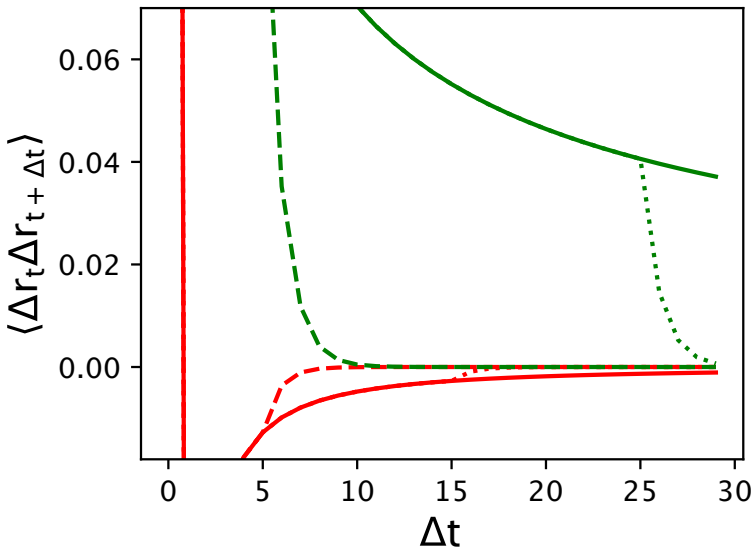

Figure IV.6: **Temporal correlations of fBm with finite decorrelation time.** Autocovariance of increments of the fBm trajectory, with finite correlation time τ_c, in the sub-diffusive and super-diffusive case. Red curves correspond to $\alpha = 0.6$, and $\tau_c = 5$ (dashed line), 15 (dotted line), ∞ (plain line). Green curves correspond to $\alpha = 1.4$, and $\tau_c = 5$, 25, ∞.

[15] S. Rostek and R. Schöbel. A note on the use of fractional brownian motion for financial modeling. *Economic Modelling*, 30:30–35, January 2013. doi: 10.1016/j.econmod.2012.09.003. URL https://doi.org/10.1016/j.econmod.2012.09.003.

[16] Vladimir V. Palyulin, Tapio Ala-Nissila, and Ralf Metzler. Polymer translocation: the first two decades and the recent diversification. *Soft Matter*, 10(45):9016–9037, 2014. doi: 10.1039/c4sm01819b. URL https://doi.org/10.1039/c4sm01819b.

[17] Avelino Javer, Nathan J. Kuwada, Zhicheng Long, Vincenzo G. Benza, Kevin D. Dorfman, Paul A. Wiggins, Pietro Cicuta, and Marco Cosentino Lagomarsino. Persistent super-diffusive motion of escherichia coli chromosomal loci. *Nature Communications*, 5(1), May 2014. doi: 10.1038/ncomms4854. URL https://doi.org/10.1038/ncomms4854.

[18] Daniel Han, Nickolay Korabel, Runze Chen, Mark Johnston, Anna Gavrilova, Victoria J Allan, Sergei Fedotov, and Thomas A Waigh. Deciphering anomalous heterogeneous intracellular transport with neural networks. *eLife*, 9, March 2020. doi: 10.7554/elife.52224. URL https://doi.org/10.7554/elife.52224.

[19] Wei Wang, Andrey G Cherstvy, Aleksei V Chechkin, Samudrajit Thapa, Flavio Seno, Xianbin Liu, and Ralf Metzler. Fractional brownian motion with random diffusivity: emerging residual nonergodicity below the correlation time. *Journal of Physics A: Mathematical and Theoretical*, 53(47):474001, November 2020. doi: 10.1088/1751-8121/aba467. URL https://doi.org/10.1088/1751-8121/aba467.

[20] Marco Gherardi, Ludovico Calabrese, Mikhail Tamm, and Marco Cosentino Lagomarsino. Model of chromosomal loci dynamics in bacteria as fractional diffusion with intermittent transport. *Physical Review E*, 96(4), October 2017. doi: 10.1103/physreve.96.042402. URL https://doi.org/10.1103/physreve.96.042402.

[21] Maxence Arutkin, Benjamin Walter, and Kay Jörg Wiese. Extreme events for fractional brownian motion with drift: Theory and numerical validation. *Physical Review E*, 102(2), August 2020. doi: 10.1103/physreve.102.022102. URL https://doi.org/10.1103/physreve.102.022102.

[22] Shi Yu, Jiaxin Wu, Xianliang Meng, Ruizhi Chu, Xiao Li, and Guoguang Wu. Mesoscale simulation of bacterial chromosome and cytoplasmic nanoparticles in confinement. *Entropy*, 23(5):542, April 2021. doi: 10.3390/e23050542. URL https://doi.org/10.3390/e23050542.

[23] Ralf Metzler, Jae-Hyung Jeon, Andrey G. Cherstvy, and Eli Barkai. Anomalous diffusion models and their properties: non-stationarity, non-ergodicity, and ageing at the centenary of single particle tracking. *Phys. Chem. Chem. Phys.*, 16(44):24128–24164, 2014. doi: 10.1039/c4cp03465a. URL https://doi.org/10.1039/c4cp03465a.

[24] Weihua Deng and Eli Barkai. Ergodic properties of fractional brownian-langevin motion. *Physical Review E*, 79(1), January 2009. doi: 10.1103/physreve.79.011112. URL https://doi.org/10.1103/physreve.79.011112.

[25] Stas Burov, Jae-Hyung Jeon, Ralf Metzler, and Eli Barkai. Single particle tracking in systems showing anomalous diffusion: the role of weak ergodicity breaking. *Physical Chemistry Chemical Physics*, 13(5):1800, 2011. doi: 10.1039/c0cp01879a. URL https://doi.org/10.1039/c0cp01879a.

[26] Jae-Hyung Jeon and Ralf Metzler. Inequivalence of time and ensemble averages in ergodic systems: Exponential versus power-law relaxation in confinement. *Physical Review E*, 85(2), February 2012. doi: 10.1103/physreve.85.021147. URL https://doi.org/10.1103/physreve.85.021147.

[27] Elvis Geneston, Rohisha Tuladhar, M. T. Beig, Mauro Bologna, and Paolo Grigolini. Ergodicity breaking and localization. *Physical Review E*, 94(1), July 2016. doi: 10.1103/physreve.94.012136. URL https://doi.org/10.1103/physreve.94.012136.

[28] Hanna Loch-Olszewska, Grzegorz Sikora, Joanna Janczura, and Aleksander Weron. Identifying ergodicity breaking for fractional anomalous diffusion: Criteria for minimal trajectory length. *Physical Review E*, 94(5), November 2016. doi: 10.1103/physreve.94.052136. URL https://doi.org/10.1103/physreve.94.052136.

[29] Jochen Kursawe, Johannes Schulz, and Ralf Metzler. Transient aging in fractional brownian and langevin-equation motion. *Physical Review E*, 88(6), December 2013. doi: 10.1103/physreve.88.062124. URL https://doi.org/10.1103/physreve.88.062124.

[30] Yasmine Meroz and Igor M. Sokolov. A toolbox for determining subdiffusive mechanisms. *Physics Reports*, 573:1–29, April 2015. doi: 10.1016/j.physrep.2015.01.002. URL https://doi.org/10.1016/j.physrep.2015.01.002.

[31] T. Kosztołowicz, K. Dworecki, and St. Mrówczyński. How to measure subdiffusion parameters. *Physical Review Letters*, 94(17), May 2005. doi: 10.1103/physrevlett.94.170602. URL https://doi.org/10.1103/physrevlett.94.170602.

[32] Lloyd P. Sanders and Tobias Ambjörnsson. First passage times for a tracer particle in single file diffusion and fractional brownian motion. *The Journal of Chemical Physics*, 136(17):175103, May 2012. doi: 10.1063/1.4707349. URL https://doi.org/10.1063/1.4707349.

[33] Gorka Muñoz-Gil, Giovanni Volpe, Miguel Angel García-March, Ralf Metzler, Maciej Lewenstein, and Carlo Manzo. The anomalous diffusion challenge: single trajectory characterisation as a competition. *Emerging Topics in Artificial Intelligence 2020*, Aug 2020. doi: 10.1117/12.2567914. URL http://dx.doi.org/10.1117/12.2567914.

[34] Martin Lysy, Natesh S. Pillai, David B. Hill, M. Gregory Forest, John W. R. Mellnik, Paula A. Vasquez, and Scott A. McKinley. Model comparison and assessment for single particle tracking in biological fluids. *Journal of the American Statistical Association*, 111(516):1413–1426, October 2016. doi: 10.1080/01621459.2016.1158716. URL https://doi.org/10.1080/01621459.2016.1158716.

[35] Jens Krog, Lars H Jacobsen, Frederik W Lund, Daniel Wustner, and Michael A Lomholt. Bayesian model selection with fractional brownian motion. *Journal of Statistical Mechanics: Theory and Experiment*, 2018(9), September 2018. doi: 10.1088/1742-5468/aadb0e. URL https://doi.org/10.1088/1742-5468/aadb0e.

[36] Peter K. Koo and Simon G. J. Mochrie. Systems-level approach to uncovering diffusive states and their transitions from single-particle trajectories. *Physical Review E*, 94(5), November 2016. doi: 10.1103/physreve.94.052412. URL https://doi.org/10.1103/physreve.94.052412.

[37] Samudrajit Thapa, Michael A. Lomholt, Jens Krog, Andrey G. Cherstvy, and Ralf Metzler. Bayesian analysis of single-particle tracking data using the nested-sampling algorithm: maximum-likelihood model selection applied to stochastic-diffusivity data. *Physical Chemistry Chemical Physics*, 20(46):29018–29037, 2018. doi: 10.1039/c8cp04043e. URL https://doi.org/10.1039/c8cp04043e.

[38] Hippolyte Verdier, Maxime Duval, François Laurent, Alhassan Casse, Christian Lyngby Vestergaard, and J-B Masson. Learning physical properties of anomalous random walks using graph neural networks. *Journal of Physics A: Mathematical and Theoretical*, apr 2021. doi: 10.1088/1751-8121/abfa45. URL https://doi.org/10.1088/1751-8121/abfa45.

[39] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

[40] George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 837–848. PMLR, 16–18 Apr 2019. URL http://proceedings.mlr.press/v89/papamakarios19a.html.

[41] Justin Alsing, Tom Charnock, Stephen Feeney, and Benjamin Wandelt. Fast likelihood-free cosmology with neural density estimators and active learning. *Monthly Notices of the Royal Astronomical Society*, July 2019. doi: 10.1093/mnras/stz1960. URL https://doi.org/10.1093/mnras/stz1960.

[42] Stefan T. Radev, Ulf K. Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Kothe. BayesFlow: Learning complex stochastic models with invertible neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, 2020. doi: 10.1109/tnnls.2020.3042395. URL https://doi.org/10.1109/tnnls.2020.3042395.

[43] Ivan Kobyzev, Simon Prince, and Marcus Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. doi: 10.1109/tpami.2020.2992934. URL https://doi.org/10.1109/tpami.2020.2992934.

[44] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[45] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems*, 31, 2018.

[46] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds, 2018.

[47] Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W Pellegrini, Ralf S Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe. Analyzing inverse problems with invertible neural networks. *arXiv preprint arXiv:1808.04730*, 2018.

[48] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[49] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[50] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[51] John F Monahan. *Numerical methods of statistics*. Cambridge University Press, 2011.

[52] Stephanie C Weber, Andrew J Spakowitz, and Julie A Theriot. Bacterial chromosomal loci move subdiffusively through a viscoelastic cytoplasm. *Physical review letters*, 104(23):238102, 2010.

[53] Julia F Reverey, Jae-Hyung Jeon, Han Bao, Matthias Leippe, Ralf Metzler, and Christine Selhuber-Unkel. Superdiffusion dominates intracellular particle motion in the supercrowded cytoplasm of pathogenic acanthamoeba castellanii. *Scientific reports*, 5(1):1–14, 2015.

[54] Gardiner Crispin. *Handbook of Stochastic Methods: for Physics, Chemistry and natural sciences*. Springer, 4th edition, .

[55] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer. ISBN 0-387-31073-8. URL https://www.xarg.org/ref/a/0387310738/.

[56] Joshua J Waterfall, Fergal P Casey, Ryan N Gutenkunst, Kevin S Brown, Christopher R Myers, Piet W Brouwer, Veit Elser, and James P Sethna. Sloppy-model universality class and the vandermonde matrix. *Physical review letters*, 97(15):150601, 2006.

[57] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[58] Gert-Jan Both and Remy Kusters. Fully differentiable model discovery. *arXiv preprint arXiv:2106.04886*, 2021.

[59] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[60] Martin Simonovsky and Nikos Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3693–3702, 2017.

V – Maximum mean discrepancy approach to detect subtle changes in biomolecule dynamics

This chapter's content has been adapted from the following pre-publication.

Verdier Hippolyte, Françcois Laurent, Alhassan Cassé, Christian L Vestergaard, Christian G Specht, and Jean-Baptiste Masson. A maximum mean discrepancy approach reveals subtle changes in α-synuclein dynamics. *bioRxiv*, 2022

Contents

V.1 Introduction

Most inference schemes developed to address biomolecule trajectories properties focus on estimating physical parameters such as the diffusion coefficient, the anomalous diffusion exponent, the type of random walk model, or other ad-hoc quantities measuring particular

aspects of the dynamics. Here, instead of describing trajectories using a set of explicitly defined features, we rely on an *encoder* neural network, in order to characterise each trajectory by a *latent vector* of features. The goal of this encoder is to automatically learn optimised features that describe random walks beyond predefined canonical models and features.

We employ the encoder network to project recorded trajectories into the fixed-dimensional latent space. We develop a statistical test on this latent space to test for differences in dynamics between two sets of trajectories. Our methodology can in particular be used to compare dynamics observed in different biological conditions and different cell organelles, by comparing the sets of latent vectors computed from trajectories observed in the respective microscopy recordings or regions of the cell. The central advantage of the testing methodology we propose is that it is not dependent on the specification and selection of a model of the recorded random walks. This enables statistically robust testing of differing biological conditions, which are likely to induce different levels of cellular heterogeneity and do not necessarily generate canonical random walks.

We train the encoder network using a simulation-based inference framework (detailed below), allowing it to provide a representation of trajectories without assuming that they are realisations of a canonical random walk model. The subsequent statistical test seeks to differentiate the distributions of latent vectors coming from different conditions or organelles (or both). It is based on the maximum mean discrepancy () test [2], which uses a kernel approach to compare two distributions. This test allows us to compare sets composed of different numbers of trajectories, and provides a means for interpreting the differences between biological conditions. Finally, we show the robustness of the approach to the intrinsic variability of biological observations, and demonstrate that the statistical differences do not stem from a single experiment, nor from an outlier composed of a minority of trajectories.

