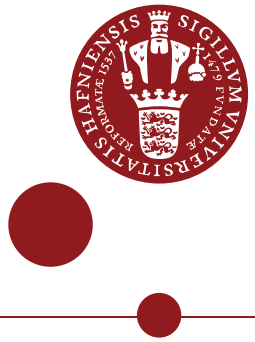
UNIVERSITY OF COPENHAGEN
NIELS BOHR INSTITUTE



Ph.D. Thesis

Mind the Gradient

Differentiable Computational Methods in Microorganism Behaviour Studies

Albert Alonso

Advisor: Julius Bier Kirkegaard

This thesis has been submitted to the Ph.D. School
of The Faculty of Science, University of Copenhagen
on November 30, 2024.

The good of the scorpion is not the good of the frog, yes?

—Charles "Charlie" Kelly

Abstract

At the microscopic scale, where viscosity dominates motion and sensing is constrained by physical limits, microorganisms rely on strategies highly adapted to their environments. This thesis uses differentiable programming techniques to develop computational methods and mathematical models to explore navigation, sensory integration, and behavioural adaptations under such physical constraints.

The first project addresses the challenge of tracking overlapping organisms in microscopy images – a critical step for understanding behavioural changes. I present a deep learning-based method capable of assigning identities during occlusions and detecting thousands of organisms simultaneously. Trained on synthetic data of simulated nematode motion in viscous environments, the model enables the analysis of massively dense populations.

The second and third studies focus on decision-making during chemotaxis, where physical constraints shape optimal sensing strategies. Using reinforcement learning, I find a continuous transition between temporal and spatial sensing, identifying a regime where integrating both results in a more efficient navigation. Furthermore, spatial sensing in amoebas is modelled as a finite-resource competition between protrusions, revealing how persistence in motion enables cells to overcome the physical limitations in sensing, in agreement with experimental observations.

The fourth work explores the optimal placement of cell-surface receptors for gradient estimation, leading to clustering at the tip of symmetry-breaking protrusions. These findings suggest an evolutionary link between physical mechanisms and functional advantages. For the fifth and last project, the adaptation of biological transport networks is modelled, showing how nodes optimally distribute based on resource delivery costs and environmental constraints.

This thesis showcases how the integration of differentiable computational methods and physical models can advance our understanding of microbial behaviour, and offers a foundation for future applications.

Resumé

Mikroskopiske organismer udnytter nøje tilpassede strategier for at overvinde de fysiske love, der dominerer deres miljø: viskositet og termisk støj. Denne afhandling anvender differentierbar programmering til at udvikle matematiske metoder og modeller til at undersøge navigation, sensorisk integration og adfærdsadaptationer under disse fysiske begrænsninger.

Det første projekt takler problemet med at detektere overlappende organismer i mikroskopibilleder et kritisk skridt for at forstå adfærdsændringer. Jeg præsenterer en deep-learning metode, der hurtigt kan identificere tusindvis af organismer på trods af overlap. Modellen er trænet på syntetiske data af simuleret nematode-bevægelser i viskøse miljøer, og muliggør analyse af ekstremt tætte populationer.

De andet og tredje projekt fokuserer på beslutningstagning under chemotaxis, hvor basal fysik begrænser de mulige sanse-strategier. Ved brug af reinforcement learning finder jeg en kontinuerlig overgang mellem tidslig og rumlig sansning, og identificerer fysiske parametre, hvor integration af begge disse metoder resulterer i mere effektiv navigation. Herudover modelleres rumlig sansning i amøber som en konkurrence om begrænsede ressourcer, hvilket afslører, hvordan vedholdenhed i bevægelse gør det muligt for celler at overvinde fysiske begrænsninger i sansning, i overensstemmelse med eksperimentelle observationer.

Det fjerde projekt undersøger den optimale placering af receptorer i celleoverfladen for at estimere retning, hvilket fører til en sammenklumpning af receptorer. Disse resultater påpeger en evolutionær forbindelse mellem fysiske mekanismer og funktionelle fordele. For det femte og sidste projekt modelleres adaptation af biologiske transportnetværk og viser, hvordan knudepunkter fordeler sig optimalt for a minimere omkostningerne for distribuering af ressourcer.

Denne afhandling demonstrerer, hvordan integrationen af differentierbar programmering og fysiske modeller kan fremme vores forståelse af mikrobiel adfærd og tilbyder en platform for fremtidige anvendelser.

Resum

A escala microscòpica, on la viscositat domina el moviment i la detecció està limitada per restriccions físiques, els microorganismes depenen d'estratègies altament adaptades als seus entorns. Aquesta tesi utilitza tècniques de *differentiable programming* per desenvolupar mètodes computacionals i models matemàtics que exploren la navegació, la integració sensorial i les adaptacions de comportament sota aquestes limitacions físiques.

El primer projecte aborda el repte de fer seguiment d'organismes superposats en imatges de microscòpia, un pas crític per comprendre els canvis de comportament. Presento un mètode basat en *deep learning* capaç d'assignar identitats durant les oclusions i detectar milers d'organismes simultàniament. Entrenat amb dades sintètiques de simulacions del moviment de nematodes en entorns viscosos, el model permet analitzar poblacions massivament densificades.

El segon i tercer estudi se centren en la presa de decisions durant la quimiotaxi, on les restriccions físiques configuren les estratègies òptimes de detecció. Mitjançant *reinforcement learning*, identifico una transició contínua entre la detecció temporal i espacial, descobrint un règim on la integració d'ambdues estratègies resulta en una navegació més eficient. A més, la detecció espacial en amebes es modelitza com una competició de recursos finits entre protrusions, revelant com la persistència en el moviment permet a les cèl·lules superar les limitacions físiques en la detecció, en concordança amb les observacions experimentals.

El quart treball explora la col·locació òptima de receptors a la superfície cel·lular per a l'estimació de gradients, resultant en una agrupació a la punta de les protrusions quan es trenca la simetria. Aquests resultats suggereixen una connexió evolutiva entre mecanismes físics i avantatges funcionals. Finalment, en el cinquè projecte, es modelitza l'adaptació de xarxes de transport biològic, mostrant com els nodes es distribueixen de manera òptima basant-se en els costos de lliurament de recursos i les limitacions del domini.

Aquesta tesi demostra com la integració de mètodes computacionals diferenciables i models físics pot avançar la nostra comprensió del comportament dels microorganismes i ofereix una base per a futures aplicacions.

Acknowledgements

Proper etiquette dictates that the first person acknowledged in a thesis should be your supervisor, so I would like to begin by sincerely thanking Julius. His guidance and support have been a major reason why I truly enjoyed this experience. I also deeply appreciate his openness, always allowing me to make jokes without censoring – a true mark of a good supervisor.

Although we had to move from NBI halfway through the ph.d., my connection to Biocomplexity has still remained strong. I genuinely believe that Biocomplexity and its people are unequivocally the best part of NBI. Miguel, I wanted to say something funny about Amager along with my gratitude but I think I've run out of jokes about it... so now I can't properly thank you enough for everything, but I hope you know. Similarly, it would be hard to name every reason I should thank Malthe for, so I won't, but thank you. To Lukas, Anja and Alessandra, I've pleasantly enjoyed all our interactions – and even though I was not named in someone's thesis, I am not the kind to hold grudges (oh wait...). Jonas, you too. It would be very wrong if I did not deeply thank Tuan, the travelling scholar, for undoubtedly being the best part of this ph.d. journey, and for being a good friend. And Bente, the kindest and most incredible person I have had the pleasure of meeting – a true force for good. If I could save only one person from a fire, I would probably cut both Bente and Tuan in half to save half of each. To the young experimentalists, Stanley and Adrien, let your future research career be bright and lead you to nice office views. Speaking of young, I'll finish this paragraph by deeply thanking my cross-armed very Norwegian office mate, or as he usually refers to me, my brother from another mother, Lars Erik.

Many thanks to Robert, whose support made my time abroad an unexpectedly transformative part of my ph.d. Antonio and Martina, thank you for indulging me during my jolly old time in London town. I will never forget that I shared office with a bronze medallist from Antequera – who could? I should also acknowledge the other wing at Imperial. My heartfelt thanks to Elephes Sung, Angela, Smitha, Daniel, Sergio, Emma, Charlotte, and their supervisor – whose name strangely escapes me at the moment, but thank you nonetheless!

To the IMAGE section (pending name?), it has been a truly nice experience to spend the final period of my ph.d. there. Thanks to my fellow phds: my countryman Michele, the not-so-Arab Reza, the Slavs Anna and Tudor, and lastly, the less exotics Bjørn, Roberta and Malte, and, of course, Peidi. A special mention to the angry German ghost of DIKU, Thomas, for his invaluable help during this final phase.

I would mention my non-academic friends, but I have a strong suspicion they are not going to read this – but if they do, then I'll owe them a beer. Finally, I am grateful to my family, whose expertise in theoretical physics and complex mathematical abstractions have always been very insightful.

P.S. Oh! And thanks to Mireia!

To those who audibly gasped upon learning I'm still in my twenties: your acknowledgement is spitefully retracted. And if I've forgotten someone (likely), feel free to reproach me.

List of Publications

The work presented in this thesis has lead to the following publications.

1. Albert Alonso and Julius B. Kirkegaard. Fast detection of slender bodies in high density microscopy data. *Nature Communications Biology*, 6(1):1–12, 2023. doi: 10.1038/s42003-023-05098-1.

2. Albert Alonso and Julius B Kirkegaard. Learning optimal integration of spatial and temporal information in noisy chemotaxis. *PNAS Nexus*, 3(7), 2024. doi: 10.1093/pnasnexus/pgae235

3. Albert Alonso, Julius B. Kirkegaard, and Robert G. Endres. Persistent pseudopod splitting is an effective chemotaxis strategy in shallow gradients, 2024. (preprint)

4. Albert Alonso, Robert G. Endres, and Julius B. Kirkegaard. Receptors cluster in high-curvature membrane regions for optimal spatial gradient sensing, 2024

5. Albert Alonso, Lars Erik J. Skjegstad, and Julius B. Kirkegaard. Adaptive Node Positioning in Biological Transport Networks, 2024. (preprint – shared first author)

Furthermore, work during the PhD not included in this thesis, has lead to the following publication

1. Tuan Pham, Albert Alonso, and Karel Proesmans. Irreversibility in Non-reciprocal Chaotic Systems, 2024. (preprint)

Contents

Preface

Here lies my ph.d.