We demonstrate our methodology by studying the dynamics of α-synuclein inside and outside of synapses. α-synuclein is a small, soluble, and highly mobile protein (140 amino acid residues) that is strongly accumulated in presynaptic boutons (reviewed in [3]). Experiments based on fluorescence recovery after photobleaching [4] have shown the existence of at least two main modes of diffusion, one in which α-synuclein is transiently bound to synaptic vesicles in the synaptic bouton, and another in which the protein diffuses freely both in axons and in synaptic regions. The existence of an immobile population of α-synuclein molecules, taking the form of protein aggregates at synapses, has also been proposed [4]. In response to strong depolarising signals the bound population of α-synuclein dissociates from its synaptic binding sites and disperses in the neighbouring axon [5]. In agreement with these earlier studies, we found that α-synuclein dynamics differ between synapses and axons. Furthermore, depolarisation of the neurons shifts the relative frequency of the proteins from a less mobile to a highly mobile state, but it does not appear to induce qualitative changes in the type of diffusion dynamics the molecules follow. It is not clear what role this dynamic shift of α-synuclein plays in vesicle cycling and in the regulation of synaptic transmission. Single molecule based imaging in living neurons can help to address this question and yield new information about the physiological function of α-synuclein at synapses, as well as its involvement in pathological processes.

V.2 Materials and methods

V.2.1 Recording αSyn:Eos4 dynamics

a) Neuron cultures and αSyn:Eos4 expression

Primary murine cortical neuron cultures were prepared at embryonic day E17 as described previously [6]. Cortices were dissected, the tissue was dissociated and the cells were seeded at a concentration of 5 x 10^4 cm^{-2} on glass coverslips that had been coated with poly-D,L-ornithine. Neurons were kept at 37°C and 5% CO_2 in neurobasal medium supplemented with Glutamax, antibiotics and B27 (all from Gibco, Thermo Fisher Scientific), infected at day *in vitro* (DIV) 11-24 with lentivirus driving the expression of α-synuclein tagged at its C-terminus with the photoconvertible fluorescent protein mEos4b (αSyn:Eos4) under the control of a ubiquitin promotor, and used for experiments 7 days later. All cell culture and imaging experiments were conducted at the Laboratory for cellular synapse biology at IBENS (Paris). Procedures involving animals were performed according to the guidelines set out by the local veterinary and administrative authorities.

b) Single molecule localisation microscopy (SMLM)

Living neurons expressing αSyn:Eos4 were imaged in modified Tyrode's solution (in [mM]: 120 NaCl, 2.5 KCl, 2 $CaCl_2$, 2 $MgCl_2$, 25 glucose, 5 pyruvate, 25 HEPES, adjusted to pH 7.4) at room temperature, using an inverted Nikon Eclipse Ti microscope equipped with a 100x oil objective (NA 1.49), an Andor iXon EMCCD camera (16 bit, image pixel size 160 nm), and Nikon NIS acquisition software. First, an image of the chosen field of view (average of 10 image frames taken with 100 ms exposure time) was taken in the green channel (non-converted mEos4b fluorescence) using a mercury lamp and specific excitation (485/20 nm) and emission filters (525/30 nm). This was followed by a streamed acquisition of 25 000 movie frames recorded with 15 ms exposure and Δt = 15.4 ms time lapse (total duration: 6 min 25 s) in the red channel using a 561 nm laser at a nominal power of 150 mW for excitation (inclined illumination), together with pulsed activation lasers applied during the off time of the camera (405 nm, approx. 1-5 mW; 488 nm, 10 mW; 0.45 ms pulse). The red emission of the photo-converted mEos4b fluorophores was detected with a 607/36 nm filter (Fig. V.1A).

After recording of the baseline dynamics of αSyn:Eos4, the buffer composition was changed with the addition of Tyrode's solution containing elevated KCl at the expense of NaCl (final concentrations in [mM]: 78 NaCl, 44.5 KCl, 2 $CaCl_2$, 2 $MgCl_2$, 25 glucose, 5 pyruvate, 25 HEPES, pH 7.4). This treatment causes the depolarisation of the neurons leading to the dissociation of α-synuclein from its binding sites in the synaptic bouton [5]. A reference image was taken in the green channel, followed by a second SMLM recording (Fig. V.1B), starting approximately 7.5 min after the first acquisition. Finally, the neurons were fixed with the addition of phosphate buffer at pH 7.4 containing 4% paraformaldehyde and 1% sucrose (final concentration 2% PFA), and a third reference image (green) and SMLM movie (red channel) were acquired in the presence of the fixative (Fig. V.1C).

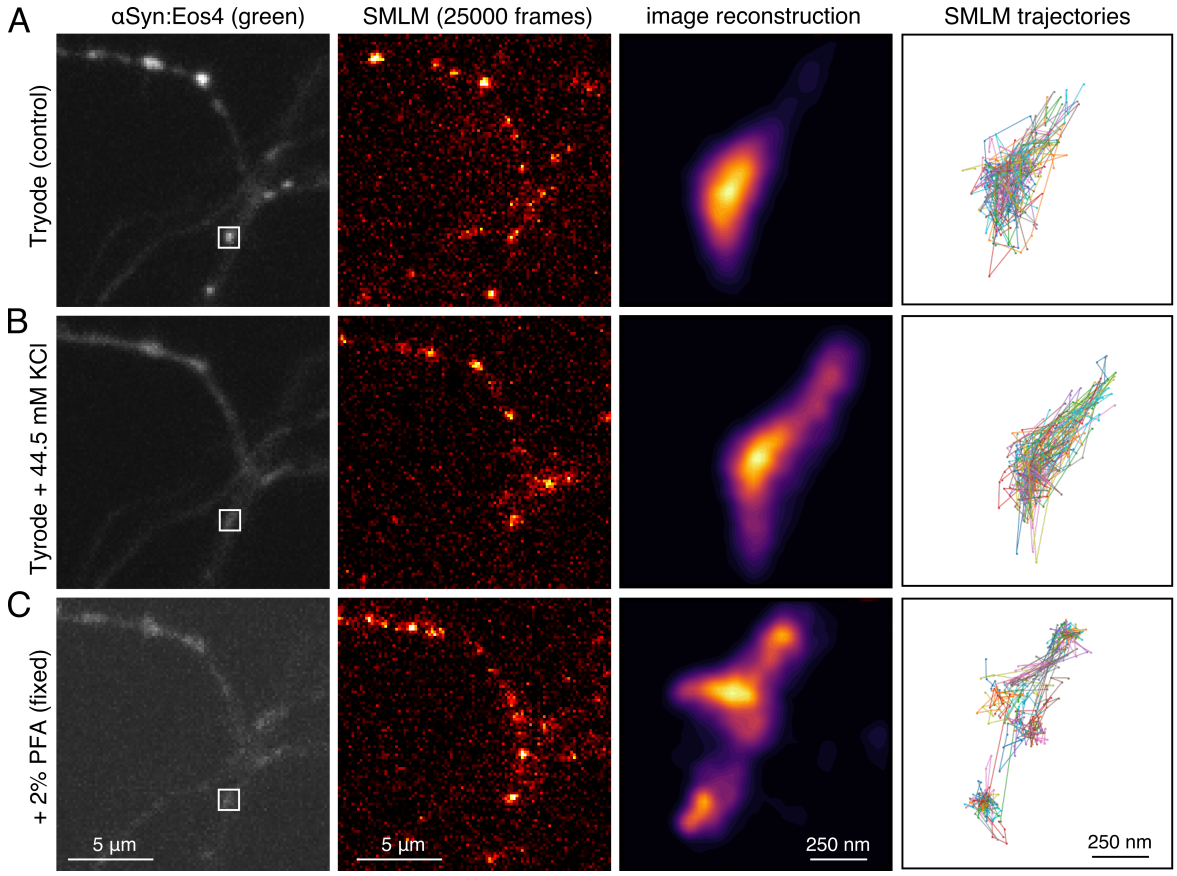

Figure V.1: **Single molecule localisation microscopy (SMLM) of α-synuclein in cortical neurons.** (**A**) Neurons expressing αSyn:Eos4 were first imaged in control condition. A reference image (left panel) was taken in the green channel, followed by a SMLM movie of 25 000 frames in the red channel (panel two). (**B**) A second recording of the same field of view (image and movie) were then acquired in the presence of elevated KCl concentration. Note the dispersal of αSyn:Eos4 in response to depolarisation compared to the control (left panels). (**C**) A third image and movie were acquired after addition of 2% paraformaldehyde. The third column of images shows zoomed SMLM reconstructions of the synaptic bouton indicated in the first image. The fourth column depicts a subset of trajectories from the same synaptic terminal.

c) Image processing and analysis

SMLM image stacks (tiff files) were pre-processed in order to remove background fluorescence using a quantile filter computed on a sliding window. Then, localisations were detected using the algorithm described in [7], based on a wavelet analysis. Subpixel localisation was performed using the radial symmetry center algorithm introduced in [8]. Sample drift was corrected by subtracting the displacement that yielded the best correlation between densities of successive temporal slices grouping 10 000 localisations each. To isolate axons, we applied a Sato filter [9] with a width of 3 pixels on the logarithm of the pixel-wise mean intensity. Then, we used a local thresholding algorithm, provided by [10] to compute a mask over the image. All steps of the analysis were implemented in Python, the code is available at `http://gitlab.pasteur.fr`. Synapses were manually detoured using an ad-hoc graphic user interface. In total, our analysis includes 321 synapses for which more than 150 trajectories were recorded, coming from 10 different fields of view. A synapse is counted twice if it appears in two or three recordings based on the same field of view but done in different conditions (e.g. some synapses appear in control, KCl and fixed conditions).

In the analysis of experimental trajectories, we considered only trajectories located in the axons, and we split them into two groups: those located outside the synaptic region and those located inside. Synaptic regions were delimited by a density threshold of one tenth of the maximum density of detections in the synapse. The density was estimated using a Gaussian kernel method with a bandwidth of 150 nm. We estimated the apparent effective diffusivity [11] of a trajectory from the sample variance of its single-time-lapse displacements, i.e. $\hat{D} = \sum_{i=1}^{N} ||\Delta\mathbf{r}_i - \boldsymbol{\mu}||^2)/[4(N-1)\Delta t]$, where $\Delta\mathbf{r}_i = \mathbf{r}_i - \mathbf{r}_{i-1}$ is the displacement between the $(i-1)$th and ith recorded positions and $\boldsymbol{\mu} = \sum_{i=1}^{N} \Delta \mathrm{r}_i/N$ is the average displacement.

V.2.2 Describing trajectories with latent vectors

The first step of the analysis is to build a latent representation of random walks that does not require strong assumptions about the underlying generative models. In this section, we present the simulation-based inference scheme, the architecture of the neural network used to compute latent vectors from trajectories, and the visualisation of these vectors.

a) Simulation-based inference

In order to ensure that our characterisation of trajectories is accurate, robust and length-independent, it should be trained on an as wide as possible variety of random walks. Hence, we chose to rely on a simulation-based inference procedure [12]. It consists in generating data on which the neural network is subsequently trained. In our case, this amounts to simulating trajectories of a variety of models known to encapsulate different properties of biomolecule dynamics in cells. The physical parameters chosen to simulate these trajectories should at least cover the range of the experimentally observed ones, in order to ensure that the network is able to encode relevant information about the recorded trajectories on which the inference will eventually be performed after its training.

To ensure the diversity of the training set, we simulated trajectories using five different canonical random walk models covering a wide spectrum of possible random walk characteristics:

- the Levy walk (LW), which has non-Gaussian increments and exhibits weak ergodicity breaking;
- scaled Brownian motion (sBM), which is Gaussian, non-stationary and weakly non-ergodic;
- the Ornstein Uhlenbeck process (OU), a Gaussian, stationary process with exponentially decaying autocorrelations;
- fractional Brownian motion (fBM), which is Gaussian, stationary and exhibits slowly decaying temporal correlations;
- and the continuous time random walk (CTRW), which is non-Gaussian, shows weak ergodicity breaking, ageing, and has discontinuous paths.

The models' parameters were drawn from the same distributions throughout the entire study and were chosen to cover the entire ranges observed experimentally. Trajectory lengths were drawn from a log-uniform distribution between 7 and 25 points, which corresponds to a mean length of 14 points. The effective diffusivity was drawn from a log-normal distribution with $D_0 = 1\mu m^2/s$, $\langle \log_{10}(D/D_0) \rangle = -0.5$ and $\mathrm{Var}(\log_{10}(D/D_0)) = 0.5^2$. For the OU model, the relaxation rate θ was drawn from a uniform distribution

We added uncorrelated localisation noise to each point of the trajectories, drawn from a centred Gaussian distribution with standard deviation drawn uniformly between 15 and 40 nm (to include the signal intensity dependence of the localisation precision). The time lapse between recordings was set equal to that of the camera (15,4 ms).

The neural network (the architecture of which is detailed below) was then trained to infer two characteristics of interest from the trajectories: their anomalous diffusion exponent (if applicable), and the random walk model from which they were generated among the five described above. Throughout the training, the network processed $\sim 10^6$ independent simulated trajectories.

b) Graph neural network and random walks

Here, we explain how we construct an informative size-independent representation of random walks. We feed trajectories into a neural network, which we train to compute a vector of summary statistics from each trajectory exactly like in III. We have shown earlier that a such statistics vector contains information relevant to the characterisation of random walks. Although observed trajectories are not all of the same length, the summary statistics vector is of constant size. This is particularly relevant here, as it allows it to be used to compare trajectories of different sizes in subsequent steps of the analysis.

c) Latent representation of trajectories

Once processed by the encoder, each trajectory is reduced to a 16-dimensional vector. For visualisation purposes, we projected this vector on a 2D plane using a parametric-UMAP to perform the projection [13]. This variation of UMAP allows us to learn the transformation projecting the data from 16 to 2 dimensions solely on simulated trajectories, so that it is independent of the experimental trajectories and is only trained once. The GNN was trained

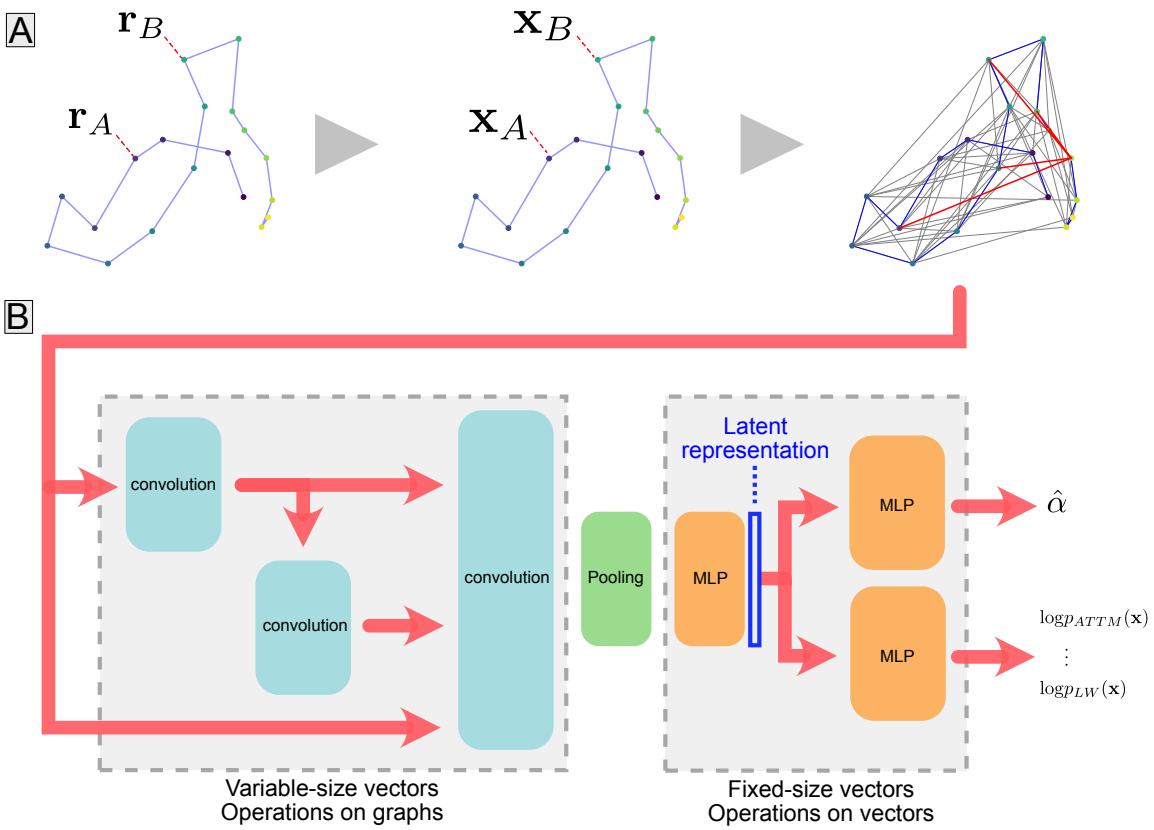

Figure V.2: **Model architecture.** (**A**) Building a trajectory graph. Node and edge features are computed, and edges are drawn between edges following a pre-determined pattern. (**B**) Graph neural network. The graph is passed through a series of graph convolution layers (shown in blue), which propagate information along edges. The pooling operation (green) combines all node feature vectors from a graph into a vector of fixed size representing the graph. This vector is then passed to a multi-layer perceptron (MLP in orange), whose output we refer to as the "latent representation" of the trajectory. The latent representation is fed to two task-specific MLPs: one that predicts the trajectory's anomalous exponent α and one that assigns a vector of probabilities for the trajectory to have been generated by each of the models considered.

first, then the parametric-UMAP projection was learnt and both their sets of weights were frozen.

We designate as “2D latent representation" the two-dimensional vector, output by the parametric-UMAP, which represents a trajectory in the latent space. Figure V.3A shows that latent representations of simulated and experimental data largely overlap, and that the experimental trajectories fall within the region covered by the simulated ones. Figures V.3B and V.3C show that the random walk model and the diffusivity are prominent determinants of the latent space structure. Figures V.3D, V.3E and V.3F illustrate the diversity of αSyn:Eos4 trajectories that can be found in a presynaptic bouton, and how this diversity is captured by the latent representation. Figure V.3F highlights the fact that there is a high variability of trajectory dynamics even within a given synapse, suggesting that α-synuclein molecules can transition between various dynamic modes.

Using the approach described above, we can associate to any set of trajectories, a set of constant-sized vectors characterising their dynamics. Each microscope recording, or organelle within it, can thus be characterised by a set of N 2-dimensional feature vectors, N being the number of trajectories. Note that we could have used the latent of vectors of 16 dimensions, but in this application the 2D projection captured enough information about the random walk dynamics. The conditions are thus met to perform statistical testing.

V.2.3 Statistical testing in the space of latent representations

We develop in this section the statistical test we use to compare the dynamics of single molecules in different organelles and under different biological conditions. Each organelle is characterised by the set of its latent vectors. Thus, we base our statistical test on the comparison of the generating distributions of these vectors. In the absence of *a priori* knowledge of these distributions, we employ a kernel-based approach: the maximum mean discrepancy (MMD) test.

a) Maximum mean discrepancy

Maximum mean discrepancy (MMD), introduced in [2], is a measure of distance between distributions. It was developed to perform statistical testing between two sets of independent observations lying in a metric space $\mathcal{X}$, $X = \{x_1, \dots, x_m\}$ drawn from probability measure p and $Y = \{y_1, \dots, y_n\}$ drawn from q, with the the goal of assessing whether or not p and q are different.

Given a class $\mathcal{F}$ of functions from $\mathcal{X}$ to $\mathbb{R}$, the MMD between two probability measures p and q is defined as

$$\mathrm{MMD}[\mathcal{F}, p, g] = \sup_{f \in \mathcal{F}} \left(\mathrm{E}_x \left[f(x) \right] - \mathrm{E}_y \left[f(y) \right] \right),$$

where E_x and E_y denote expectation w.r.t. p and q, respectively.

If the function class is the unit ball in a Reproducing Kernel Hilbert Space (RKHS) [14] $\mathcal{H}$, the square of the MMD can directly be estimated from data samples. Denoting k the kernel operator such that $\forall f \in \mathcal{F}, f(x) = \langle f, k(x, \cdot) \rangle$, an unbiased estimator of the square of

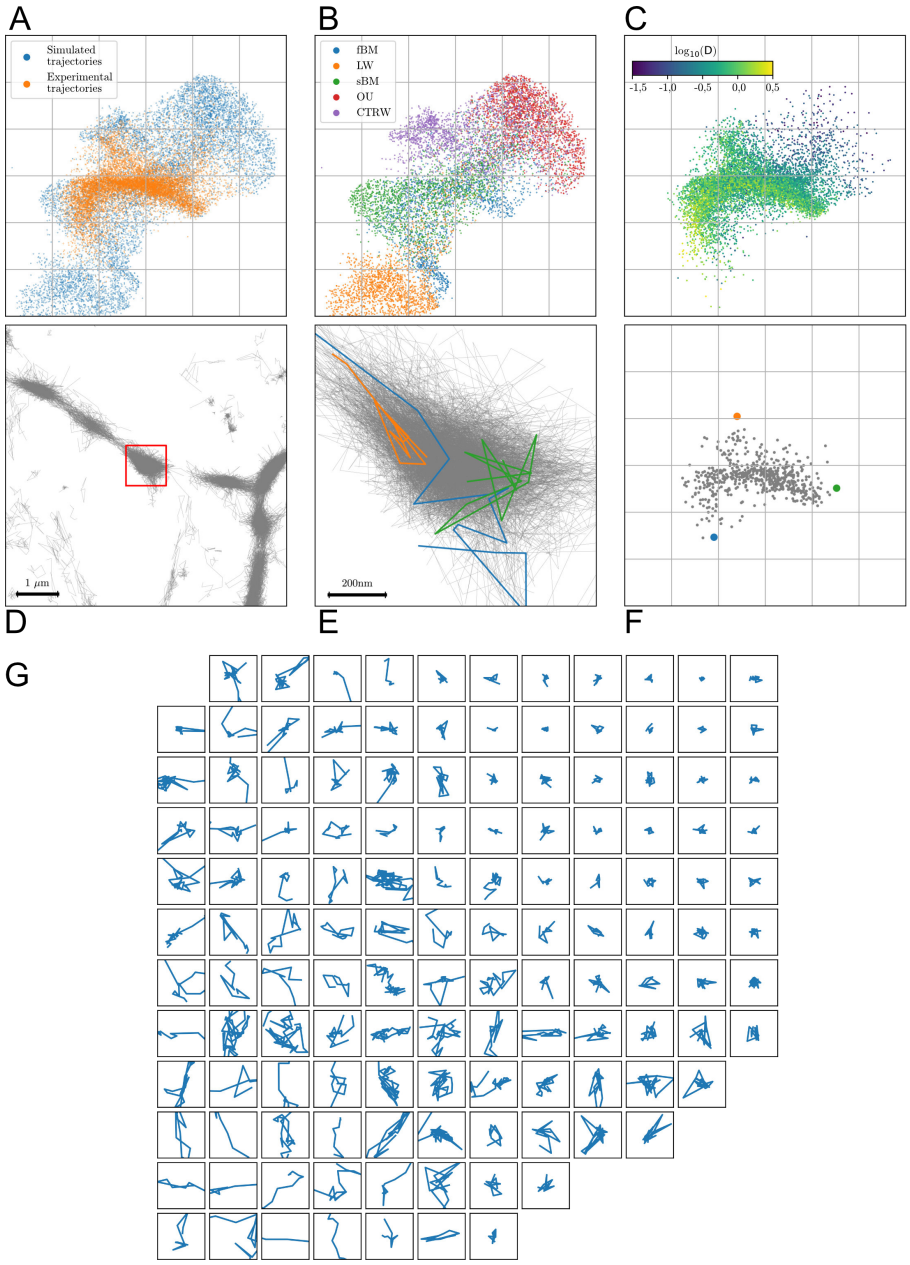

Figure V.3: **2D Latent space of trajectories.** (**A**) Latent representations of simulated versus experimentally recorded trajectories. (**B**) Latent representations of simulated trajectories, coloured by random walk model (fBM: fractional Brownian motion; LW: Levy walk; sBM: scaled Brownian motion; OU: Ornstein-Uhlenbeck process; CTRW: continuous-time random walk). The cropped region at the bottom contains mostly simulated Levy walks. (**C**) Latent representation of experimentally recorded trajectories, coloured according to the estimated log-diffusivity. (**D**) Recorded trajectories at synapses and in the axon. (**E**) Zoom on a presynaptic bouton, delimited by the red square in panel D. Three individual trajectories are highlighted. (**F**) Latent representation of the trajectories at this synapse, with colored dots corresponding to the three trajectories highlighted in E. (**G**) Examples of acquired trajectories, located according to their position in the 2D latent space. Each square has a side length of 1 micrometer.

the MMD between X and Y is given by:

$$\mathrm{MMD}_u^2[\mathcal{F}, X, Y] = \frac{1}{m(m-1)} \sum_{\substack{i,j \\ i \neq j}} k(x_i, x_j) + \frac{1}{n(n-1)} \sum_{\substack{i,j \\ i \neq j}} k(y_i, y_j) - \frac{2}{nm} \sum_{i,j} k(x_i, y_j)$$

In our case, $\mathcal{X} = \mathbb{R}^2$, and we used the classical Gaussian kernel $k : x, y \to k(x, y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{||x-y||_2^2}{2\sigma^2}\right)$. We set the kernel bandwidth σ either to the median of the pairwise Euclidian distances between samples from X and Y or we optimised it in specific conditions.

The MMD is capable of detecting subtle differences such as the ones between data generated by generative adversarial networks (GANs) and real data [15]. It has also proved efficient in discovering which variables exhibit the greatest difference between datasets [14, 2].

b) Statistical test

We adapted the bootstrap test described in [2] to assess whether dynamics of α-synuclein observed in two experimental conditions exhibit different properties. We denote by X and Y the two sets of trajectories observed in the two different conditions, drawn from unknown probability densities p and q. For simplicity, we assume that X and Y have the same number of elements, $m = n$, using the same notation as in the last section. In practice, the number of observed trajectories varies significantly across experimental replicates. To ensure that all replicates have an equal importance when the two sets do not have the same number of trajectories we randomly sub-sampled the larger of the two sets to equalise their sizes. The null hypothesis H_0 of the statistical test is that $p = q$, i.e. the two conditions lead to the same distribution of random walks. Under H_0, we approximated the distribution of $\mathrm{MMD}_u^2[\mathcal{F}, X, Y]$ by bootstrapping, i.e. we drew random samples from the union of X and Y and distributed them in two groups X' and Y' (whose sizes respectively match those of X and Y), on which we computed $\mathrm{MMD}_u^2[\mathcal{F}, X', Y']$. We repeated this procedure a sufficient number of times to obtain an estimation of the distribution of MMD_u^2 under the assumption that X' and Y' are drawn from the same distribution. Then, if the original $\mathrm{MMD}_u^2[\mathcal{F}, X, Y]$ was greater than the $1 - \alpha$ quantile of this distribution, we rejected H_0. This test is said to be of level α, because with probability α, we will reject the null hypothesis when it is actually true.

V.3 Results

V.3.1 Detecting differences between sets of simulated trajectories

To assess the performance of the full statistical testing framework, we applied it on simulated data. We set the level of statistical significance to $\alpha = 0.05$, and we simulated trajectories as described in Material and Methods.

A first case of our test is to detect changes in the proportions of given types of trajectories between two sets of observations. This is illustrated in Fig. V.4, where we show example comparisons in the 2D latent space between two sets with different proportions of their trajectories generated by fBM and sBM. We compared fBM and sBM, since they share numerous features and because for a large range of values of the anomalous exponent they are challenging to distinguish [16]. Furthermore, these two random walk models are highly

representative of our experimental data, as can be seen by their latent space occupations (compare Figs. V.3A and B).

The difficulty of separating the two populations depends on their relative proportions in the two datasets, and we see that both the amplitude of the witness function (Fig. V.4C) as well as the value of the test statistic (Fig. V.4D) decrease as the ratio is closer to 1:1. When the two sets are drawn from the same 50/50 distribution, the test does not, and should not, find significant differences between them.

The other main factor determining the difficulty of detecting a difference, is the size of the datasets. Experimentally, changes in biological conditions lead not only to changes in the properties of the random walks. It also leads to changes in the total number of trajectories of a given type. This causes challenges in performing proper statistical testing. To quantitatively assess the effect of both the number of trajectories and the relative proportions belong to different random walk classes, we conducted numerical experiments where we varied these two parameters systematically (Fig. V.10A).

Besides differences in the proportions of trajectories generated by different random walk models, the sets may also differ in the models' parameter values. We thus additionally evaluated the test's ability to distinguish two sets of fBMs, one with anomalous diffusion exponent $\alpha = 1 - \delta$ and the other with $\alpha = 1 + \delta$. Our results indicate that in both cases 1 000 trajectories are sufficient to detect subtle changes between distributions (Fig. V.10). Fewer trajectories are needed to detect starker differences. In cases where the compared sets are drawn from the same distribution ($\nu = 0$ and $\delta = 0$), the null hypothesis is rejected in about 5% of cases, consistent with our chosen α-level.

One way to further improve these results is to optimise the kernel used to compute the MMD. We show in Fig. V.9 how kernel bandwidth and shape affect the power of the test. If the kernel bandwidth is too small, this weakens the test by making it too sensitive to noise. Conversely, if its bandwidth is too large, this prevents the test from detecting subtle changes. We tested the effect of the kernel characteristics in the same setting as illustrated in Fig. V.4 and supplementary Fig. V.10A, with $N = 200$ trajectories in each set and comparing sets with 70% fBM / 30% sBM and 30% fBM / 70% sBM. We observed that a Gaussian kernel with radius σ equal to the median pairwise distance in the dataset (i.e. σ between 1.5 and 2) yields a near-optimal test, in agreement with earlier findings [2]. Finally, while we have here focused on optimizing type II error while controlling type I error (i.e. fixed α-level), the parameters and the functional form of the kernel can be adjusted to control either type I or II error while optimising the other [17].

V.3.2 Differences of α-synuclein mobility in axons and at synapses in response to membrane depolarisation

We analysed trajectories of αSyn:Eos4 molecules in the axons of cultured cortical neurons and compared them between different subcellular regions, outside or inside the synaptic bouton, and experimental conditions, control, high KCl (leading to synaptic depolarisation) and fixed cells.