This thesis encompasses my works, or at least those considered interesting enough to be converted into papers, of the last three years. It is a synopsis thesis, which is just a fancy way to say that I did not need to rewrite my projects. Instead, I have just placed the contents of the manuscripts and called them Chapters – it was more work than it sounds like...

Each chapter, i.e., research project, is on its majority independent from the rest. As in many theses before this one, the challenge during writing has been to present these projects as though they were part of a carefully planned and cohesive narrative. My attempt at this can be found in the introductory chapter. Since the topics are a bit diverse, the topics that need to be in the introduction are not very well defined. Instead, my approach has been to describe the projects and their motivation, including any relevant information I found appropriate along the way. Nevertheless, these are individual works, and one should not need to be familiar with the others to follow a specific one.

It is my firm intention that this thesis conveys the *voice* of its author – yours truly. Hence, if the text is flawed, inconsistent, subtlety sarcastic, and slightly informal, it is as per my design. It was my aim to be concise and straightforward, but the results may show some variation from that original goal. Ironically, I find that to be a nice way to show that a LLM[1] has not been tightly involved in the writing process, as using it would probably result in considerably better texts.

[1]The scientific community has not agreed on how to feel with regards to the usage of such tools, yet. Neither have I.

Chapter 1

Introduction

It may be a strong misconception that the underlying mechanisms of real biological systems can be trivially explained. Nevertheless, this is *precisely* what I have tried to do in the works presented in this thesis. Biological systems are complex, and here, I will treat them using physics principles and computational methods that either assume a great deal of simplifications or hide their complexity under hard-to-peak black boxes. This approach allows for the interpretation of the systems at their most essential representation from which, and in spite of my limited knowledge about the topics at hand, insights at their most fundamental level can be obtained.

This thesis focusses on employing advanced differentiable techniques to study and understand – from a physics perspective – the behaviour of microorganisms. It would be fair to remark that the studies presented here lie at the intersection between physics, biology, and computer science, hopefully showcasing that while *a jack of all trades is a master of none, is oftentimes better than a master of one*.

1.1 Research Motivation

Microorganisms, the simplest and most abundant form of life, exhibit incredibly sophisticated, intricate, and diverse behaviours. Assuming universality of such is likely a fool's mistake, as each microorganism can demonstrate unique responses to its environment [7]. Yet, a characteristic that distinguishes physicists from the wider scientific community is our tendency to boldly employ assumptions and simplifications when solving a problem in order to unlock possible paths to solutions. When such a strategy is applied to biological systems, underlying truths about them can be uncovered, if not hinted.

There are a wide range of ways to study the behaviours of organisms [8]. Of great interest (to me) is the microscopic realm [9], where our macroscopic intuition about sensing [10] and movement [11] tends to fail. At these scales, the limits imposed by physical laws often dictate how microorganisms interact with their surroundings [12] and respond to stimuli [13]. Understanding these can provide critical insights that can

then be transformed into more practical innovations.

Now, this is neither a theoretical nor an experimental thesis, but a computational one. The role of computational models is crucial to study complex behaviours and predict outcomes that are simply too difficult to observe or control experimentally. It is through the use of computational techniques, such as the ones I will present in this thesis, that more fundamental natural laws can then be found. However, the simulations should align and reproduce the experimental results. In the cases where those are lacking, nonetheless, computational simulations are particularly useful in exploring interesting ideas and hypothetical scenarios, not unlike those presented here.

In this thesis, I demonstrate the application of differentiable programming methods [14] to address unsolved or unanswered questions related to cell motility [1], decision-making [2, 3], cell-sensing [4], and even optimal fluid transport [5]. Hopefully, these may lay the foundations for the exploration of the underlying fundamental principles that describe these systems and, in doing so, showcase the integration of novel computation techniques with fundamental studies on biophysical systems [15].

1.2 Differentiable Programming

Extremely relevant in physics and intrinsic to life itself, as this thesis should demonstrate, is the need to compute gradients. After all, differentiation is everywhere (without it, everything would be quite static and unchanging). It is for that reason that differentiation is one of the first topics scientists need to master when starting their advanced studies in natural sciences. When we learn the basics of differentiation, we typically deal with simple expressions that are somewhat possible to derive. These often involve just a few variables in a well-behaved domain and can easily be solved with pen and paper. For more complex expressions, symbolic calculation methods allow many scientists to acquire, albeit occasionally cumbersome, exact expressions [16, 17]. In the more trickier cases, we have happily resorted to approximate solutions by calculating differences in small changes using numerical methods [18, 19]. Honourable mention here is the family of finite difference methods that have powered scientific research for decades. Having said that, it is when dealing with more complex systems – those with large numbers of parameters or intricate non-linear relationships – that one cannot help but notice (and lament) their limitations. It is only logical that these errors increase as the need for precision grows, especially when higher-order derivatives or multiple variables are invited to the party. Thankfully, a solution to this problem (explained just below), suggested as early as the mid-20th century [20], has finally been put into practice and has recently achieved the widespread recognition it likely deserves.

1.2.1 Automatic Differentiation

Automatic Differentiation (AD) manages to deal with these difficulties by offering an approach that is both exact and scalable [20, 21]. Using the same logic as when we manually derive expressions, instead of seeing a function as one large monolith expression, AD

breaks it down into simpler elementary operations with well-known differentiation rules, where it systemically applies the chain rule of calculus [22] to obtain the target value. In other words:

Let $f_1, \ldots, f_n$ be a (large) collection of elementary functions with known derivatives, then for any composition of these such as $g(x) = f_i^{(m)}(\cdots(f_k^{(j)}(\cdots(f_l^{(1)}(x))\cdots)\cdots)$ where $i, k, l \in \{1, \cdots, n\}$ and the parentheses above indicate the order of application of the function $j = 1, \ldots, m$, the derivative of g can be obtained as

$$\frac{\mathrm{d}g}{\mathrm{d}x} = \frac{\mathrm{d}f_i^{(m)}}{\mathrm{d}f_k^{(m-1)}} \cdots \frac{\mathrm{d}f_k^{(j)}}{\mathrm{d}f_l^{(j-1)}} \cdots \frac{\mathrm{d}f_l^{(1)}}{\mathrm{d}x}. \tag{1.1}$$

This rather simple approach allows for any order derivatives to be computed with machine precision, avoiding the errors and instability mentioned before. Naturally, obtaining an explicit expression is out of the question, but if your primary concern is evaluating the derivative – as is often the case in scientific computing – then this method performs exceptionally well.

In the core of AD is the concept of a computational graph – a directed acyclic graph (DAG) illustrating how information *flows* through a function [21]. Every node in this graph represents a basic (elementary) operation , with the edges illustrating the passing of data and derivatives between these operations. This graph does not only reflect the functional architecture of the problem; it also forms the backbone for AD to follow and compute the exact gradients. By constructing and navigating this graph, AD seamlessly propagates derivatives throughout the entire function. These graphs enable the handling of more complex combinations of functions, as opposed to the straightforward, sequential graph that characterizes the scenario described by Eq. (1.1).

There are two main modes in which AD calculates the propagation of gradients in a graph: forward-mode and reverse-mode (see Figure 1.1 for a simple example of each mode). In forward-mode, derivatives are calculated at the same time as the function is evaluated, which makes it ideal for problems involving fewer input parameters than outputs. At each step, the following calculation is done

$$\frac{\mathrm{d}f_i^{(q)}}{\mathrm{d}x} = \frac{\mathrm{d}f_i^{(q)}}{\mathrm{d}f_j^{(q-1)}} \frac{\mathrm{d}f_j^{(q-1)}}{\mathrm{d}x}. \tag{1.2}$$

During this process, the program computes and propagates a derivative value at each step, often referred to as tangent value or tangent vector, while navigating the computational graph. This comes at the cost of some additional storage memory [23]. However, each pass involves matrix-vector multiplications to calculate gradients, which can become computationally expensive in high-dimensional output spaces. The forward-mode aligns with the natural way of performing these calculations when the number of inputs is relatively small compared to the number of outputs, because derivatives are calculated *on the go* as the program evaluates the function. On the other hand, when the function has many parameters that result in a single-scalar output, as is the case for *deep-learning* models, where parameters are optimised following the gradient of a single

loss value, reverse-mode AD becomes the more sensible method. Reverse-mode starts by evaluating the function and then traces back the derivatives from the output to the inputs, such as

$$\frac{\mathrm{d}f_i^{(m)}}{\mathrm{d}f_k^{(q-1)}} = \frac{\mathrm{d}f_i^{(m)}}{\mathrm{d}f_i^{(q)}} \frac{\mathrm{d}f_i^{(q)}}{\mathrm{d}f_k^{(q-1)}}. \tag{1.3}$$

Although reverse-mode requires an additional pass through the computational graph to propagate the derivatives backward, the computational benefits generally outweigh this cost for scalar-valued outputs. Instead of propagating tangent values, reverse-mode propagates adjoint values – representing the contributions of each variable to the output gradient. In these cases, reverse-mode reduces the computational cost of gradient calculation by replacing Jacobian-vector products (used in forward-mode) with vector-Jacobian products, which are computationally more efficient when the number of outputs is much smaller than the number of inputs[1].