The three experimental conditions (control, KCl, fixed) and two subcellular regions (extra- and intra-synaptic) define six populations of trajectories, whose latent space occupation densities are shown in Figs. V.5A and V.5B. In the remaining four columns of Fig. V.5, we illustrate a few comparisons performed between pairs of trajectory populations

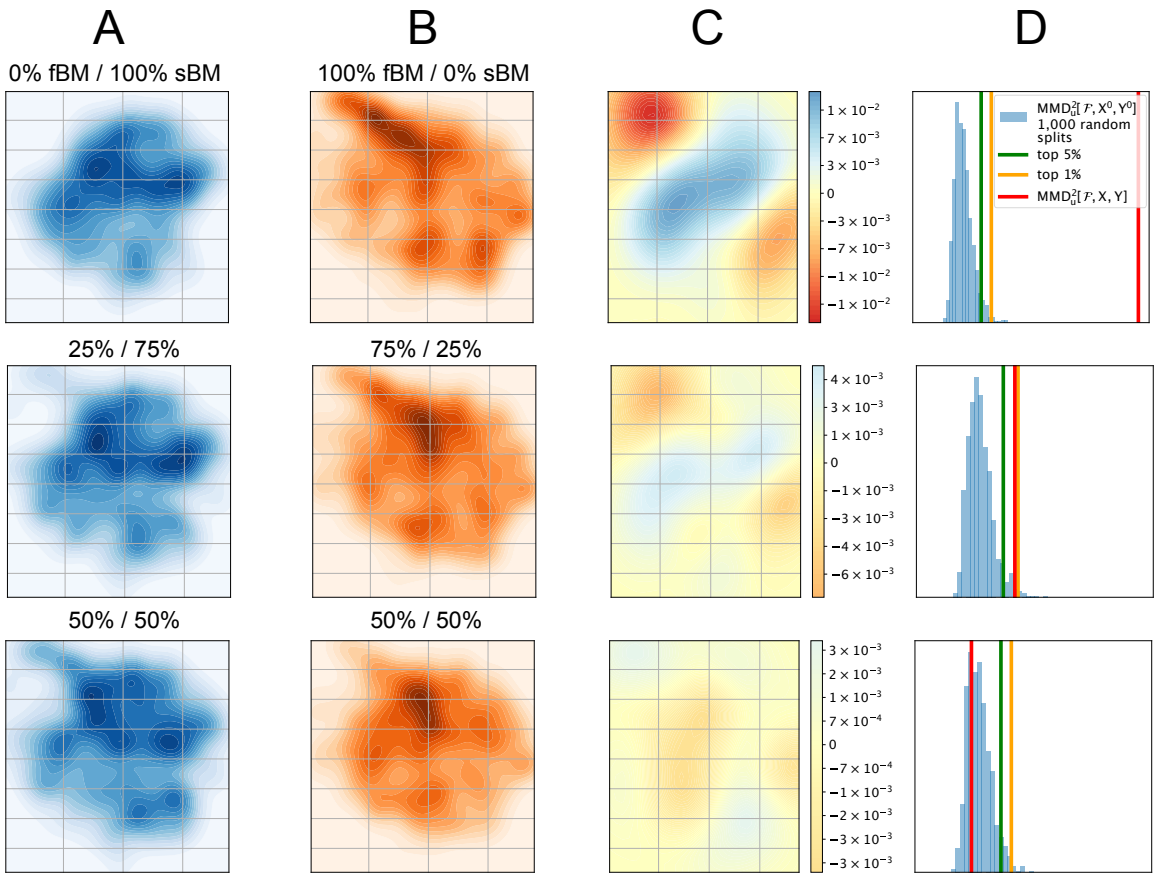

Figure V.4: **MMD-based statistical test.** (**A**), (**B**) Densities of latent vectors in the 2D plane, for two sets of 500 trajectories with different ratios of fBM / sBM trajectories (top: 0% fBM / 100% sBM vs. 100% fBM / 0% sBM, middle: 25% / 75% vs. 75% / 25%, bottom: 50% / 50%). (**C**) Witness functions of the MMD test for difference between A and B, i.e. the function attaining the maximum in Eq. a), based on the available samples. (**D**) Distribution of the test statistic MMD_u^2 between sets of equal size composed of randomly chosen trajectories of the two sets, with its top 1% (yellow line) and top 5% percentiles (green), as well as the unbiased estimate of the square MMD between the two sets (red).

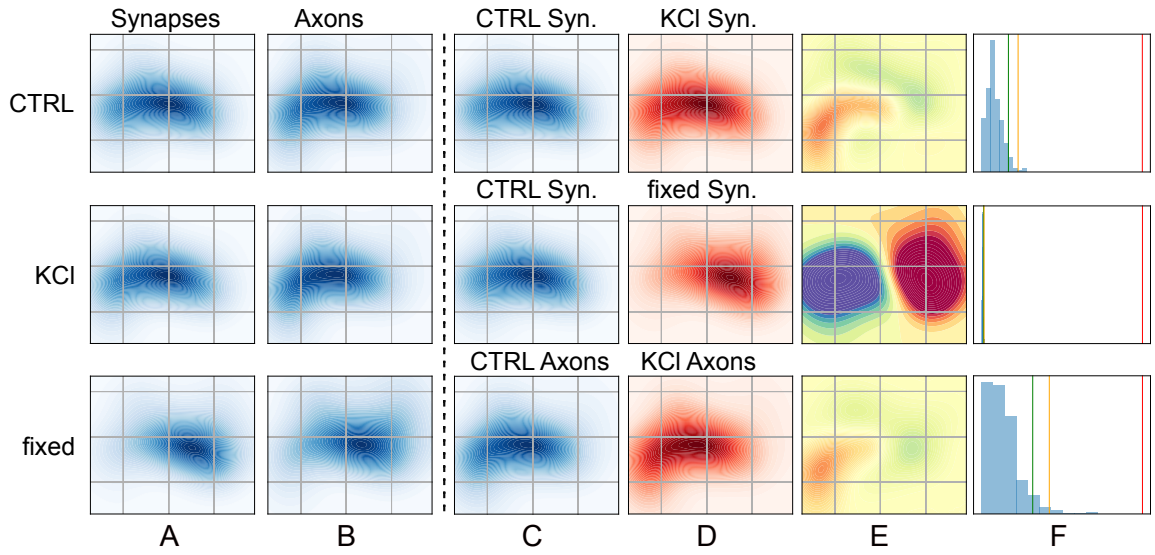

Figure V.5: **Latent space occupation & statistical testing.** Left part: (**A**), (**B**) Latent space occupation densities of αSyn:Eos4 trajectories observed in synapses and axons, in the three experimental conditions (control, high KCl, fixed). Right part: each row is one comparison of two sets of trajectories. (**C**), (**D**) Side by side comparison of the latent space occupation densities of the two sets of trajectories used in the comparison in E and F. (**E**) Witness function of the comparison. The colour scale is preserved across rows. (**F**) Histograms illustrating the statistical test based on the MMD. The green and yellow vertical lines represent the top 5% and top 1% quantiles of the null distribution of the squared MMD, respectively, while the red line shows the squared MMD for the experimental data.

using our statistical test. Figures V.5C and V.5D show the latent space densities for the two conditions that are compared in each case, the differences of which are the witness functions shown in Fig. V.5E. The distributions of the test statistics under the null hypothesis, obtained after 1 000 bootstrapping iterations, are shown in Fig. V.5F, as well as its top 1% and 5% quantiles and the test statistic obtained on the actual two compared populations.

In all of the three cases illustrated in Fig. V.5, the empirical MMD_u^2 is significantly higher than the top 1% quantile of the null distribution, meaning that our test detects a significant difference in the properties of the trajectories of the two compared subsets at the 1% α-level. Note the wide range of magnitudes spanned by these differences, which is well judged by looking at the absolute intensity of the witness functions shown in Fig. V.5E. We see that the difference induced by fixation on α-synuclein mobility at synapses (Fig. V.5E, middle) is much more pronounced than the one induced by high KCl treatment (Fig. V.5E, top). This is also apparent when looking at how large MMD_u^2 is compared to its distribution under the null hypothesis: in the case of the fixed vs control comparison (Fig. V.5F, middle), the histogram of the MMD_u^2 values obtained under the null hypothesis is completely squeezed because the empirical MMD_u^2 is more than two orders of magnitude larger than the top 1% quantile of the distribution under the null hypothesis. In comparison, KCl treatment produces a less drastic change in α-synuclein trajectories located in synaptic terminals (Fig. V.5F, top), although this effect is also highly statistically significant ($p \ll 0.01$).

We further observed that, while their magnitudes differ, the witness functions of the control/KCl comparisons in axons and in synaptic boutons (Fig. V.5E, top and bottom) exhibit similar patterns. This indicates that the addition of KCl to the medium can affect the physical properties of many if not all α-synuclein molecules in a similar manner, irrespective of their subcellular location. In contrast, α-synuclein mobility in fixed neurons appears to be almost entirely abolished, which is seen not only in the amplitude of the change, but also in the fact that the occupation of the latent space displays massive qualitative differences in this condition. This demonstrates that α-synuclein is highly mobile in living cells, and helps to put our experimental findings into perspective.

Another feature of the MMD, is the possibility of extracting the points of the feature space that are most important for distinguishing one distribution from another. By finding the local maxima of the S statistic, introduced in [18], which is given as the ratio of the mean squared amplitude and the variance of the witness function estimated by bootstrapping, we identify the regions of the latent space where the occupation differs most in the two populations. This enables a straightforward interpretation of the latent space. In Fig. V.6, we apply this method to the comparison of intra-synaptic αSyn:Eos4 trajectories in the control and KCl conditions. According to this analysis, the representative α-synuclein trajectories exhibit a greater mobility in the depolarised state. This is likely the result of a weaker binding of α-synuclein at synapses, as reflected in the overall reduction of α-synuclein molecules during KCl application (Figs. V.1A and V.1B).

Furthermore, as illustrated in Supplementary Fig. V.11, we use this statistic to check that all acquired fields of view contribute evenly to the difference between the control and KCl conditions. We looked at the proportion of trajectories coming from each field of view and condition in a region of the latent space which we define as "critical", based on the value of S (it is the contiguous domain containing the maximum of S and where S is greater than half of its maximum value). We could thus confirm that, on the one hand, trajectories located in this region of the latent space originate from all considered fields of view in a balanced manner, and on the other hand, that within each field of view, the difference of representation of each condition is in the same direction (except for one field of view, where there is almost no difference). This furthermore excludes the possibility that the observed differences are solely due to a single abnormal recording.

V.3.3 Comparing synapses

The approach we propose is not restricted to inter-condition comparisons, but can be used to compare any two subsets of trajectories. Hence, we can group trajectories by synapse and external condition (control, KCl or fixed), and compute the value of MMD_u^2 of all the pairs of so-obtained synapses. This provides us with an inter-synapse distance matrix, shown in Fig. V.7A. Using these distances, we can embed the trajectory subsets in an Euclidian space, *i.e.* summarise each subset by a vector of fixed dimension, using for instance the multi-dimensional scaling (MDS) algorithm [19]. We adapted the MDS algorithm in order to account for the uncertainty that we have in the estimation of the squared distance, which notably depends on the number of observed trajectories per synapse (see Supplementary information). We show in Fig. V.7B the vectors obtained when using this method to embed synapses in a 5-dimensional space. The clouds of points corresponding to synapses observed in each condition do not perfectly overlap, which is expected given that we previously showed

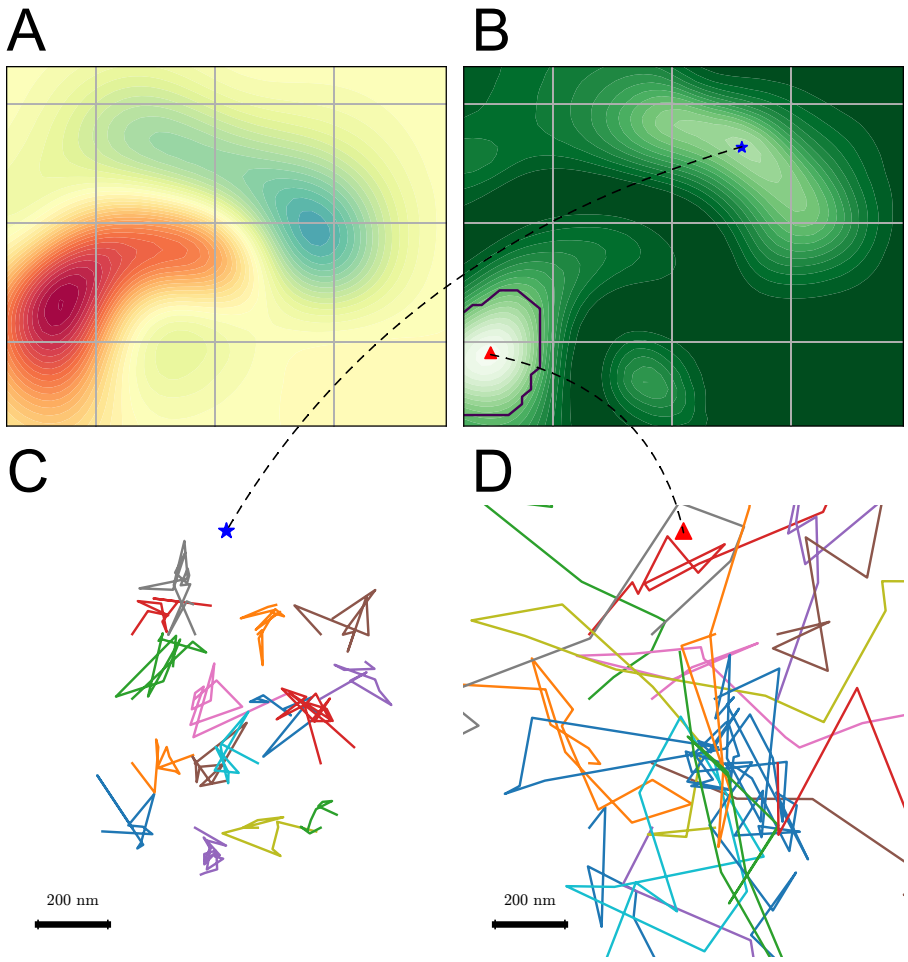

Figure V.6: **Most salient dynamics.** (**A**) Witness function of the comparison of intra-synaptic trajectories, between the control (blue) and KCl (red) conditions. (**B**) Test statistic such as defined in [18], i.e. ratio of the square amplitude divided by the variance. The black contour indicates the "critical region", *i.e.* the region of the latent space most responsible of the statistical difference. (**C**) and (**D**) Illustration, for each maximum of the test statistic, of its 16 closest trajectories.

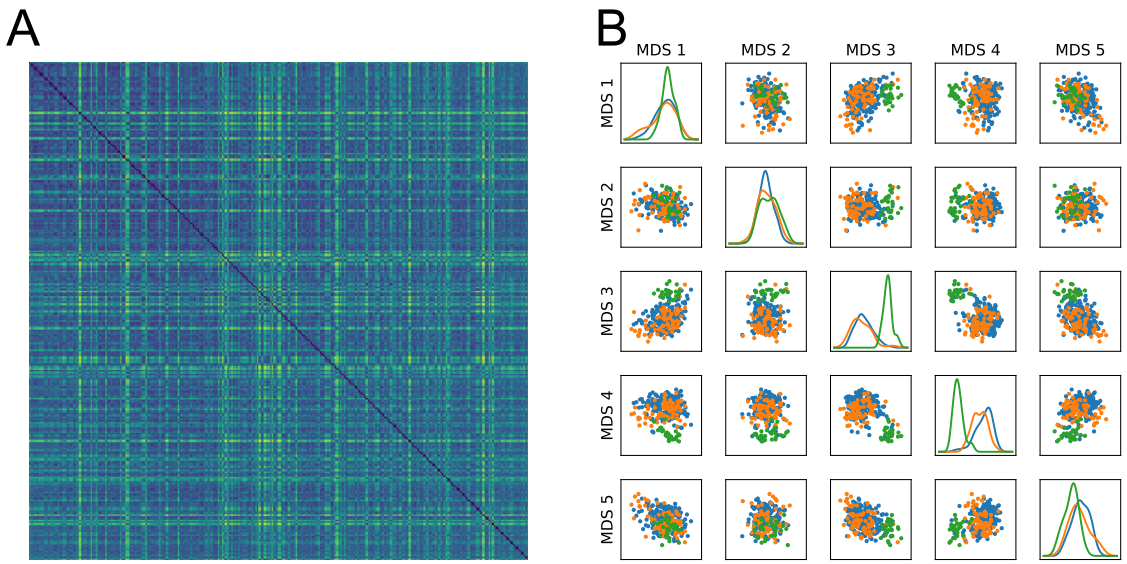

Figure V.7: **Comparing individual synapses.** (**A**) Matrix of the inter-synapse MMD: the value at row i and column j is colored according to the MMD between trajectories of synapse i and synapse j. (**B**) 5-dimensional embedding obtained by multi-dimensional scaling (MDS) based on this distance matrix. Dots are coloured according to the condition in which the synapse they represent was observed: CTRL in orange, KCl in blue and fixed in green.

that conditions were significantly different. However, this view provides an intuitive illustration of the relative extent of inter- and intra-condition variability. More generally, this type of visualisation may be used to illustrate potential continuous shift or clear separations between distinct synaptic regimes observed in different conditions.

V.4 Discussion

We have introduced a statistical procedure to compare organelles or biological conditions of single molecule experiments. This statistical test does not require explicitly defining the generative models of experimentally recorded random walks. The test consists of two steps. In the first step, an amortised inference is used to reduce any trajectory to a features vector of constant size. In the second, the distribution of features between different conditions is compared by the MMD statistical test.

V.4.1 Comparison with Bayesian model averaging

Our approach allows extracting physically and biologically relevant results without having to assign the biomolecule motion to canonical models. Although these models are instrumental for interpreting the properties of random walks, the complexity and heterogeneity of biological environments at the nanometer/micrometer scale often precludes an unambiguous model assignation.

An alternative approach that would also use a set of canonical random walks as a possible

basis for description is Bayesian Model Averaging (BMA). The BMA method does not look for the best model to describe experimental data but rather evaluates the parameters for each of the possible models and then averages the different results. It is challenging to apply this approach to our current problem for two reasons: (i) the space parameters of the different random walks are not identical and (ii) the evaluation of the parameters associated with each random walk by Bayesian methods is not possible for all models. It would be possible to develop Bayesian method variations to estimate the parameters of models that do not have tractable likelihoods. However, the marginalisation of the different models would be computationally intensive. Finally, even though BMA could include information from multiple models, it would still rely on the projection of the experimental date onto a set of canonical models.

V.4.2 Biological interest at the age of large scale experiment

We applied our approach to study the dynamics of α-synuclein molecules in axons and presynaptic boutons. In agreement with earlier studies of the population dynamics of α-synuclein [5, 4], we found that the protein assumes differing dynamic states at synapses and in axons. Depolarisation of the presynaptic terminals through the application of high potassium concentrations shifted the relative frequency of the various states, without necessarily changing the types of diffusion. In other words, our analysis demonstrates clear quantitative changes in the mobility of α-synuclein but does not identify qualitative changes due to the extensive overlap in the occupation of the latent space (with the exception of the fixed state).

This statistical testing procedure paves the way to automated analysis of single molecule experiments. Single molecule pharmacology is an emerging field [20, 21], in which the effects of drugs are evaluated at the nanometer scale by studying the spatial properties and dynamics of biomolecules of interest. The possibility to automatically compare different conditions without relying on manually selected generative models of molecule diffusion would be helpful in defining groups of conditions in which a certain effect can be detected. Even though model identification will often be impossible, the properties of the latent space can reveal the source of observed differences. The witness function can thus be instrumental in differentiating changes in the probability of occupancy of specified domains within the latent space between conditions. As illustrated on Fig. V.6, going from a region of the latent space to an intelligible trait of trajectories is rather intuitive, hence the interest of this method to orient further analysis and build biologically relevant hypotheses.

Beyond the automation of the analysis procedure between biological conditions, our approach is well suited for exploratory data analysis. The capacity to project individual, differently sized trajectories into finite sized vectors makes it possible to study precise subcellular compartments or organelles in a standardised form, and thus allows to test statistical differences between these regions. Hence, recorded single molecule data can be searched in order to detect and characterise regions of the cell that have different statistical properties. This exploration can be done even in regions with different trajectory densities, as is the case for α-synuclein at synapses versus axonal domains.

V.4.3 Limitations

One of the current limitations of the current approach is the difficulty in evaluating the type II error [17] bounds on the statistical test. The MMD test is applied within the latent space of

the GNN. This manifold is built using a set of non-linear operations, which depends both on the numerical trajectories seen during the training and on the cost function being optimised. Hence, there may be domains within the latent space that could lead to improper sensitivity of the statistical test. As can be seen in [16] and in Fig. V.3, different types of random walks occupy domains of different size and there is a large overlap of the regions. Since our approach relies on a simulation based framework, it is possible to use numerical simulations matching the experimental occupancy of the latent space to evaluate the accuracy of the test. Furthermore, extensive simulations and check if the statistical test misbehaves, even though this procedure can be time consuming. In order to further improve the statistical power of the test, one could optimise the kernel with which the MMD is computed. Along these lines, a possible variant of this method could rely on an encoder network trained not on a supervised inference task but rather to maximise the MMD between two sets of experimentally recorded trajectories. This, however, would require i) a substantially larger quantity of experimental data and ii) to re-train the network on each new comparison.

V.5 Accounting for length heterogeneities

Another limitation lies in the dependence to the upstream tracking step, and notably the spurious difference in recovered dynamics stemming from differences in spatial density. Indeed, the tracking method cuts trajectories when two possible options are found within a given search radius. This ambiguous choice is more often encountered in dense areas, and trajectories recorded in these regions thus tend to be shorter. The latent space learnt by the network is sensible to the length of trajectories: long trajectories contain more information and can thus be characterized more accurately, while shorter ones lie in a central region. Hence, the density-induced difference of trajectory length might result in false positives of our dynamics-based statistical test: two sets of trajectories having the same dynamics might be detected as significantly different simply because they have length differences. This phenomenon is illustrated in figure V.8, where it is shown that the inferred anomalous diffusion exponent of trajectories of superdiffusive fBM reconstructed from very dense regions tends to be biased towards 1. This is problematic because length is sampled independently of the dynamics' parameters throughout the simulation phase, which amounts to considering length and dynamics as two unrelated variables – a reasonable hypothesis as long as particles are not fast enough to blur the images to the point that it prevents their detection.

V.6 Supporting information

V.6.1 Multi-dimensional scaling with uncertainty on estimated distances

When estimating $\mathrm{MMD}_u^2[\mathcal{F}, p, q]$ from sets of trajectories $X \sim p$ and $Y \sim q$, the uncertainty directly depends on the number of samples in both X and Y. In our case, the uncertainty, which can be evaluated by bootstrapping, is sometimes of the same order of magnitude than the estimated value. Furthermore, the number of elements per set (here, the number of trajectories per synapse), spans more than an order of magnitude and uncertainty thus greatly varies from one measure to the other. This should be taken into account when using $\mathrm{MMD}_u^2[\mathcal{F}, X, Y]$ to embed subsets of trajectories.

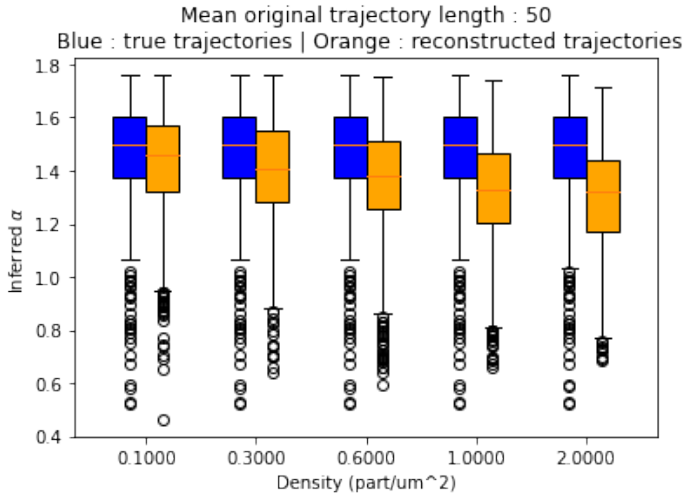

Figure V.8: Influence of the density of particles on the inferred value of α

Hence, starting from a matrix of squared distances $\mathbf{D}^2$ between N sets of trajectories, and a matrix of uncertainties of these squared distances $\mathbf{D}^2_\sigma$, we obtain a set of N Euclidian vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ by maximizing the probability of the resulting squared distances, assuming that they follow Gaussian laws whose means are the coefficients of $\mathbf{D}^2$ and standard deviations coefficients of $\mathbf{D}^2_\sigma$. This amounts to solving the following optimisation problem:

$$\max_{\mathbf{x}_1,\ldots,\mathbf{x}_N} \sum_{i<j} \log\left(P_{i,j}\left(||\mathbf{x}_i - \mathbf{x}_j||^2\right)\right)$$

$$\max_{\mathbf{x}_1,\ldots,\mathbf{x}_N} \sum_{i<j} \left(\frac{||\mathbf{x}_i - \mathbf{x}_j||^2 - \mathbf{D}^2_{i,j}}{\mathbf{D}^2_{\sigma i,j}}\right)^2.$$

We do so using a gradient ascent method, starting from the solution of the classical MDS algorithm.

V.6.2 Graph neural network features and architecture

a) Node and edge features

Prior to entering the encoder, each trajectory is turned into a graph as described in b). To each node is associated a vector containing the following features:

- the normalised time: i/N;
- the cumulative distance covered by the trajectory up to i: $\sum_{k\leq i}\|\Delta\mathbf{r}_k\|_2$;
- the cumulative squared distance covered by the trajectory up to i: $\sum_{k\leq i}\|\Delta\mathbf{r}_k\|_2^2$;
- the maximum step size up to i: $\max_{k\leq i}\|\Delta\mathbf{r}_k\|_2$.

Similarly, each edge is associated to the following set of features:

- the normalised time difference: $(j-i)/N$;

- the distance: $\|\mathbf{r}_j - \mathbf{r}_i\|_2$;
- the dot product of jumps: $\Delta\mathbf{r}_i^\top \Delta\mathbf{r}_j$ (equal to $\Delta\mathbf{r}_i \Delta\mathbf{r}_j$ for 1D trajectories);
- the distance covered by the trajectory between i and j: $\sum_{i<k\leq j}\|\Delta\mathbf{r}_k\|_2 = \sum_{k\leq j}\|\Delta\mathbf{r}_k\|_2 - \sum_{k\leq i}\|\Delta\mathbf{r}_k\|_2$
- sum of square step sizes between i and j: $\sum_{i<k\leq j}\|\Delta\mathbf{r}_k\|_2^2 = \sum_{k\leq j}\|\Delta\mathbf{r}_k\|_2^2 - \sum_{k\leq i}\|\Delta\mathbf{r}_k\|_2^2$.

Features based on distances are computed on a normalised three different versions of the trajectories, corresponding to three different normalisation factors: the total covered distance, the standard deviation of the positions, and the standard deviation of step sizes. These three scales are concatenated to the output of the pooling layer and are thus processed by the perceptron which produces the latent representation of a trajectory (see Fig. V.2).

b) GNN architecture

The architecture of the GNN used in the summary network is similar to the encoder network proposed in [16], with the difference that we here additionally apply edge features. Node and edge features are first passed to perceptrons, which embeds them in a homogeneous space. The network is then composed of three successive convolution layers (one taken from [22] and two edge-conditioned layers taken from [23]) outputting node features matrices $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$, each of 32 dimensions, which are summed to form $\mathbf{x}^{(f)}$. The rows of this matrix of nodes features are then averaged during the pooling step, to keep just one row per graph, i.e., per trajectory. This vector is subsequently passed to a three-layer perceptron, the output of which is the summary statistics vector. All multi-layer perceptrons have a leaky-ReLU activation with slope 0.1 for negative values. We summarise their shapes in table V.1

MLP	**Layers**
edge features embedding	(13,32,32,16)
node features embedding	(10,32,32,10)
edge convolution (x2)	(16,32,32,1024)
final embedding	(38,32,16,16)
alpha predictor	(16,128,64,64,16,1)
model classifier	(16,32,16,5)

Table V.1: Shapes of multi-layer perceptrons used in the network.

Bibliography

[1] **Verdier Hippolyte**, Françcois Laurent, Alhassan Cassé, Christian L Vestergaard, Christian G Specht, and Jean-Baptiste Masson. A maximum mean discrepancy approach reveals subtle changes in α-synuclein dynamics. *bioRxiv*, 2022.

[2] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. 13(25):723–773. URL http://jmlr.org/papers/v13/gretton12a.html.

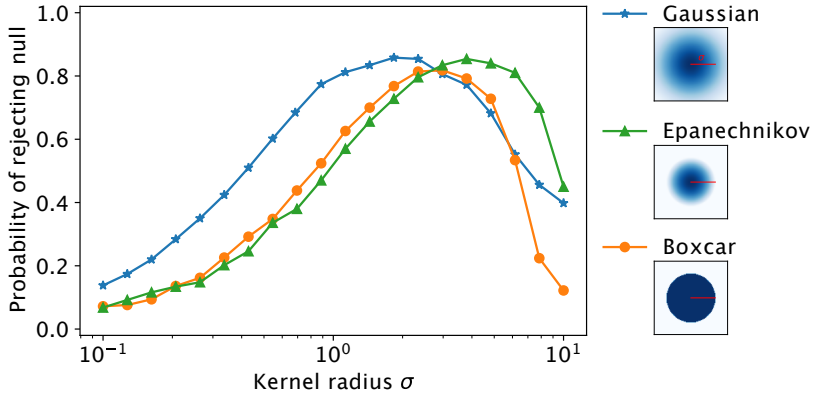

Figure V.9: **Influence of kernel parameters on the test's power.** Probability of rejecting the null hypothesis that the two sets of trajectories are drawn from the same distribution, with varying kernel types and radii. The sets are drawn with the same characteristics as those used for Fig. V.10A with $N = 200$ and $\nu = 0.2$.

[3] Christian G Specht. A quantitative perspective of alpha-synuclein dynamics–why numbers matter. *Frontiers in Synaptic Neuroscience*, 13, 2021.

[4] Kateri J Spinelli, Jonathan K Taylor, Valerie R Osterberg, Madeline J Churchill, Eden Pollock, Cynthia Moore, Charles K Meshul, and Vivek K Unni. Presynaptic alpha-synuclein aggregation in a mouse model of parkinson's disease. *Journal of Neuroscience*, 34(6):2037–2050, 2014.