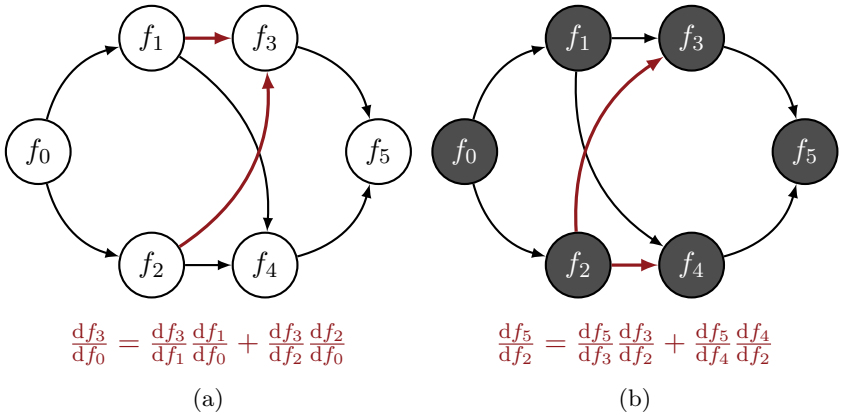


Figure 1.1: Computational graph diagram where (a) Forward mode differentiation and (b) Reverse mode differentiation are shown. The final objective of both methods is to obtain $\frac{\mathrm{d}f_5}{\mathrm{d}f_0}|_{x^*}$ with completely different approaches to obtain the gradients.

The capabilities of AD are likely a significant factor behind the surge in machine learning applications; merely attempting to manually write the backpropagation equations, i.e., calculating the gradients using the reverse-mode approach, on a ResNet [24] highlights how discouraging it would be to train new models. Yet, its benefits are not constrained to deep learning models. In Physics, AD is applied to simulate physical systems in a wide range of fields such as in fluid dynamics [25], quantum mechanics [26–29], classical mechanics [30, 31], astrophysics [32] and biophysics [4, 5, 33] among others. Within optimisation algorithms, AD is critical to efficiently relax a system to its optimal configuration.

[1]The efficiency difference arises from the Jacobian's shape and the algorithm used. Forward-mode computes Jacobian-vector products efficiently for tall Jacobians (more rows than columns), while reverse-mode computes vector-Jacobian products efficiently for wide Jacobians (more columns than rows), avoiding explicit Jacobian construction and leveraging gradient accumulation.

Overall, AD is an effective method to obtain gradient evaluations. However, it comes with challenges, including the overhead of maintaining computational graphs and the associated memory usage. Furthermore, AD struggles with non-smooth functions, where derivatives may not exist or where discontinuities complicate the propagation of gradients[2]. Addressing these challenges is an ongoing area of research, with new algorithms and techniques being developed to improve its efficiency and broaden its applicability [34, 35].

Similarly, with the rise in usage of parallel accelerators such as Graphical Processing Units (GPUs) and Tensor Processing Units (TPUs) in scientific computing [25, 36], automatic differentiation (AD) frameworks such as JAX [37], PyTorch [38], AutoGrad [39] and diffTaichi [40] with support for such hardware have cemented their status as essential tools for computational scientists. Recent hardware advances have also resulted in significantly more efficient implementations, enabling AD to tackle increasingly vast and intricate systems. Looking ahead, the integration of AD with emerging hardware paradigms, such as Stochastic Processing Units (SPUs) [41] – which relies on thermodynamic principles to perform linear algebra – offers an interesting future for the scope and efficiency of computational sciences.

1.2.2 Deep Learning

While most researchers must already be familiar with the concept of deep learning, let it be briefly introduced here.

Deep learning (DL) models aim to approximate a function $f : \mathbb{R}^n \to \mathbb{R}^m$ that maps inputs $\boldsymbol{x}$ (e.g., pixel intensities from images) to outputs $\boldsymbol{y}$ (e.g., classifications or trajectories), by learning an optimal set of parameters θ such that $\boldsymbol{y} \approx f(\boldsymbol{x}; \theta)$. The function f is commonly built as an artificial neural network [42], which in its simplest form is just a sequence of matrix multiplications

$$x^{(i+1)} = \sigma(W_\theta^{(i)} \cdot x^{(i)} + b_\theta^{(i)}), \tag{1.4}$$

where $x^{(0)}$ is the input, $W^{(i)}$ and $b^{(i)}$ are the weights matrix and bias vector at the i-th layer, and σ is a non-linear activation function.

The optimisation of $W_\theta, b_\theta \in \theta$ is achieved through iterative optimisation (training steps), where a scalar loss function $\mathcal{L}(\boldsymbol{y}, f(\boldsymbol{x}; \theta)$ quantifies the difference between predictions and true values, and its gradient – calculated via backpropagation – guides parameter updates using stochastic gradient descent (SGD).

At its foundation, DL builds on the Universal Approximation Theorem [43], which states that a feedforward neural network with a sufficient number of neurons can approximate any continuous function, given a suitable set of activation functions. This explains DL's ability to learn complex relationships between inputs and outputs across many domains [44]. Notably, DL assumes models of many layers of neurons (hence

[2]One way to see if someone has struggled with such cases can be seen on how they approach in constructing the square root of a distance that may be zero. This will likely reveal whether they have faced divergent gradient calculations and its unforeseen complications.

the deep term), where each layer transforms the input variables into progressively more abstract representations, which makes them particularly capable of handling intricate patterns in high-dimensional data, such as images, text, and speech [45, 46].

These models are often described as *black boxes* since it is rather hard to obtain a proper meaning of the resulting parameters θ. Note, however, that the reason for their success lies in their capability to learn from data rather than relying on manually defined rules. This capability has made DL an essential tool in data-rich fields where conventional approaches struggle, often due to the sheer complexity of the data.

1.3 Summary of Projects

This section provides an overview of each research project, explaining their motivation, how they relate to the study of microorganism behaviour, and the relevant physics and biological principles involved. It also summarises the key results presented in the papers, as well as the role of AD in their implementation. Section 1.3.1 introduces the work in Chapter 2, focussing on the computer-vision challenge to track overlapping bodies in microscopy images and the physical principles involved in simulating artificial worms to solve it. Section 1.3.2 describes the studies in Chapters 3 and 4, which investigate optimal strategies for cellular decision-making using reinforcement learning under different environmental conditions. Finally, Section 1.3.3 shifts the focus away from machine learning and introduces how AD can be applied directly to optimise receptor locations (Chapter 5) or study adaptability in biological transport networks (Chapter 6) by calculating gradients through numerical solvers.

1.3.1 Data acquisition: Detection in microscopy images

Biology is fundamentally an observational science. It is through, ideally, well-designed experiments that we can observe how microorganisms interact with their surroundings. However, our conclusions cannot be based solely on anecdotal observations; that would be poor scientific rigour. Noticing that microorganisms reduce their motion after introducing a new drug in their environment is not enough to claim causality. Instead, it is essential that these changes are quantified, and to do so we need to perform data acquisition. Either by directly measuring physical properties with lab equipment or by performing post-processing on the collected data, these allow scientists to – hopefully rigorously – make informed conclusions about the systems at hand.

When studying microorganism behaviour, it has always been an effective way to explore phenotypic changes by examining how external conditions affect their *motility*, i.e., their ability to move independently. Indeed, motility is one of the most fundamental aspects on a microorganism existence [47–51]. Consequently, being able to accurately *track* microorganism trajectories in experimental recordings has been proven to be one of the most effective way to quantify these changes [52–55].

Before the widespread inclusion of computer-aided image processing, scientists relied heavily on manual tracking. Imagine spending hours at a microscope, watching microor-

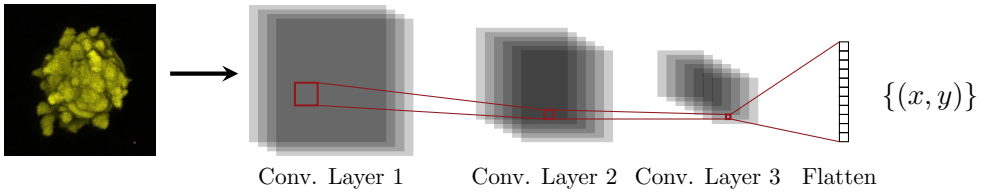


Figure 1.2: Schematic representation of a classical convolutional neural network (CNN) architecture. The input image of a bacterial colony (from Cordero et al. [47]) is processed through three convolutional layers, followed by a flattening operation, which transforms the spatial features into a vector representation, likely representing the desired quantity. Notice how at each change of layer, it is common to reduce the dimension of the image while increasing its channels, which promotes a more abstract representation of the input data. Highlighted connections illustrate the propagation of specific features through the network.

ganisms wriggle across the plate while annotating their movements by hand. This, of course, was – and in some places still is – an arduous task. Fortunately, the use of classical computer image-analysis techniques, such as thresholding and edge detection, greatly enhanced their ability to automate much of this work. This would drastically transform microscopy imaging and allow for faster and more accurate processing of visual information.

Segmentation methods capable of converting images into binary data enable to distinguish the organism from the background, which greatly facilitates the study of shape trajectories. While these methods are a ludicrous step forward, they are often inadequate when dealing with very noisy, low-resolution microscopy data or in scenarios where the steps to find the solutions cannot be easily defined. This inherit noise on experimental data, primarily caused by factors such as uneven illumination [56] or sensor limitations [45, 57], still presents a challenge to the image analysis community. For this very same reason, microscopy images have become the perfect candidate for computer vision tasks powered by deep learning (DL) methods.

CNNs are neural networks specifically designed for image processing that exploit translation invariance to reduce the amount of parameters and make them computationally efficient, while still being effective in recognising patterns and features within images [58] (see Figure 1.2 for a schematic of a standard CNN). It would be very challenging to define the classical algorithm capable of distinguishing different types of cells based on subtle differences in shapes or textures. Note that some classical approaches sometimes show performances comparable to DL methods, even in areas dominated by CNN, such as medical imaging [59]. It is, however, in more complex scenarios – where solutions cannot rely on predefined rules or hand-engineered features – that deep learning truly shines [45, 57].