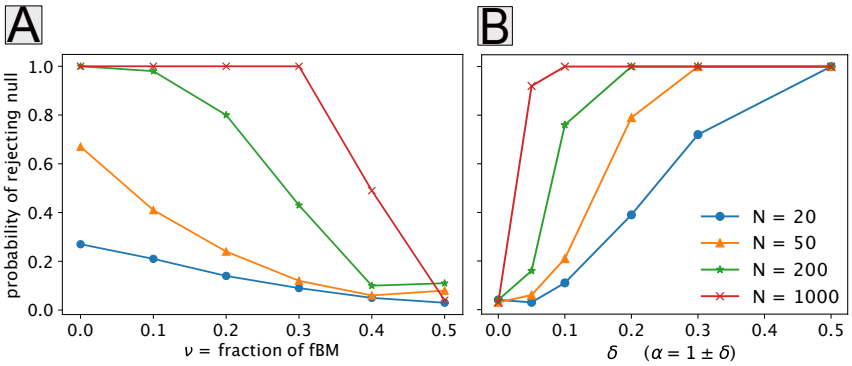

Figure V.10: **Performance of the statistical test on simulated trajectories. A**: Probability of detecting a difference between two sets of N trajectories composed of a fraction ν of fractional Brownian motions and $1 - \nu$ of scaled Brownian motions. **B**: Probability of detecting a difference between two sets of N fractional Brownian motions, one with anomalous diffusion exponent $\alpha = 1 - \delta$ and the other with $\alpha = 1 + \delta$. Probabilities are estimated by performing the test 100 times; at each trial, new trajectories are simulated and bootstrap-estimation of the distribution of MMD_u^2 under the null hypothesis is done using 100 random splits.

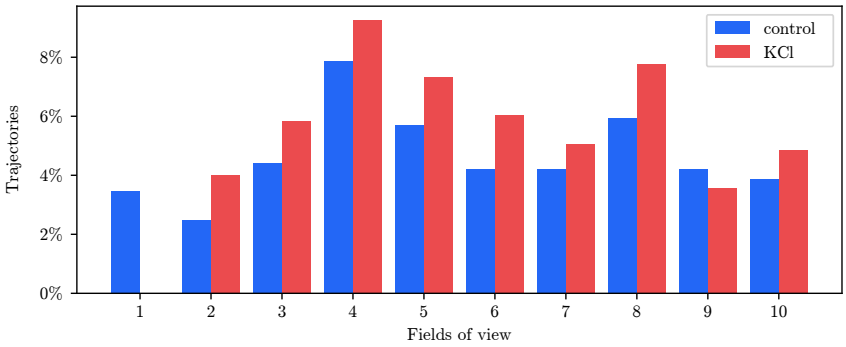

Figure V.11: **Origin of trajectories found in the critical region.** These counts were obtained using a set composed of the same number $n = 1,000$ of trajectories from each microscopy recording. We randomly subsampled those who had more intra-synaptic trajectories, and discarded those who had less than 1,000 (hence the column with missing number of KCl trajectories).

[5] Doris L Fortin, Venu M Nemani, Susan M Voglmaier, Malcolm D Anthony, Timothy A Ryan, and Robert H Edwards. Neural activity controls the synaptic accumulation of α-synuclein. *Journal of Neuroscience*, 25(47):10913–10921, 2005.

[6] Anastasia Ludwig, Pablo Serna, Lion Morgenstein, Gaoling Yang, Omri Bar-Elli, Gloria Ortiz, Evan Miller, Dan Oron, Asaf Grupi, Shimon Weiss, et al. Feasibility analysis of semiconductor voltage nanosensors for neuronal membrane potential sensing, 2019.

[7] I Izeddin, J Boulanger, V Racine, CG Specht, A Kechkar, D Nair, A Triller, D Choquet, M Dahan, and JB Sibarita. Wavelet analysis for single molecule localization microscopy. *Optics express*, 20(3): 2081–2095, 2012.

[8] Raghuveer Parthasarathy. Rapid, accurate particle tracking by calculation of radial symmetry centers. *Nature methods*, 9(7):724–726, 2012.

[9] Yoshinobu Sato, Shin Nakajima, Nobuyuki Shiraga, Hideki Atsumi, Shigeyuki Yoshida, Thomas Koller, Guido Gerig, and Ron Kikinis. Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *Medical image analysis*, 2(2):143–168, 1998.

[10] Stefan Van der Walt, Johannes L Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. scikit-image: image processing in python. *PeerJ*, 2:e453, 2014.

[11] Christian L Vestergaard, Paul C Blainey, and Henrik Flyvbjerg. Optimal estimation of diffusion coefficients from single-particle trajectories. *Physical Review E*, 89(2):022726, 2014.

[12] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

[13] Tim Sainburg, Leland McInnes, and Timothy Q Gentner. Parametric umap embeddings for representation and semisupervised learning. *Neural Computation*, 33(11):2881–2907, 2021.

[14] Arthur Gretton. Reproducing kernel hilbert spaces in machine learning. page 133. URL `www.gatsby.ucl.ac.uk/~gretton/coursefiles/rkhscourse.html`.

[15] Danica J Sutherland, Hsiao-Yu Tung, Heiko Strathmann, Soumyajit De, Aaditya Ramdas, Alex Smola, and Arthur Gretton. Generative models and model criticism via optimized maximum mean discrepancy. *arXiv preprint arXiv:1611.04488*, 2016.

[16] **Verdier Hippolyte**, Duval Maxime, Laurent François, Cassé Alhassan, Vestergaard Christian L, and Masson Jean-Baptiste. Learning physical properties of anomalous random walks using graph neural networks. *Journal of Physics A: Mathematical and Theoretical*, 54(23):234001, 2021.

[17] Larry Wasserman. Hypothesis testing and p-values. In Larry Wasserman, editor, *All of Statistics: A Concise Course in Statistical Inference*, Springer Texts in Statistics, pages 149–173. Springer. ISBN 978-0-387-21736-9. doi: 10.1007/978-0-387-21736-9_10. URL https://doi.org/10.1007/978-0-387-21736-9_10.

[18] Wittawat Jitkrittum, Zoltán Szabó, Kacper P Chwialkowski, and Arthur Gretton. Interpretable distribution features with maximum testing power. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper/2016/hash/0a09c8844ba8f0936c20bd791130d6b6-Abstract.html.

[19] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, softcover reprint of the original 1st ed. 2006 edition edition. ISBN 978-1-4939-3843-8.

[20] Qiaoqiao Ruan, Patrick J Macdonald, Kerry M Swift, and Sergey Y Tetin. Direct single-molecule imaging for diagnostic and blood screening assays. *Proceedings of the National Academy of Sciences*, 118(14): e2025033118, 2021.

[21] Anders Gunnarsson, Arjan Snijder, Jennifer Hicks, Jenny Gunnarsson, Fredrik Hook, and Stefan Geschwindner. Drug discovery at the single molecule level: inhibition-in-solution assay of membrane-reconstituted β-secretase using single-molecule imaging. *Analytical chemistry*, 87(8):4100–4103, 2015.

[22] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[23] Martin Simonovsky and Nikos Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3693–3702, 2017.

VI – Software tools for experimental data analysis

In this chapter, I present the software tools that I developed in order to make my work more accessible to the broader research community. While Palmari is at a more advanced stage of its development and was released earlier this year, Tracktor and its web platform, are still under development but aim for a broader public. Besides, I briefly present Genuage, a virtual reality tool for visualization of three-dimensional point clouds.

VI.1 Palmari: from PALM movies to trajectories

Biomolecule trajectories are reconstructed from PALM movies after a sequence of processing steps, many of which admit a few parameters, which the user can leave to their default values or set according to the specifics of its acquisitions. I thought that an integrated tool comprising all processing step would be beneficial to those wishing to understand which step of the processing, and which parameter, were responsible for particular aspects of the final set of trajectories retrieved from their PALM movies. Moreover, having to analyse experiments comprising tens or hundreds of microscope recordings, I often had to resort to scripts parsing folders, manipulating and formatting files and calling functionalities from various software tools. I heard and read echoes of other scientists confronted to similar situations and dealing with the same sort of ad-hoc fixes. Both these reasons motivated me to develop a tool which would comprise both a graphic interface, so as to provide the essential visual feedback allowing one to choose appropriate values for the processing parameters, and a programmable interface for swiftly processing batches of acquisitions, able to leverage high performance computing resources.

The tool can be installed simply using `pip install palmari` or the Napari package installer, and its documentation is available online: https://palmari.readthedocs.io/. In the following, I explain the main functionalities and design choices.

VI.1.1 Graphical interface

I took advantage of the plugin interface of the open-source Python image viewer Napari [1] and developed my tool as an add-on of this software. It provides an interactive visualization of images as well as overlays of points and trajectories, especially useful to illustrate the steps of detection, localization and tracking.

The interface is shown in figure VI.1. On the right panel, processing steps are listed in chronological order, with their associated parameters. Each step adds its outputs as a layer

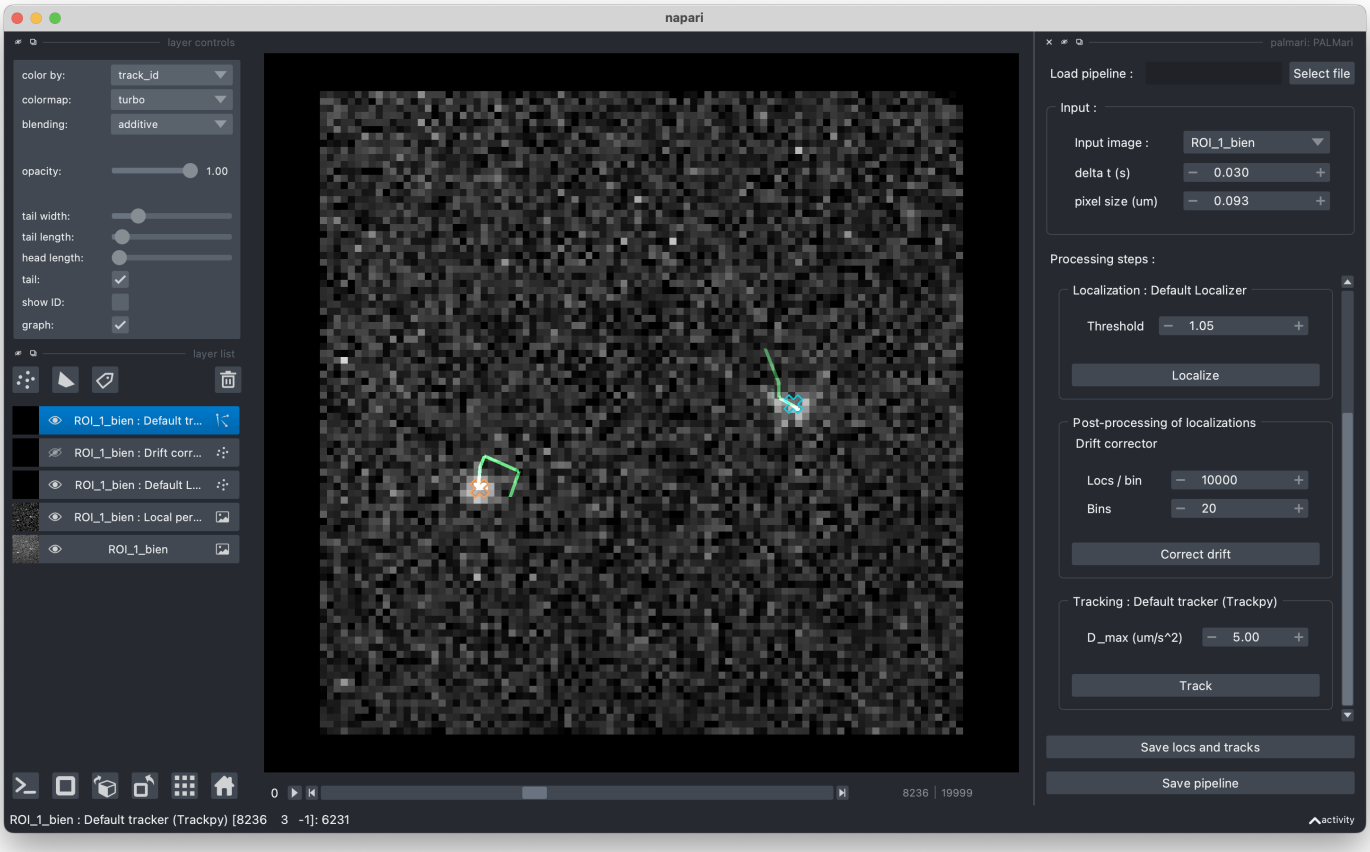

Figure VI.1: Palmari's graphical interface

to the viewer, whether images, points or trajectories. Layers are displayed in the central region, and listed in the left panel. They can be hidden, and viewed side by side and/or with stacked time slices, as shown in figure VI.2: the raw image is at the center of the bottom row, and the last layer containing reconstructed trajectories is at the top-left. This "mosaic" view (provided by Napari) is useful to have a quick overview of the processing pipeline, as it can for instance help pinpoint regions where localizations are not turned into trajectories or where the intensity is unexpectedly high on the image.

Processing steps are run one after the other, and can be re-run if one wishes to change the value of one parameter. It is thus easy to compare the results obtained with various parameter values, as shown in figure VI.3, where two values of maximum diffusivity are used for the tracking step. This particular parameter is for instance particularly critical when the density of particles is high, and it is thus useful to have a visual intuition of its effect.

Once one is satisfied with the parameters, the so-obtained processing pipeline can be exported under the form of a `YAML` file in order to be re-used later on other images (see VI.1). Similarly, it is possible to load a previously configured pipeline using the "load pipeline" field at the top of the right panel. Localizations and tracks can as well be exported in a table with tab-separated columns.

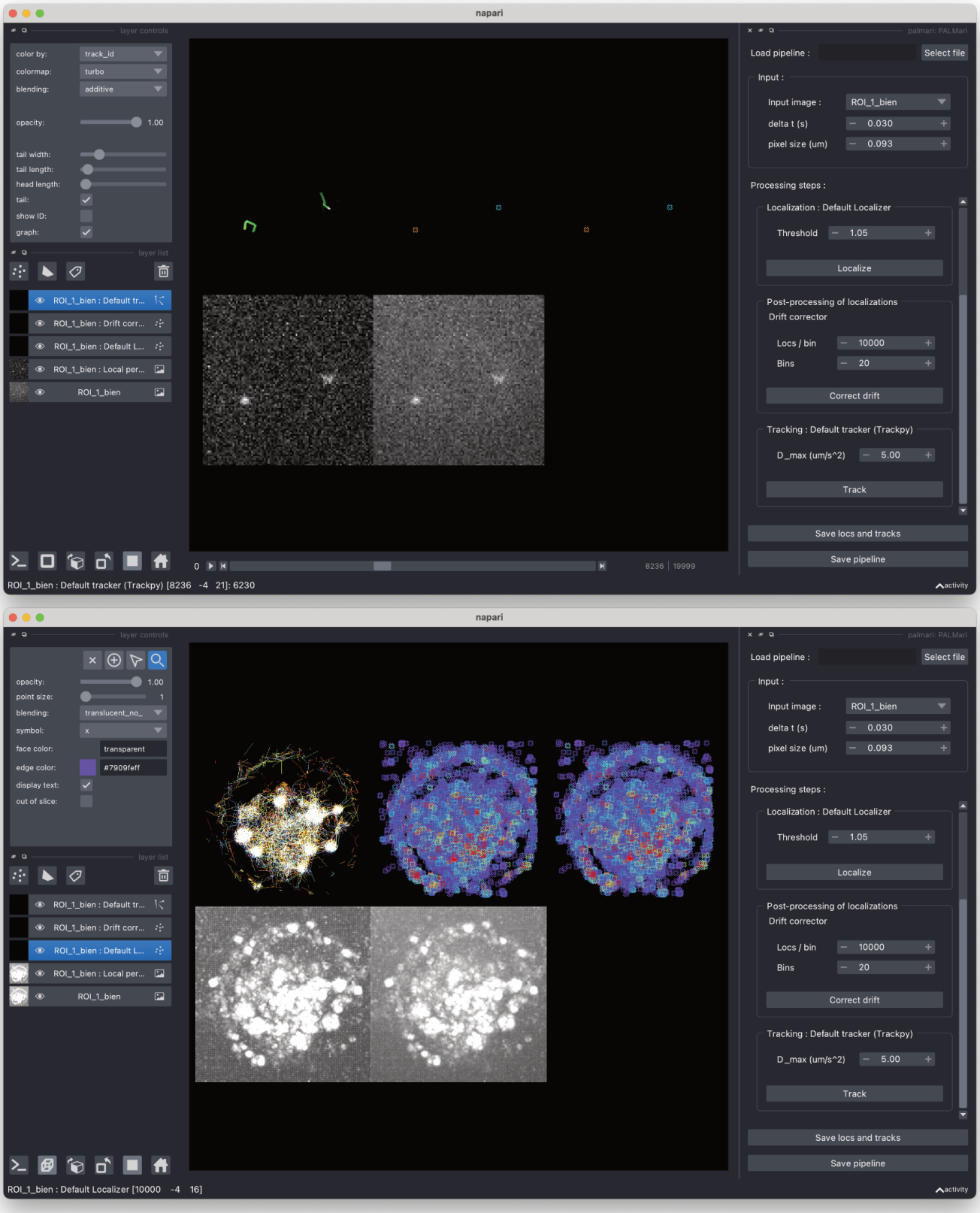

Figure VI.2: Mosaic view of the same movie's processing steps layers, with (top) and without (bottom) time stacking.

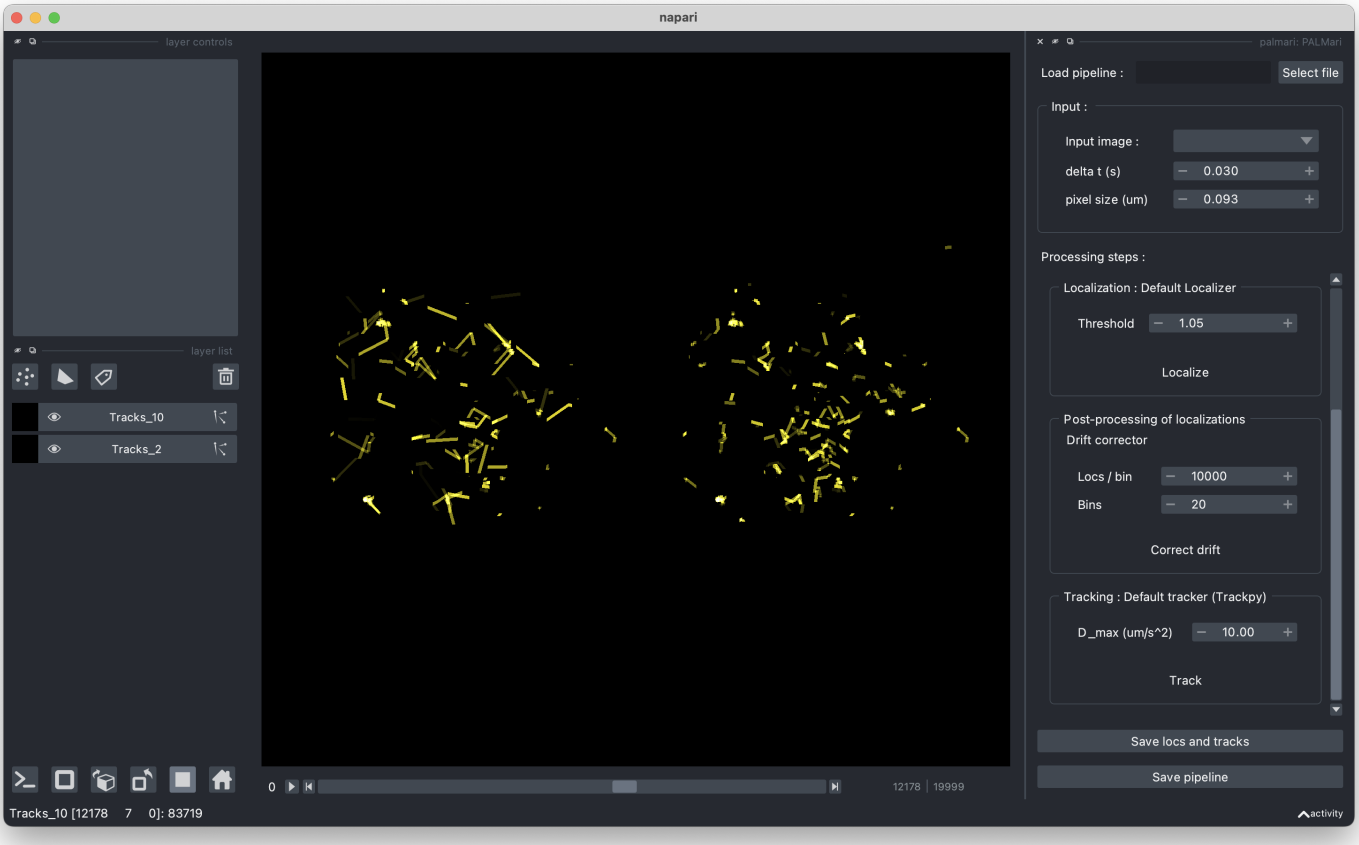

Figure VI.3: Comparison of tracks obtained with the maximum diffusivity set to 10 (left) and 2 (right) square microns per second.

Listing VI.1: Content of a YAML file describing a processing pipeline in Palmari

```
localizer:
  DefaultLocalizer:
    threshold_factor: 1.5
name: stricter_than_default
tracker:
  TrackpyTracker:
    max_diffusivity: 5.0
```

VI.1.2 Customizing the processing pipeline

Palmari comes with a set of basic processing steps, but the code is structured so that custom steps can be implemented by users with little effort, and possibly incorporated in future releases of the code so as to provide a greater choice of tools. Steps are split in four categories each corresponding to a parent class and custom steps must be implemented as classes inheriting these base classes. Categories of steps are summarized in table VI.1. A pipeline

can comprise several successive image and localization processing steps, but only one localizer and one tracker. When implementing a new processing step, specifying the type of its parameters (float, integer, boolean,...) is mandatory, so that a graphic interface allowing the user to set them can be automatically generated and incorporated to the Palmari widget.

Order	Type of step	Mandatory	Multiple sub-steps ?	Included
1	Image processing	No	Yes	`WindowPercentileFilter`
2	Localizer	Yes	No	`DefaultLocalizer`
3	Localizations processing	No	Yes	`DriftCorrector`
4	Tracker	Yes	No	`TrackpyTracker`

Table VI.1: Types of processing steps and their base class, to be inherited by eventual custom processing steps

VI.1.3 Programmatic interface

Palmari consider a set of acquisitions to be treated with the same parameters as an experiment. It is enough to specify the path of a folder to create an experiment from all the `.tif` files it contains. The user can specify rules to extract information from the hierarchy of sub-folders (if files are grouped by conditions or replicates for instance) or directly from the filename. The experiment object leaves the data folder untouched and stores all the information it needs in a separate folder specified by the user.

Pipelines can be run programatically in a few lines, either on a whole experiment or on single acquisitions. The Dask framework [2], which is used to handle images and tables of localizations, leverage all available computing resources by default and limits the memory footprint so as to prevent overflow. Palmari can thus be run on laptops as well as on large computing clusters, leveraging the resources of the latter without additional configuration. In a nutshell, Dask creates a graph of computation which it uses to perform operations on small slices of large arrays and concatenate results appropriately. Thus, only a limited amount of slices are loaded in memory at the same time, and independent operations can run in parallel. It requires little additional code and can be parameterized so as to restrict to a fraction of available resources, if desired.

Once the processing is over, it is possible, by calling `.view_in_napari()` to visualize the results in Napari. Palmari provides functions to compare results obtained via different pipelines, so as to compare, for instance, the number of localizations or tracks.

VI.1.4 Perspectives for improvement

A first type of improvement would be to enlarge Palmari's set of built-in processing steps, so as to offer alternatives to the basic ones included in the first version. The most needed ones would surely be the maximum-likelihood estimation of localizations, as well as the possibility to do detect several localizations in a given region.

Moreover, enabling a direct feedback between a step's parameter values and the resulting output would make the interface even more intuitive. For now, one has to run the step on the entire movie before seeing its output, which separates the action of setting the parameters

from seeing the results often by a few minutes. If the step was solely run on a temporal slice of the data, it would be possible to render results faster. Note however that for steps relating successive frames to each other (e.g. tracking), it might not always be possible to guarantee that the results on a given slice will remain exactly the same when the step will be run on the entire image.

Finally, it would be beneficial to the tool to be more generalist and not only apply to SPT-PALM data but also to other SMLM modalities. Many of the functionalities included in ThunderSTORM [3] could be replicated to converge towards this objective, which would also require the pipeline structure to be made more flexible (the tracking step, notably, might not always be necessary).

VI.2 Tracktor: a Python package for MMD comparisons of trajectories

Once removed the explicit mentions to synapses, axons, and the exact labels of conditions, the tools developed for the analysis presented in chapter V could actually be made generic, applicable to any experimental data consisting of sets of trajectories to be compared, and could serve to address various questions about these sets of trajectories. Therefore, I refactorized the code, which will soon be released under the form of a Python package named Tracktor. In the following, I explain the concepts of this multi-factor comparison method and detail its application to various analysis use-cases.

VI.2.1 Properties, groups and units

An experiment consists in a set S of trajectories, identified here to their latent vectors and for which the user has specified a number of *properties* indicating, for instance, the day on which trajectories were imaged, an identifier of the movie from which they originate, of the cell in which they evolve or of the algorithm used to reconstruct them from localizations. Properties can be specified both at the file level (all trajectories contained in the file then take the file's property value) and at the trajectory level if a more detailed separation is required (when organelles are tagged, for instance). Formally, we write as $\boldsymbol{\theta}_{\mathbf{r}} = (\boldsymbol{\theta}_{\mathbf{r}}^{(1)}, \dots, \boldsymbol{\theta}_{\mathbf{r}}^{(p)})$ the p-dimensional vector of properties of a trajectory $\mathbf{r}$. We denote V_i the set of values of the i-th property. For instance, we can have $\boldsymbol{\theta}_{\mathbf{r}} =$ ("File 1", "Cell 3") and $V_1 = \{"File1", "File2", "File3"\}$

Each set of properties thus defines a partition of S in *groups*: given P a subset of $\{1, \dots, p\}$, we can indeed define $V_P = \prod_{j \in P} V_j$ (Cartesian product) and, for all v in V_P, the group $G(v) = \{\mathbf{r} \in S | \forall i \in P, \theta_{\mathbf{r}}^{(i)} = v_i\}$, then S can be decomposed in disjoint groups as $S = \cap_{v \in V_P} G(v)$. The most fine-grained partition, obtained when P contains all properties which the user wants to consider, defines *units*. Each trajectory thus belongs to a single unit. For instance, in the case presented in chapter V, units corresponded in figure V.5 to the groups defined by the file and the "in/out of synapse" property, and in figure V.7, which focuses on inter-synapses variations, units were defined simply by unique synapse identifiers. To run comparisons, the user must specify which properties define units, and, if applicable, which ones define groups.

VI.2.2 Types of comparisons

The MMD comparison is based on a permutation test, let us here quickly recall the principle of these. Given two sets X and Y, the first step is to compute a distance $D = d(X, Y)$. Then, during a sampling phase, elements of X and Y are randomly split in sets X_i and Y_i, and a distance $D_i = d(X_i, Y_i)$ is computed. Then, the p-value of the test is estimated by dividing by the total number of splits considered the number of times that D_i was greater than D.

When comparing two groups, an equal number N of trajectories is sampled from each unit to compose the sets used to compute distances. For simplicity, the sub-sampling of units containing more than N trajectories is done once and for all at the start of the comparison. Units containing less than N trajectories are discarded. Groups are very often composed of a multitude of sub-units: the set of trajectories observed in a condition corresponding to a given group might originate from several movies for instance, or from several cells. By default, this information is not taken into account passed the construction of the sets X and Y to be compared and randomly mixed. One might however want to investigate whether all units contribute equally to the eventual difference that the MMD test might reveal between the considered groups. To address this question, Tracktor offers the possibility to do the sampling phase of the comparison ensuring at each iteration that trajectories of each given unit are in the same set. Thus, in an extreme case where the difference between two groups would only be due to one unit containing defect trajectories, these will always be in one same set and the distances computed in the bootstrap phase will not significantly exceed that of the one measured with the true partition of units across sets. This test is more conservative than the default comparison as its power is limited by the number of units n_1 and n_2 in each of the two groups: the p-value estimation's precision is lower-bounded by the inverse of the total number of possible permutations $\binom{n_1+n_2}{n_1}$. It is notably of interest when not much is known of the expected level of heterogeneity in a condition, as it allows one to probe whether two units from each groups are "on average" distinguishable.

Tracktor finally offers a last mean of probing the inter-group and intra-group variances, using a Multi-dimensional scaling (MDS) built from the MMD distance matrix of units, as shown on figure V.7. Coloring points on the scatterplot according to properties allows the user to understand which factors influence the dynamics of trajectories.

VI.3 Tracktor web: a web platform for trajectory analysis

The methods described in chapter V and implemented in the , though generic and applicable to trajectories originating from a broad spectrum of systems and acquired with various techniques, are not straightforward to run. Although I have released the necessary code under the form of Python packages, the entry barrier remains high for anyone unfamiliar with Python. Therefore, to widen the scope of potential users, I have initiated the development of Tracktor, a web platform allowing scientists to use the method on their trajectories without writing a single line of code. At the time of writing, this platform is under development, with the help of Tidiane Camaret N'Dir, but has already been beta-tested.

VI.3.1 Description of the functionalities

Users can upload files containing trajectories in a variery of formats (`.csv`, `.mat`, Trackmate...) if column labels are not provided, an algorithm based on the row ordering tries to automatically retrieve them. Files from which tracks could successfully be read are added to the index of files, which is user-specific and persists for a week. The interface provides a table for the user to specify information about files (corresponding to "properties" presented in the previous section), so as to group them in the subsequent analysis. Depending on the analysed experiment, labels can correspond to various criteria: cell type, imaged protein, biological replicate... At this stage, the application provides a viewer which allows the user to rapidly visualize trajectories present in each file. Once the user has selected the files to include, the analysis can be launched. Several analyses can be run successively and their results stored and accessed separately.