1.3.1.1 Slender body tracking

In Chapter 2, I address the specific problem of tracking large ensembles of overlapping, slender bodies, such as swimming nematodes. This is sadly not a simple plug-and-play

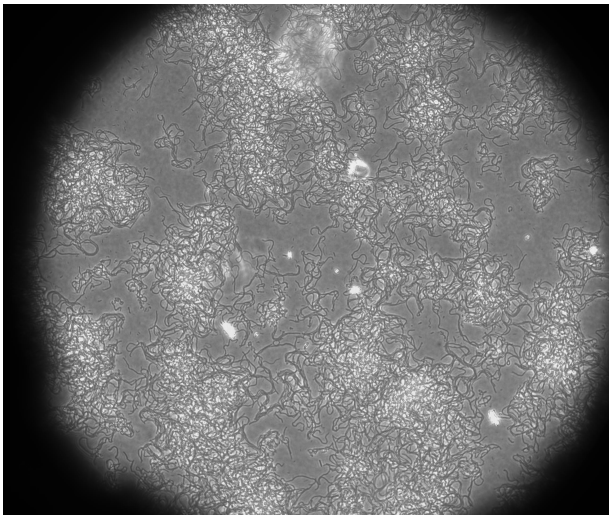
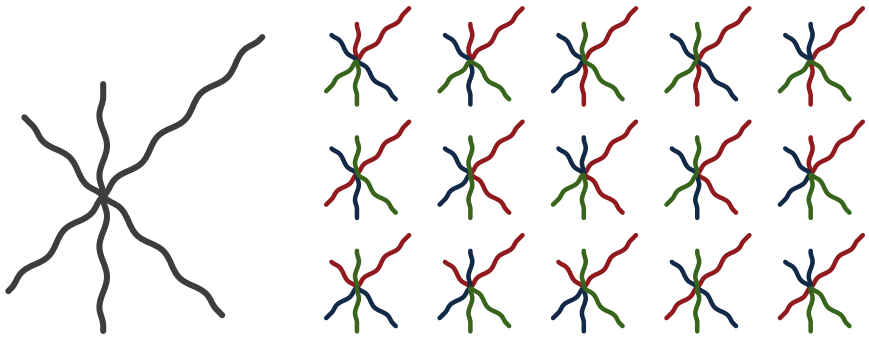


Figure 1.3: Microscopy image showing overlapping *Bacillus subtilis*, kindly provided by Mireia Cordero. This image highlights the challenge of tracking large ensembles of slender bodies that frequently cross paths and occlude each other. Such conditions make it difficult for deep learning models to differentiate between individual organisms, presenting significant obstacles in accurately tracking their motion and behaviour. Classical methods often struggle with scalability in such high-density recordings.

a problem for deep learning solutions. For example, when observing active organisms in a microscope, individual organisms often cross paths or become occluded, making it difficult for standard DL methods to differentiate between them (see Figure 1.3 for a nice visual example of how organisms tend to cross and overlap in microscopy data). DL models have struggled to detect and track swimming nematodes under these conditions, particularly due to the number of occlusions and overlaps that occur in large-scale microscopy recordings [53, 60]. Tracking many organisms that constantly cross each other is a significant challenge, and resolving these overlaps is a key roadblock in applying deep learning models to these kind of microscopy studies.

When high-resolution images are available and only a few organisms are being tracked simultaneously, DL-based solutions work rather well [61, 62]. For instance, pixel-wise segmentation models can often accurately segment individual cells or organisms when the data is clear and some occlusion occurs. But these approaches largely ignore the problem of overlapping, which is much more common in high-density recordings. In these cases, classical methods remain the *de facto* approach, as they have historically provided more reliable results for shape estimations [63, 64].

Classical methods often rely on probabilistic approaches, such as Kalman filters or combinatorial optimisation, to resolve the identities of overlapping organisms. These approaches attempt to assign each visible body part to a specific organism based on motion patterns, size, and trajectory, but their effectiveness diminishes as more organisms overlap. For example, resolving the identities of two crossing nematodes might be feasible with classical algorithms, but once three or more organisms are involved, the complexity increases drastically and the algorithm may fail to assign identities correctly (see Fig-



(a) (b)

Figure 1.4: Identity resolution in crossings can rarely be evaluated based on the current configuration alone due to the many possible solutions scenario, as the one shown in (*a*). For a simple crossing of three simultaneous organisms, and assuming the quality of the image is good enough to accurately extract their *skeleton*, identity assignments are not trivial. As shown in (*b*), a classical identity assignment method would need to assess all $N = \binom{6}{2}$ combinations using models or likelihoods before proceeding to the next timestep (or frame). This scales poorly the more simultaneous crossing events there are, as well as the more organisms are involved in a crossing.

ure 1.4 for an example of the combinatorial nature of identity assignment). Simply put, as the number of organisms in a sample increases, these classical approaches struggle with scalability and lead to slower processing times and reduced accuracy.

On the other hand, deep learning methods, while offering high throughput and the potential to process large datasets more efficiently, have not yet been able to consistently resolve the issue of overlapping organisms, or at least in the massive scale aimed here. In typical deep learning detection models, the system generates many potential detections (more than target objects in the image), each accompanied by a score that reflects the confidence level in that specific detection. It then eliminates those with lower confidence by assessing the degree of overlap in the predicted areas, a procedure known as non-maximum suppression [58], which results in a single detection per object. This, of course, does not translate well when objects are highly overlapped. Some models have been successfully able to overcome the limitations of occlusion in few organisms [65–67], mainly relying on rigid structures of the cell body for extrapolation. Fancier methods, using newer techniques, are capable of distinguishing trickier shapes but result in slower execution times and poor scalability [68].

Additionally, the lack of extensive labelled datasets for large-scale microscopy images, particularly those with low resolution and many overlapping and crossing organisms, makes training deep learning models for these applications even more challenging.

My work, as described in Chapter 2, addresses these two challenges – the handling of overlapping detections and the lack of a labelled dataset – and provides a deep learning-based solution for tracking many slender organisms in microscopy images. To resolve the identity assignment issue on overlapping organisms, a phenotypic encoding technique is used, which encodes the characteristic motions of the detected organisms and uses that

to perform the non-maximum suppression. This allows for accurately assigned correct identities to the detections suggestions of the model, which eases detecting through crossings and obfuscations.

To solve the lack of proper dataset, a synthetic one with simulated worms was curated. This approach results in perfect labels (as they are used to create the input), and freedom to generate situations that would be highly improbable to be obtained in *in vitro* systems or existing datasets. Of course, another alterative could involve manually labelling hard-to-resolve images, but persuading a bachelor student into it would probably result in some awkward pitching that was best avoided.

One key aspect of the modelling of swimming nematodes that the manuscript may not emphasise enough is accounting for the physics of motions at such small scales [11]. Specifically, this involves modelling a slender object navigating in a viscous media. At these scales, the Reynold number (Re) – a dimensionless metric indicating the ratio of inertial to viscous forces – is quite small, i.e., $Re \ll 1$. In such environments, where inertia can be neglected, the fluid dynamics given by the Navier-Stokes equations simplify and reduce to the linear form

$$\mu \nabla^2 \boldsymbol{v} = \nabla p \tag{1.5}$$

$$\nabla \cdot \boldsymbol{v} = 0 \tag{1.6}$$

where p is the pressure, μ the viscosity of the fluid and $\boldsymbol{v}$ the velocity field. This has strong consequences on the motility of organisms, as nicely put by Purcell [11]. Regardless of this notably simplification, solving the Stokes equations for complex shapes such as the ones in slender bodies is still computationally cumbersome. Thankfully, the study of how slender objects interact with the surrounding fluid is and has been of great scientific interest. As a result, Gray and Hancock [69] developed an approximation theory known as Resistive Force Theory (RTF), which calculates the expected motion due to the oscillations of the slender body. RTF estimates hydrodynamic forces by focussing on the local forces acting along the length of the body. This effectively simplifies the problem by decoupling the body's segments, neglecting long-range interactions because of viscous damping. Although this approximation reduces computational complexity, it remains accurate enough for many practical applications.

The force per unit on a segment of the slender body is given by the linear relation

$$\frac{\mathrm{d}\boldsymbol{f}(s)}{\mathrm{d}s} = -\xi_{\perp} \boldsymbol{u}_{\perp}(s) + \xi_{\parallel} \boldsymbol{u}_{\parallel}, \tag{1.7}$$

where the contributions are split between perpendicular ($\perp$) and parallel ($\parallel$) components, with ξ and $\boldsymbol{u}$ being the resistive force coefficient and the local velocity relative to the segment, respectively.

Having such a simplified description of the forces applied to each segment allows the determination of the net hydrodynamic force $\boldsymbol{F}$ and torque $\boldsymbol{\tau}$ acting on the body. By integrating over the length of the body, we solve for the motion under the condition of

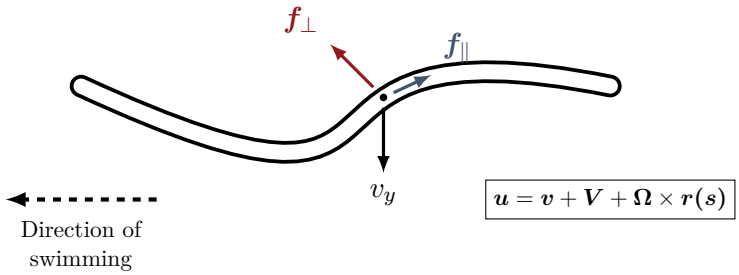


Figure 1.5: Application of resistive force theory (RFT) to a slender body, such as a nematode. The body is depicted as a curve, with local forces decomposed into perpendicular (red arrow, $\boldsymbol{f}_\perp$) and parallel (blue arrow, $\boldsymbol{f}_\parallel$) components at specific points along its length. The forces are calculated using Eq. (1.7), which relates the force per unit length to the resistive force coefficients $\xi_\perp$ and $\xi_\parallel$ and the corresponding local velocities $\boldsymbol{u}_\perp$ and $\boldsymbol{u}_\parallel$. The local velocity of a segment is given by the shown equation, where $\boldsymbol{v}(s)$ is the velocity due to body oscillation, $\boldsymbol{V}$ is the translational velocity of the centre of mass, and $\boldsymbol{\Omega}$ is the angular velocity about the centre of mass, with $\boldsymbol{r}(s)$ being the position of the segment relative to the centre. This formulation accounts for the combined contributions of body oscillations, translational motion, and rotation in determining the local hydrodynamic forces.

net-zero force and torque, such that:

$$\boldsymbol{F} = \int_0^L \boldsymbol{f}(s)\, \mathrm{d}s = 0, \quad \boldsymbol{\tau} = \int_0^L \boldsymbol{r}(s) \times \boldsymbol{f}(s)\, \mathrm{d}s = 0. \tag{1.8}$$

where $\boldsymbol{r}(s)$ is the position vector along the body.