Upon termination of the analysis, results can be visualized and downloaded. The first tab of the results panel provides a selection of "classic" trajectory indicators such as the apparent diffusion, the anomalous exponent and predicted class of the model output by the neural network. All graphs are interactive and the user can choose according to which criteria to group files. The second tab allows one to visualize the latent space occupancy under the form of a two-dimensional scatter plot. The third tab shows a mapping of files obtained *via* multi-dimensional scaling of the MMD distance matrix. This gives an intuition of the relative importance of inter- and intra-group variability, and of the main "degrees of freedom" of the latent space occupancy densities. In the last and fourth tab, p-values of inter-group comparisons are shown, as well as the corresponding witness functions and representative trajectories, resembling the visualization showed in figure V.6.

VI.3.2 Architecture of the application

The two main pieces of the applications are the front-end, which the user accesses via a web browser, and the back-end, split in four sub-components, which handles requests coming from the front-end and performs the computations submitted through the web interface. The overall structure of the deployed application is schematized in figure VI.4. The compartmentalization of the app is well visible on the schema: blue boxes symbolize independent "pods", which are containerized and whose orchestration is managed by a Kubernetes cluster. At the cost of a relatively complex configuration, Kubernetes offers a level of abstraction which notably allows one to rapidly scale resources allocated to the application to match users' demand, and to monitor the activity of all its components.

The front-end is coded in JavaScript using the React framework, which allows the Hypertext Markup Language (HTML) content of web pages to be dynamically modified, as well as to send and receive Hypertext Transfer Protocol (HTTP) requests. Once compiled, the Javascript bundle is served by an Nginx instance and the code implementing the interface's logic is executed on the client side.

Requests reach the back-end through an Application Programming Interface (API) server pod, which handles them and sends appropriate responses. In the current version of the application, the API only handles request originating from the front-end, but in the future, it could possibly be adapted to communicate with other clients, such as a command-line tool or the Palmari plugin for instance. The server (i.e., the mechanism to answer requests) is coded in Python, using the Flask framework and a handful of Flask extensions. Users are

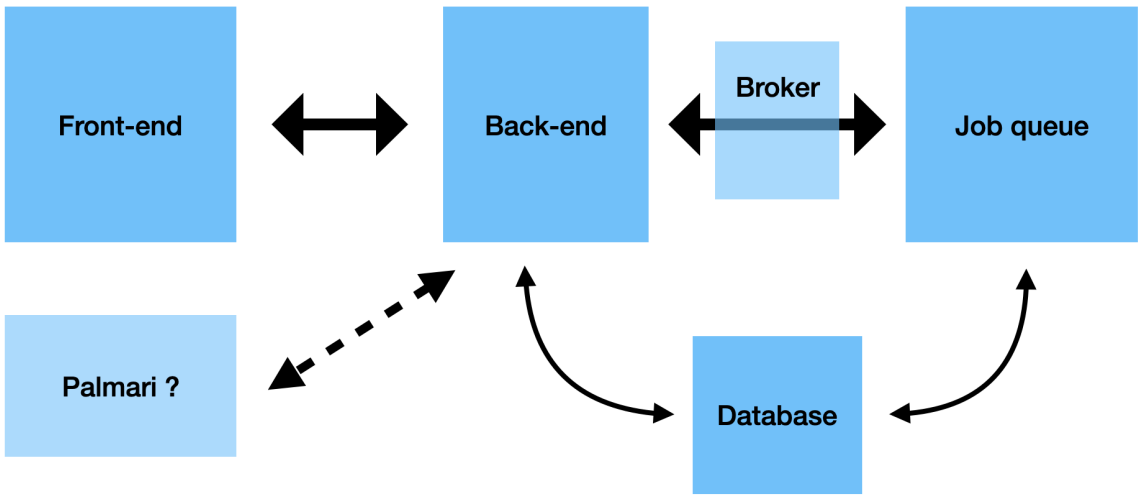

Figure VI.4: Tracktor's architecture. Blue boxes correspond to pods running independently, arrow symbolize communications between pods

identified using a session system requiring no login, via a key stored in a cookie on their browser. A new key is generated each time the site is visited by a user which has no such cookie, and a key's lifetime is prolonged of three days each time it is used to authenticate a user. This identification scheme allows the data uploaded by a user to be hidden from other users, and reciprocally. Users sessions are stored in a database, along with information on the uploaded trajectory files and submitted analyses.

When the API receives a request to launch an analysis, it passes an instruction to run the corresponding computation to a separate pod containing a task queue, implemented in Python using the Celery package. The instruction is carried between the two pods by a Redis broker. This queuing system limits the number of computations running concurrently, according to the resources allocated by the Kubernetes instance. When a computation terminates, the status of the analysis is updated in the database from "running" to "done", and the API can query the results and pass them to the front-end. Computations are made using the Python package presented in VI.2.

VI.3.3 Possible extensions

To provide even more flexibility, Tracktor could even offer a mean for users to upload their generative models (under the form of Python scripts for instance), on which the network could be trained. It could also offer a mean to run other standard inference and provide a larger set of conventional quantities: an MSD plot for instance, or more advanced estimations of the diffusivity and the localization error, using tools introduced in [4]

VI.4 Running Gratin interactively through a VR interface

The analysis of random walks by graph neural networks using the GRATIN method has also been included in the Genuage software, which leverages virtual reality to provide an intuitive

visualization of point clouds and trajectories in three dimensions [5]. This application developed by Thomas Blanc builds upon the DIVA project [6]. Genuage communicates with the Python script ready to receive trajectories, which it passes as input to a pre-trained neural network, sending the outputs back to Genuage. The processing of trajectories by the neural network occurs in a non-blocking way and the frame rate of the visualization remains high, so as not to cause discomfort to the user.

Bibliography

[1] Napari Contributors. napari: a multi-dimensional image viewer for python. *Zenodo https://doi.org/10.5281/zenodo*, 3555620, 2019.

[2] Matthew Rocklin. Dask: Parallel computation with blocked algorithms and task scheduling. In *Proceedings of the 14th python in science conference*, volume 130, page 136. Citeseer, 2015.

[3] Martin Ovesnỳ, Pavel Křížek, Josef Borkovec, Zdeněk Švindrych, and Guy M Hagen. Thunderstorm: a comprehensive imagej plug-in for palm and storm data analysis and super-resolution imaging. *Bioinformatics*, 30(16):2389–2390, 2014.

[4] Christian L Vestergaard, Paul C Blainey, and Henrik Flyvbjerg. Optimal estimation of diffusion coefficients from single-particle trajectories. *Physical Review E*, 89(2):022726, 2014.

[5] Thomas Blanc, **Verdier, Hippolyte**, Louise Regnier, Guillaume Planchon, Corentin Guerinot, Mohamed El-Beheiry, Jean-baptiste Masson, and Bassam Hajj. Towards human in the loop analysis of complex point clouds: advanced visualizations, quantifications and communication features in virtual reality. *Frontiers in Bioinformatics*, page 87, 2021.

[6] Mohamed El Beheiry, Charlotte Godard, Clément Caporal, Valentin Marcon, Cecilia Ostertag, Oumaima Sliti, Sébastien Doutreligne, Stéphane Fournier, Bassam Hajj, Maxime Dahan, et al. Diva: natural navigation inside 3d images using virtual reality. *Journal of molecular biology*, 432(16):4745–4749, 2020.

Conclusion & perspectives

Throughout this work, we have developed new approaches at the frontier of physics and amortized inference – whether variational or not – to model and analyze biomolecule dynamics. We took advantage of the flexibility of this framework to develop a new approach to characterizing trajectories, considering simultaneously several degrees of variability instead of focusing on conventional physical parameters, more precisely defined yet harder to estimate on short trajectories observed with noise. We have adapted graph neural networks to process trajectories and have demonstrated the potential of this architecture, well suited to capturing symmetries, on the analysis of random walks.

Concerning the neural network architecture, there are two topics which would be interesting to explore further. The first is the possibility to learn a truly scale-free estimation of the anomalous exponent of fractional Brownian motion, which, trained on trajectories whose length is bounded, could be applied to longer records and take advantage from all the information (including correlations at ranges exceeding the length of training trajectories) to provide more precise measurements. This would amount to enforcing a central limit theorem into the representation of trajectories learnt by the network. Another interesting development would be to replace the summary statistics vector with a distribution. This might provide an even clearer mapping of the random walk properties as it would disentangle the dynamical properties from the amount of information available about them.

Further research efforts should be oriented towards the interpretation of the findings originating from this new method. Spatial integration of dynamic properties is a natural extension of the work presented here. Comparing trajectories observed in neighbor regions could indeed reveal the contours of membrane domains and highlight local variations of their physical properties.

The interest of the methods developed in this paper notably lies in their applicability to a wide range of biological systems and questions. We hope that the released software tools will enable cell biologists to use these methods, or trigger the curiosity of other physicists who will develop them further.

Hippolyte VERDIER **mardi 4 octobre 2022**

Sujet : Inférence amortie de la dynamique des biomolécules

Résumé : Les trajectoire de protéines, observées en cellules vivantes grâce à la microscopie de localisation par photo-activation (PALM), sont révélatrices des propriétés à petite échelle de l'environnement de ces biomolécules. Un large spectre de dynamiques différentes ont été observées, et pour les caractériser de manière quantitative plusieurs méthodes ont été développées, spécifiques à certains systèmes biologiques ou au type de mouvement considéré. Pourtant, la présence simultanée dans le cytoplasme de différents types d'interactions crée des trajectoires composites, qui souvent s'éloignent des modèles canoniques de marches aléatoires pour lesquels sont conçus les estimateurs conventionnels. Par ailleurs, il a été montré que les réseaux de neurones fournissent des estimations plus précises des propriétés des marches aléatoires que les estimateurs analytiques. Dans cette optique, nous commençons dans cette thèse par présenter une architecture de réseau de neurones sur graphe (GNN) capable de traiter des trajectoires. Les représenter sous forme de graphes permet de tenir compte des symétries des trajectoires ainsi que des dépendances temporelles à de multiples échelles. De plus, cette architecture requiert nettement moins de paramètres que la plupart des autres réseaux de neurones.
Nous démontrons sa pertinence en l'utilisant dans le cadre d'une inférence amortie, pour mesurer des propriétés de trajectoires simulées, et vérifions son adaptabilité en l'appliquant à des trajectoires de modèles non vus à l'entraînement. L'inférence amortie doit son nom à la phase d'entraînement réalisée en amont, amortie lorsque les inférences sont réalisées rapidement sur les données expérimentales. Afin d'explorer plus avant le potentiel de cette architecture, nous la couplons ensuite à un réseau de neurones inversible permettant d'inférer des distributions. Nous inférons ainsi la distribution a posteriori des paramètres de trajectoires générées par le modèle de mouvement Brownien fractionnaire. L'existence d'une vraisemblance analytique pour ce modèle permet de minorer la variance atteignable par un estimateur non-borné, à laquelle nous comparons celle obtenue par notre estimateur. Alors que sa complexité algorithmique est fonction linéaire de la longueur des trajectoires (là où celle de l'estimateur de vraisemblance maximale est quadratique), notre estimateur amorti atteint une précision proche de l'optimalité. La méthode permet en outre de mesurer un temps caractéristique au-delà duquel s'effacent les corrélations du mouvement – un aspect souvent négligé dans l'analyse de trajectoires expérimentales. Puis nous présentons une méthode de caractérisation du mouvement basée sur la représentation vectorielle des trajectoires, générée par le GNN. Nous estimons la significativité des différences entre des ensembles de trajectoires observées expérimentalement. Par rapport aux méthodes conventionnelles, cette méthode a l'avantage de considérer conjointement une variété de critères, permettant notamment de mesurer la variabilité observée au sein d'une même condition ou entre plusieurs conditions biologiques, sans formuler au préalable d'hypothèse quant à la nature de cette variabilité. Les différences relevées par la méthode peuvent être interprétées simplement. Enfin, nous avons développé une plateforme web permettant aux scientifiques non-programmeurs d'utiliser cette méthode d'analyse sur leurs trajectoires, à l'aide d'une interface graphique. Les résultats peuvent être exportés ou visualisés sur la plateforme. Cet outil se veut généraliste et n'est pas limité à l'analyse de trajectoires acquises en PALM. Puisque les hypothèses sur lesquelles elle repose sont assez permissives, cette méthode ouvre la voie vers l'automatisation de la pharmacologie basée sur la microscopie de localisation par molécule unique. Nous l'avons déjà testée sur plusieurs systèmes, comme l'alpha-synucléine observée dans les synapses et divers récepteurs membranaires.

Mots clés : Inférence amortie, inférence variationnelle, marches aléatoires, apprentissage profond, microscopie de localisation par molécule unique, biophysique, tests statistiques

Subject : Amortized inference of biomolecule dynamics

Abstract: Photo-activated localization microscopy (PALM) enables high-resolution recording of single proteins trajectories in live cells, providing precious insights of small-scale properties of biomolecules environment. A broad spectrum of biomolecule dynamics have been observed and analysis schemes tailored to specific biological systems and random walk models have been devel- oped to quantitatively characterize their motion. Yet, in most cellular environments, the complex interplay of interactions creates trajectories of composite nature, often not resembling canonical models of stochastic motion and thus challenging the assumptions on which tailored estimators rely. Neural networks have been shown to provide better estimations of random walk parame- ters than analytical methods. In this thesis, we first introduce a graph neural network (GNN) architecture able to process trajectories. Representing trajectories as graphs allows to account for their symmetries as well as for time-dependencies which might exist at various scales, while requiring substantially fewer parameters than conventional neural architectures. We demonstrate the relevance of this architecture by using it in an amortized inference scheme, to infer properties of simulated trajectories. We furthermore demonstrate its robustness to trajectories generated by models unseen during training. As this inference method requires an upfront, computationally intensive, training phase before it can be used to perform inference with a limited number of computation steps, it is named "amortized".
In order to further demonstrate its potential, we couple the GNN module with an invertible network so as to perform variational inference. We apply this to infer the posterior distribution of parameters of fractional Brownian motion trajectories, for which the existence of a tractable likelihood allows us to compute a lower bound of the variance reachable by an unbiased estimator. We show that our estimator, whose marginal complexity, one trained, scales linearly with the trajectory length (with which the maximum-likelihood estimator scales quadratically), reaches a precision close to optimality. Besides, we show that this scheme can be used to measure an eventual cut-off time in temporal correlations, an aspect of biomolecule dynamics which is often discarded. We then present a trajectory characterization method in which we take advantage of the fixed-size vector of summary statistics computed by the GNN for each trajectory. Using statistical tests based on the maximum mean discrepancy, we assess the significance of differences between sets of experimentally observed trajectories. This characteriza- tion has the advantage over conventional ones to provide a holistic description of random walks, encapsulating a variety of aspects of the dynamics. This notably measure, without prior assump- tions regarding its nature, intra- and inter-condition variability. Besides, we provide means of interpreting the nature of the eventual differences, so as to pinpoint subtle changes in dynamics not necessarily captured by traditional indicators. Finally, we developed a web platform allow- ing other researchers to perform the analysis on their trajectories via a graphical interface, with visualization and exports of results. This is intended to be a generalist trajectory analysis tool, able to process large batches of experiments and whose scope could extend beyond SPT-PALM. Thanks to its very permissive assumptions, the analysis method presented in this thesis paves the way to automated single molecule-based pharmacology: we have tested the approach on a variety of biological examples, from alpha-synuclein in synapses to immune checkpoints at the membrane of T-cells.

Keywords : Amortized inference, variational inference, random walks, deep learning, single molecule localization microscopy, biophysics, statistical tests