This simple formulation of such a complicated motion allowed for the simulation of millions of swimming *Caenorhabditis elegans*, which led to the creation of a large and physically consistent data set of synthetic worms as ground-truth labels. After creating these labels, I performed some post-processing to generate realistic microscopy images by simulating optical effects, hoping that they closely resembled experimental data. Then these synthetic data were used to train a deep learning convolutional model that tracks real swimming *C. elegans*, even in challenging scenarios with overlapping bodies.

Although not thoroughly validated, this impressive generalisation to *in vitro* samples highlights the effectiveness of using physical principles to train a DL model and shows as a nice example of the capability of RFT in describing the real movements of slender bodies. Similarly, Costa et al. [70] nicely showed the usage of RFT to study the dynamics of *C. elegans* as Markovian processes.

Chapter 2 shows how by employing these two – albeit simple in hindsight – solutions, a tracking algorithm is defined that provides a bast improvement to current methodologies and results in an embarrassing parallel detection and tracking approach ideal for quantitative microscopy analysis.

1.3.2 Evolution as a non-differentiable optimisation process

A common research area in the biophysical sciences is the study of evolutionary optimisation. Evolutionary processes often lead to optimised biological systems, where

organisms adapt, through the course of generations, their structures and behaviours for survival and reproduction [50]. This natural optimisation process has long fascinated researchers because it provides a window into how biological systems might converge on efficient, adaptive solutions over time, the utility of which ranges from biological insight to human-designed applications [71, 72]. These solutions tend to reflect optimal configurations that maximise the fitness of the organism in its environment [73]. This makes optimisation problems incredibly relevant on systems biology – and matches perfectly with optimisation mechanisms such as those that AD makes possible.

One of the most interesting aspects of evolutionary process revolves around the fact that the *fitness* function landscape needs not to be differentiable. In fact, fitness landscapes are often rugged and irregular, filled with local optima that make it difficult to predict the best minimisation (or maximisation) path [74]. For example, in a fitness landscape with multiple peaks, a population of organisms may evolve to settle on a local peak, which represents an optimal configuration for that environment, but not necessarily the global peak, which would be the absolute best possible configuration. This is *not uncommon* on the topic of stochastic optimisation, now extensively studied in the field of machine learning, but where many techniques have been explored to deal with such degenerate systems [75–77].

The exact shape of this fitness landscape is usually unclear and the optimal configuration of the systems is not *a priori* known. Thus, when optimising such systems, stochastic exploration – drawing inspiration of evolutionary processes [78] – is required.

The concept of evolution tightly couples with that on a physicist view of the world, where the principle of energy minimisation – all objects tend to the minimal energy configuration in equilibrium – is one of the fundamental laws of the universe. Systems naturally evolve toward states that minimise their free energy, whether it is the formation of stable molecules, the growth of crystals, or the motion of planets. In much the same way, biological systems can be thought of as minimising some form of "biological energy" or maximising fitness to survive in a given environment.

One powerful tool for solving optimisation problems in complex stochastic systems is reinforcement learning (RL) [79], a branch of unsupervised machine learning that, despite its many pitfalls [80], has shown incredible performance in the optimisation of difficult tasks [81–83]. Particularly useful for problems where the objective function is not differentiable or where the solution requires balancing immediate and long-term rewards.

Reinforcement learning differs from traditional optimisation techniques, such as gradient descent, in several important ways. In a standard optimisation problem, you often have a clear objective function (such as minimising error) and a well-defined method for calculating gradients, which allows the system to approach an optimal solution gradually. However, RL operates in situations where the optimal solution may not be obvious from immediate feedback alone (similar to many strategy-based scenarios). Instead, it optimises what is known as a policy – a function that guides the system's reaction – considering both immediate and future rewards. Simply, RL works by training an agent to make decisions through an iterative process of trial and error (see Figure 1.6 for the

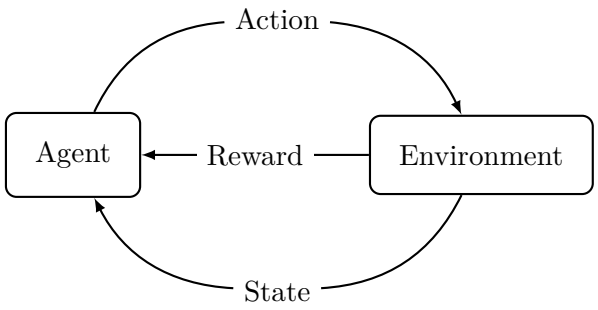


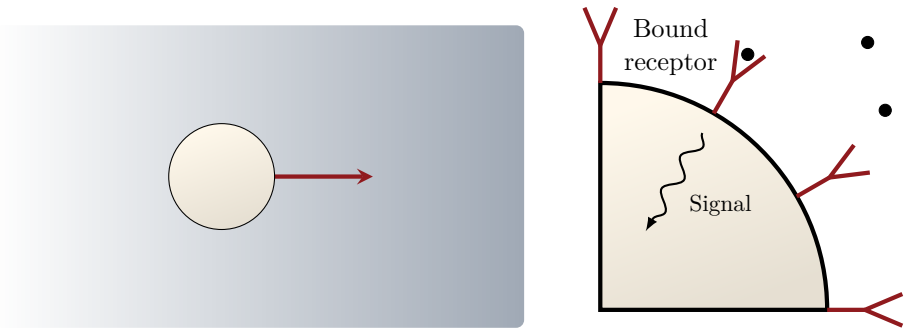
Figure 1.6: Diagram illustrating the fundamental mechanisms of reinforcement learning. The interaction between the agent and the environment is highlighted through a cyclical process. The agent selects an action based on its policy and sends it to the environment. In response, the environment provides feedback in the form of a new state and a reward, which the agent uses to learn and adjust its policy. The flow emphasizes the critical components: action, state, and reward, showcasing how the agent iteratively improves its decision-making strategy through trial and error, guided by reward signals.

classical representation of the RL workflow). The agent receives feedback in the form of rewards and uses these to adjust its actions to increase the cumulative future reward, thus optimising the policy [79]. This long-term reward approach should sound rather similar to the way evolution is capable of optimising real biological systems [78]. Biting a sleeping lion might provide an instant reward to solve hunger, but it is unlikely that such risky behaviour contributed to the survival and propagation of the biter's genes [74]. Hence, the reason for exploring the use of RL techniques to study such processes is seldom explained.

A similar class of optimisation techniques is that in the evolutionary algorithm category [84]. Albeit very similar to RL – and extremely similar to natural selection [78] – those optimisation methods are more compute intensive and lack the smooth convergence of gradient descent. Although not used here, they are also great tools for studying systems such as those presented in this thesis [85, 86]. Interestingly, there exist hybrid approaches that combine evolutionary algorithms and reinforcement learning to address their respective limitations. These techniques, known as Evolutionary Reinforcement Learning, offer an interesting avenue for studies on biological optimisation [87].

There is nothing relating automatic differentiation to reinforcement learning *per se*. Precisely, the use of RL is needed due to the non-differentiability of the loss (or reward) landscape, which is counter intuitive to the advantages of AD. However, the decision-making mechanism can be defined to allow gradient descent in its parameters. In fact, DeepMind demonstrated the capabilities of reinforcement learning (RL) combined with deep learning – deep reinforcement learning (DRL) – when Mnih et al. [82] successfully trained an agent to play Atari games. This breakthrough showed that DRL could solve drastically more complex tasks and led to far more impactful models [88–90]. This makes DRL a great tool for studying the complex internal mechanisms that cells may have evolved and optimised over time, in a rather simplified manner.

However, once deep models – neural networks with more than a couple of layers – are involved, interpretability often becomes a distant hope. Fortunately, there are



(a) (b)

Figure 1.7: (a) A cell perform chemotaxis by moving up (or down – but in this case up) the gradient, here indicated by the coloured background. (b) Ligand receptors (red) on the surface of the cell (black) bind to single particles (dot), that diffuse by Brownian motion on the environment. For the non-absorbing case, this particle would unbind from the receptor after a set binding time, and will return to the diffusive environment, with *some* probability of rebinding to another (or to the same) receptor. The binding of the receptor sends a discrete signal to the cell. But, since the cell tends to have thousands of receptors, these discrete signals provide the cell with gradient information.

techniques for understanding these models, such as the *Integrated Gradients* method proposed by Sundararajan et al. [91], which extracts information on how input values influence the resulting output.

1.3.2.1 Chemotaxis

In this thesis, I study one example of such an evolutionary process; the ability of an organism to navigate their environment as a response to chemical cues, i.e., perform chemotaxis (Fig. 1.7a). Chemotaxis is a fundamental biological process in which a cell detects chemical signals in its environment and moves either toward or away from them, depending on whether they are beneficial or harmful [92–94]. The ability to navigate chemical gradients is clearly essential for survival in many microorganisms [50]. As in any competitive environment, quickly getting to the food source (or away of the *poison*), is crucial to increase your change of survival over the other organisms. Bacteria, such as the overstudied *Escherichia coli*, swim toward higher concentrations of nutrients by detecting astonishingly small changes in chemical concentrations through receptors on their surface [51, 95]. These receptors initiate signalling cascades inside the cell, guiding it to adjust its motion accordingly [96].

Estimating gradients is one of those things that seems a quite straightforward strategy for us on the macroscale. You can easily determine the direction of the sun by feeling the warmth on your skin (as long as you are outdoors and exposed to sunlight[3]). But, as with the case of motion and its lack of inertia, sensing is not so intuitive when it is performed on the microscale. Due to their small size, cells live in what is considered

[3]This may be a bad example to make during Danish winter.

fundamental limits of sensing [10, 13, 97, 98], where physical constraints significantly influence their ability to interact with the environment.

These limits mostly appear due to the stochastic nature of molecular processes (Fig. 1.7b). At small scales, where thermal noise is ever dominant, molecules diffuse randomly, and the number of molecules that receptors can bind is inherently limited by the cell size. In fact, according to the work of Berg and Purcell [10], the mean number of binding events for a cell during a measuring time T can be approximated with

$$N \approx 4\pi DcaT, \tag{1.9}$$

where D is the diffusion constant of the molecules, c is the chemoatractant concentration at the cell location and a is the cell size. For smaller cells, this makes it impossible to detect concentration differences across their surface and requires them to seek alternative strategies to perform chemotaxis [99].

Due to the binding kinetics of chemoreceptors, the constant random fluctuations of surrounding molecules significantly affect the ability of a microorganism to distinguish meaningful signals from noise. This noise causes the binding events to follow a Poisson distribution, introducing variability in the number of detected molecules that scale as $\delta N \propto \sqrt{N}$. As a result, the signal-to-noise ratio (SNR) – often characterised by the steepness of the gradient relative to the background concentration – becomes an important factor in studying sensory precision [13, 100]. Additionally, the energy costs of processing information and maintaining the molecular machinery required for sensing further constrain a cell's ability to effectively interact with its surroundings [101, 102]. The counterintuitive nature of these constraints when making decisions highlights the merit of physics-based studies in exploring their navigation strategies.

Studying the optimality of such noisy and unintuitive systems is not an easy task – and its formulation is far from trivial. Most concerning is the fact that the fitness landscape of navigating strategies is rarely well defined and is likely to be riddled with many suboptimal solutions [103]. This is where modern computational techniques, such as AD and RL, come into play.

In the context of cellular chemotaxis, RL can be used to simulate how a cell learns to navigate chemical gradients by adjusting its movement over time. As simple as it sounds, the task becomes challenging when accounting for the physical limits of sensing [104]. For example, a cell might initially make random movements, but, over time, learn that certain reactions may ultimately lead to higher concentrations of nutrients [86, 105]. This very much relates to the experience over evolutionary time-scales, as an optimal decision-making policy improves the cell's chances of survival [50]. Therefore, identifying in which observational patterns organisms exploit while learning to navigate the gradient can yield critical insight into their internal mechanisms.

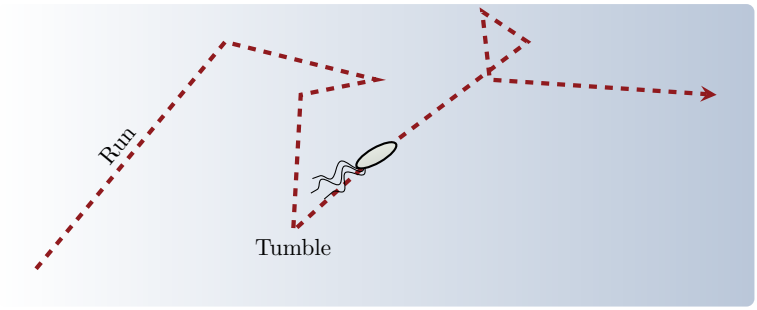
1.3.2.2 Sensing strategies

An interesting consequence of such physical limits is the apparent dichotomy[4] in chemotaxis strategies performed by microorganisms (see Figure 1.8 for visual examples of these). It is well known that small, fast organisms – such as bacteria – rely on temporal comparisons to evaluate whether to correct their direction of motion [92]. On the other hand, large and slow organisms take advantage of their spatial advantage to estimate the gradient by comparing measures along their bodies [94]. The clear observational difference between these two strategies is of evident interest for any physicist. In fact, Wan and Jékely [7] captures this distinction by showing that when displaying the size versus speed of the organism on a log-log diagram, a distinct boundary between those relying on temporal comparisons versus spatial sensing appears. Their analysis highlights how navigation strategies cluster into distinct regions and, interestingly, divide prokaryotic and eukaryotic behaviours. But this is not a new concept, Tan and Chiam [106] and Dusenbery [107] compare the performance of both sensing strategies under different conditions and similarly to Rode et al. [108] and Metzner [109], observe that their effectivity is determined by a set of environmental and morphological parameters. While very insightful, these studies assume prior knowledge of both strategies, leaving open the question of whether these strategies optimally exploit the organism's sensing limitations. It was already shown by Berg and Purcell [10] in their remarkable analysis of chemoreception that the relative noise in sensing is inversely proportional to the measurement time, which agrees with Endres and Wingreen [13] observation of the accuracy of gradient estimation at the limits of sensing. Therefore, even for a direct spatial sensing mechanism, it is clearly beneficial to use memory to average the noise in the measurements. This shows that the transition between temporal and spatial is already blurred. However, temporal strategies do not rely on temporal measurements to average values, but instead rely on memory to perform temporal comparisons [110]. This raises important questions: When and how does the optimal use of memory transition – from comparison to averaging, and ultimately to being redundant? After all, it is within the transition regime where the most fascinating phenomena tend to occur [111].

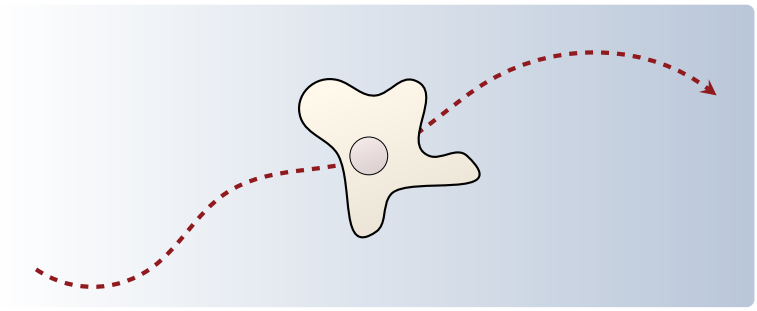
Since these limits control when the transition occurs, the change in strategies can be observed by either modifying the environment or adjusting the cell properties [108, 109]. In the work presented here, this is approached by smoothly increasing cell size, and using DRL to estimate the best performant policy that leverages both memory and instant sensing information. Notably, this is done without constraining the internal processing of the cell – as it is modelled as a small artificial neural network capable of approximating any function – but by only limiting the type of information it has access to.

Consider a self-propelled active particle that can *choose* the change in orientation at each timestep based on the detected information. This is very similar to Colabrese et al. [83] problem setup but where the effects of the surrounding fluid are neglected. The reorientation of the cell is set by its angular velocity $\omega(t)$, which describes the rate

[4] Rarely has there been the situation where I've liked one of Julius pretentious words. This is one of those rare cases.



(a)



(b)

Figure 1.8: The diagrams here depict the two primary strategies for cellular chemotaxis: (*a*) Temporal comparison, used mainly by bacteria, involves a *run-and-tumble* approach, where bacteria move in a straight line (*run*) and then *tumble* randomly, adjusting their direction based on chemical gradients detected over time. For the sake of simplicity, I ignore here the twitching mechanism. (*b*) Spatial gradient sensing, used by larger, mostly eukaryotic cells, involves directly sensing spatial variations in chemical concentrations across the body of the cell, allowing for more directed (and slower) movement.

at which the orientation changes over time and can be described with the following Langevin dynamics:

$$I\frac{d\omega(t)}{dt} = -\gamma\omega(t) + \eta(t) + \tau_{\text{ext}}(t), \tag{1.10}$$

where I is the moment of inertia, τ_{ext} is the external torque and $\eta(t)$ is the fluctuations that create rotational diffusion. However, since the cell is a neutrally buoyant particle small enough for inertial effects to be ignored, i.e., low Reynolds numbers ($Re \ll 1$), the dynamics are described on the overdamped regime, which results in the change in orientation (or angular velocity) as

$$\gamma\omega(t) = \tau_{\text{ext}}(t) + \eta(t), \tag{1.11}$$

where $\omega = \frac{\mathrm{d}\theta}{\mathrm{d}t}$. Interestingly, the cell orientation is defined just by the active torque and its rotational diffusion (due to thermal or motility fluctuations). This has the implications that even for a *perfect navigating* cell capable of controlling $\tau_{\text{ext}}(t)$ and thus performing deterministic steering, rotational diffusion creates an upper limit to the length scale for valid memory of previous forces.

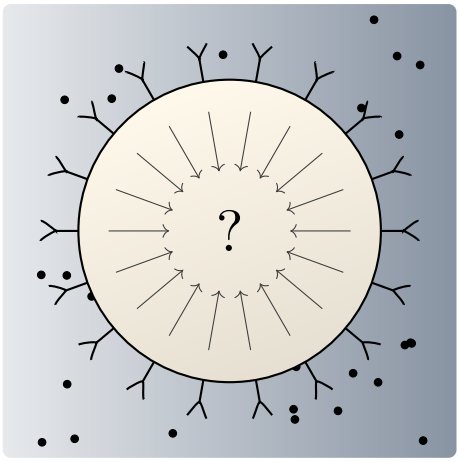


Figure 1.9: The cell decision-making process based on the sensors located on its surface. This process is not a mathematical one, but a biochemical one. And while in some cases as in *E. coli*, the signalling pathway and the transformation from binding to motor rotation is, on its majority, understood, that is not the case for more complex organisms. We have some understanding of the signalling networks, but some fundamental truths can be hidden behind so much complexity. For that, there is merit in oversimplifying the decision process.

Chapter 3 shows the smooth transitions between strategies and explores how the optimal integration of those two types of sensing would look like. At the limits of small and large cell sizes, the well-known strategies of temporal and spatial comparison naturally emerge as the optimal mechanisms for sensing. Remarkably, their emergence occurs without explicitly enforcing any prior knowledge into the model but arises from the fundamental limits themselves. These results validate the optimality of temporal and spatial sensing at their respective limits but also provide insight into how a cell can effectively integrate information from both sensings to achieve optimal decision-making.

1.3.2.3 Amoeboid mechanical intelligence

Having relied on artificial neural networks to obtain the optimal decision-making policy, albeit useful on its generality and unrestricted nature, it lacks the interpretability of how specific cells make such decisions[5]. For that, it is necessary to look to specific mechanisms that cells employ and understand the underlying processes that make a cell take a decision, e.g., choose the direction of movement. Therefore, Chapter 4 focusses on amoebas to explore decision-making in amoeboid chemotaxis.

Amoebas, such as *Dictyostelium discoideum*, create temporary protrusions in the cell membrane driven by the polymerisation of actin filaments, which play a central role during their navigation [48, 112, 113]. These protrusions, known as pseudopodia, grow only perpendicular to the cell membrane [114], and even though they are commonly seen as tools for its motility mechanisms (and they are), it is very likely that they play

[5]This was a quite strong concern for Referee B during the peer-review process on one of the submissions, despite not being the aim of the manuscript.

an important role in the cell's ability to sense its surroundings [115]. This raises the question of how pseudopods may enhance chemotaxis decision-making in environments with limited or noisy information?

Actin is one of the most abundant proteins in eukaryotic cells [116], and it is crucial to cell morphology [115], as it is the material of which the cytoskeleton is composed. The cytoskeleton is the backbone of the cell body and provides mechanical support for the membrane and internal mechanisms [117]. Interestingly, the cytoskeleton is composed of actin filaments (F-actin), which continuously get depolymerised and polymerised, as well as contract and expand, helped by myosin motor proteins, to allow the cell to move (and create protrusions) [114]. Polymerisation is regulated by signals from cell receptors, which bind to molecules of chemoattractant in the environment [115]. Thus, it is important to note that actin is the key component in the coupling of cell motility, cell morphology, and sensing, and any study of this coupling should focus on it.

In fact, work by Tweedy et al. [118] has already shown that the shape of the cell is closely coupled with environmental conditions, and notably that the morphology can be dictated by the signal-to-noise ratio of the surrounding chemical profile – comparable to the fundamental limits of accuracy [13]. Their observations agree with the reaction-diffusion approach proposed by Meinhardt [119], and similarly to Neilson et al. [120], suggest that the creation of new pseudopods happens as a reaction to diffusive signal from the membrane. On the other hand, Andrew and Insall [121] argues that pseudopod formation happens mainly by splitting from existing ones rather than being created on demand, and their direction is random. Chemotactic signalling, however, helps maintain the most accurate pseudopod which influences the frequency of pseudopods the gradient direction. Bosgraaf and Haastert [122] already hints at the importance of persistence in pseudopod formation for effective chemotaxis in shallow gradients. These findings emphasise the strong role pseudopods clearly play in amoeboid chemotaxis, and stresses the significance of understanding their dynamics. Based on the modelling of house-hunting honeybee swarms decision-making described by Pais et al. [123], Chapter 4 presents a model of the interaction between signal sensing and the growth of physical pseudopods reframed as a resource competition for the polymerisation of actin monomers (G-actin).

The model is hypothetical, but plausible, and describes the polymerisation contest as an out-of-equilibrium stochastic dynamical system. Therefore, polymerisation (converting G-actin into F-actin) at each pseudopod is governed by the interactions with the other pseudopods as well as the fluctuations in sensing signalling. The underlying system is then simply defined as a set of stochastic differential equations (SDEs) that allow the study of the role of pseudopods in spatial gradient sensing and directional motility. In its essence, our recruitment competition between pseudopods is a resource allocation problem, governed by the mass conservation principle. The direction of growth has an independent polymerisation rate determined by the change in the free energy of the receptors in their membrane section, which varies with the concentration of the chemoattractant. It is when noise and fluctuations in chemosensing that these competitions become interesting, and where we observe the emergence of phenomena

reported experimentally, such as the scaling of Weber-Fechner law and the advantage of persistence in static environments.

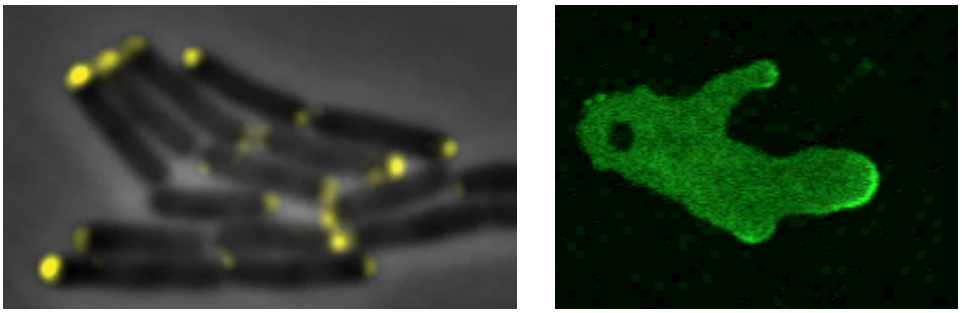
The model is then expanded to evaluate the optimal strategies that an organism would use when faced with different environmental conditions. Using DRL, it becomes clear that frontal polarisation of pseudopod directions is highly relevant in very noisy or shallow regimes, whereas in more information-dense environments, subtle nuances between directions can be exploited for *perfect* chemotaxis.

The beauty of this minimalistic system lies in its ability to capture the decision-making capabilities – or what we term *mechanical intelligence* – of amoeboid organisms, that organically emerge from biochemical and physical processes. This physics-inspired approach distils the complexity of actin polymerisation, which involves millions of filaments, into a model governed by merely six free parameters. The simplicity of the model not only makes it interpretable, but also allows for seamless integration with advanced techniques DRL. This integration enables the exploration of optimal configurations for more complex and less straightforward scenarios. Ultimately, what the project in Chapter 4 shows is an approach that yields both clarity and possible valuable insights into the intricate mechanisms driving amoeboid motility and decision-making processes.

1.3.3 Optimising with differentiable physics

The use of automatic differentiation extends beyond its well-known applications in deep learning, even though they remain its most prominent use case. The ability to compute exact gradients of arbitrary functions – without the approximation errors or cumbersome expressions associated with traditional methods – makes it an indispensable tool for optimising complex systems. Particularly interesting is when applied to physical models, an area loosely term *differentiable physics*. This approach involves propagating gradients through physical simulations during optimisation, enabling not only the integration of physical formulations into deep learning workflows but also the direct optimisation of multi-variable non-linear systems where numerical methods often struggle and result in a non-smooth, computationally expensive process [40]. Compared to other optimisation methods like reinforcement learning, differentiable physics offers certain advantages, such as eliminating the need for exhaustive exploration (and running millions of simulations). Thus, it is particularly valuable for computationally intensive applications such as fluid dynamics and robotics [33, 124, 125]. However, *it can't have the cake and eat it too*, a key limitation is that gradient descent, as a local optimisation method, usually gets stuck in local optimal solutions when dealing with non-convex landscapes. In such cases, supplementary techniques to approach global optima may be needed. Nevertheless, for problems where identifying local optima works, it provides an efficient and computationally cheap solution.

This is indeed the topic of the remaining chapters (Chapter 5 and 6), where many-component systems are optimised by using differentiable physics in order to explore optimal configurations.



(a) (b)

Figure 1.10: Receptor clustering formed at high-curvature regions observed in experimental studies. (*a*) Images from Koler et al. [130] where *E. coli* express mutant receptors. (*b*) *Dictyostelium discoideum* cell moving in a 2-ţm-deep chamber with a chemical gradient. The green fluorescent marker that reveals activated receptor proteins, clustered at the tips. Reproduced from Levine and Rappel [94], with the permission of the American Institute of Physics.

1.3.3.1 Receptor clustering

Keeping in topic with previous work, Chapter 5 studies the optimal position of cell-surface receptors for spatial gradient sensing. It is a common assumption when studying cell decision-making to assume that the information that arrives to the cell comes from all directions [10], and if receptors are discretely defined, they are usually spread uniformly along the surface of the cell body[6] [104]. After all, there is no cell front and back for a static cell. In reality, cells have shown to agree with this assumption [126], but also break such symmetry to form clusters of receptors due to polarisation [94, 113, 127, 128]. Note also that the assumption of uniformity simplifies the derivations due to all the mathematical advantages that such symmetries tend to provide [13, 129]. For the most famous case of static receptor localisation, we can look at *Escherichia coli*, where ligand receptor distribution is concentrated at the poles and forms arrays of receptors [130].

Nevertheless, if we assume that the cell mechanisms for performing signal processing before making a decision are analogous to a least-squares fitting, then it surely makes sense for cells to spread their receptors throughout their body. By doing so, it maximises the gradient information and should result in a uniform distribution. Similarly, any potential correlation between receptor measurements would encourage receptors to sample the concentration profile in a dispersed manner, minimising the likelihood of redundant molecular bindings. On the other hand, as just mentioned, experimental observations suggest that receptors do cluster. This has been shown to occur to minimise curvature energy between receptors and membrane, reduce biochemical potentials, or amplify signals. This slight paradoxical situation suggests that the receptor distribution may not be trivially defined and makes for a very interesting study.

An interesting phenomenon on systems of binding receptors is the possibility of re-

[6] A handy example of this is the model in Chapter 3 and the model proposed in the appendix section of Chapter 4

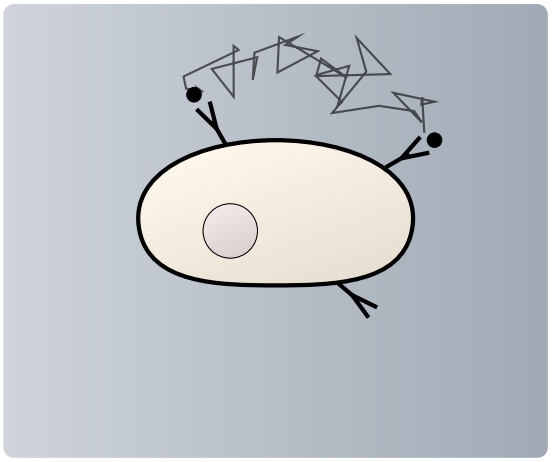


Figure 1.11: A molecule that binds to one receptor may diffuse to another receptor and bind again, generating a second signal within the cell's internal mechanisms. This rebinding effect creates a spatio-temporal correlation between binding events, influencing the accuracy of signal interpretation.

binding. The molecule binds to the receptor and, after a short period of time, is released again. Now, the probability of rebinding to the same receptor is very low (but not zero). More concerning is the chance that such molecule will rebind to another receptor (Fig. 1.11). In the non-absorbing case, this would result in the wrong counting of two signals but generated from the same single molecule. This is usually ignored; as in Berg and Purcell [10], they assume the bindings to be diffusion limited (a particle diffuses past the body faster than the expected measuring time). In contrast, this is clearly modelled in Bialek and Setayeshgar [97] with the use of the fluctuations dissipation theorem (FDT), where the kinematics of binding and unbinding are accounted for and add to the uncertainty in measurement. These dynamics have strong implications on the cell's ability to sense the gradient [100]. Similarly, Kaizu et al. [129] expanded on Berg and Purcell to account for rebinding, showing that only in the original limit is indeed a lower bound, given the diffusion-limited assumption.

In the work described in Chapter 5, the binding dynamics are ignored. Instead, the focus is on quantifying the correlation between receptor measurements. To do so, the probability that a particle can be found at both receptors during the measurement time is derived. This yields the correlation of the average measurement during an integration time, and it is used to compute the lower bound on the estimate uncertainty. This formulation, similar to that of Berg and Purcell [10], enables the analysis of the error in gradient estimation using statistical information theory and its dependence on receptor locations. To determine the optimal configuration of cell-surface receptors that minimises uncertainty in estimation, differentiable programming is used to compute gradients through the solving process. This is one of those cases where AD excels, as the computational graph of such a process is intricate and very sensitive to perturbations.

Interestingly, we find that on a perfect spherical surface, there is no need to redistribute the receptors from their uniform distribution. However, when that surface is

perturbed, receptors move towards the tips of the new protrusions, which considerably reduces the estimation uncertainty as those grow. Such a simple model shows that when receptor distribution is accounted for, quantification in uncertainty reduction due to shape changes is greatly under-reported. Not only do we quantify the relevance of receptors' locations, but we also show that they agree with the mechanical explanations previously reported. Meaning that there is a strong coupling between evolutionary advantage and physical mechanisms of receptor clustering. Notably, here, we ignore all the other reasons why receptors would like to cluster, and yet clustering emerges.

1.3.3.2 Biological Transport Networks

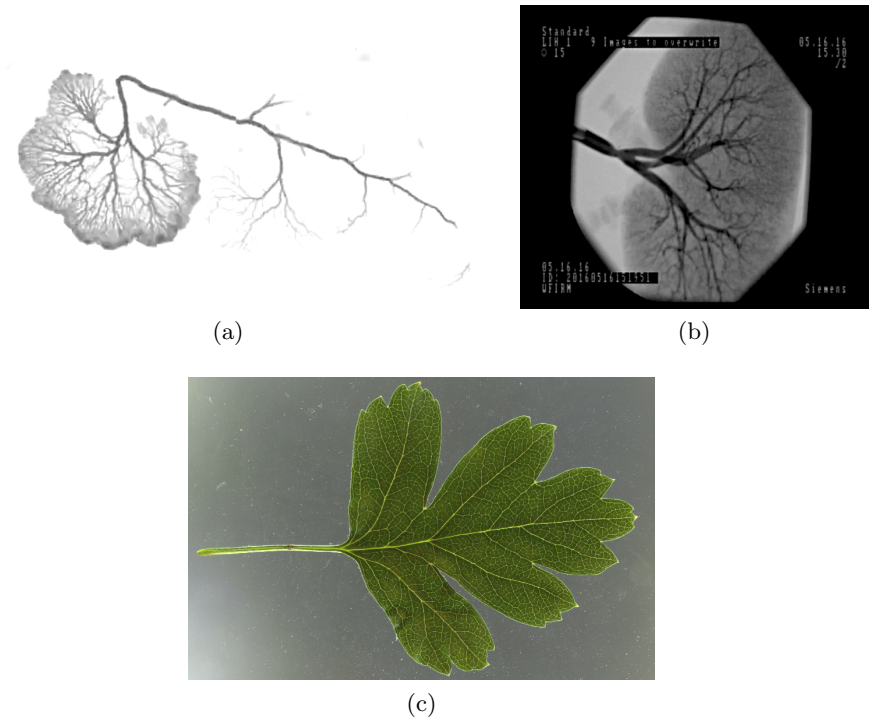
Until now, this thesis has focused on single-cell behaviour, particularly in the context of sensing and motility. These studies reveal how microorganisms dynamically change their bodies and resource allocation strategies to optimise their interactions with the environment. Similar principles have been observed to govern the morphologies of collective behaviour [131], leading to unexpected (and visually intriguing) ordered structures in their optimal configuration [132]. Some fungi and slime moulds dynamically reorganise their transport structures during foraging [133], as is the case in *Physarum polycephalum* where a trade-off between variability and resource transport efficiency dictates the foraging strategy [134] (see Figure 1.12a). Similarly, Wilking et al. [135] reports biofilms of *Bacillus subtilis* creating channel networks to overcome diffusion limits, achieving efficient transport of nutrients and waste. These results are not limited to living organisms, as efficient transport of resources is also critical to organ functionality [136] and plants [137] (Figure 1.12b exemplifies one of such systems with the vein network in a porcine kidney). These networks, much like the previous works, show how fundamental physical constraints drive biological optimisation across scales.

In the final research project of this thesis (Chapter 6), I explore the optimisation of such biological transport networks (BTNs) from a hydrodynamic perspective and show how differentiable programming can be used to study their adaptability capabilities. Albeit general, the results are strongly exemplified on the venation systems of leaves (Fig. 1.12c) as they are commonly used as the model system for such generic studies [140].

The physics behind BTNs is rather interesting, as the system needs to find a way to efficiently deliver its resources under the constraint of fundamental physical laws. This is often coupled with environmental forces and evolutionary needs [73], and results in optimal configurations from which humans can gain insight [136] as well as inspiration [141, 142]. Intrinsic to these structures is the concept of energy, as these optimal solutions need to complete the task (deliver) using the less amount of resources (energy), and thus, BTNs are heavily governed by biophysical principles. For instance, Poisseuille's well-known equation of flow of liquids in tubes, described the flow rate F in cylindrical tubes as

$$F = \frac{\pi r^4}{8\eta L}\Delta p, \tag{1.12}$$

where the pressure difference Δp and the tube dimensions (L, r) and η is the viscosity,



(a) (b)

(c)

Figure 1.12: Examples of biological transport networks (BTN) in real systems. (*a*) Network morphology during the migration of *P. polycephalum* obtained from Schick et al. [134]. (*b*) Angiographic analysis of the decellularized porcine kidney showcasing the preserved vascular network, from Corridon [138]. (*c*) The venation pattern of a *Crataegus monogyna* leaf provided by Julius, which I presume it is from Skjegstad and Kirkegaard [139].

shows the importance of tube dimensions on transport networks. Similarly, Murray [143] establishes the optimal relationship between the radius of the parent and the radii of the daughter branches in bifurcations such that

$$r_p^3 = \sum_i r_{d_i}^3. \tag{1.13}$$

which became known as Murray's law and has been observed to be fulfilled in nature [137, 139, 144]. Bohn and Magnasco [145] showcase how biological systems tend to provide energy-efficient solutions that minimise transport costs using a hydrodynamic formulation. These often result in sparsely connected networks that resemble those seen in Figure 1.12. And while Hu and Cai [146] demonstrated that local feedback formulations could reproduce those, such systems are highly degenerate and struggle to identify global optimal solutions. An approach by [147] showed that by modelling growth, it leads to better and more natural looking solutions.

Despite successfully describing robustness in BTNs [148, 149], challenges remain in fully replicating the adaptability observed in biological systems. Notably, there is a

gap between the characteristics of *in vivo* biological networks and the results of purely hydrodynamic optimisation methods.

Chapter 6 addresses one such situation, specifically when a finite regular grid is constrained by an irregular boundary. This introduces additional challenges in network optimisation that are not fully captured by existing hydrodynamic models, offering an opportunity to explore new formulations that bridge the gap between theory and biological reality.

Consider a system composed of a finite number of nodes, inside a bounded domain, and study how the positioning of the nodes is affected by the boundary. Each node represents a sink node, where there is some resource to be delivered. However, current formulations expect the nodes to be equidistant and evenly spaced, which simplifies the mathematical formulations. To address it, the hydrodynamics model is reformulated to account for the cost to delivery to a region – as regions would no longer be uniform across the domain. Thus, the energy dissipation, P, of the system is then defined as the sum of two components:

$$P = P_{\text{transport}} + P_{\text{delivery}}, \tag{1.14}$$

where $P_{\text{transport}}$ represents the energy cost for transporting resources, and P_{delivery} accounts for the delivery cost to specific regions.

This formulation allows for the study of energy-efficient configurations while incorporating the impact of irregular boundaries and non-uniform delivery regions. Importantly, this defines the formulation to allow the nodes to no longer be constrained on a static grid. We use AD to calculate the gradients through the transport optimisation, where the optimal thickness of the edges is calculated, and then perform movement of the nodes.

Optimised networks based on this energy formulation show higher adaptability and efficient resource transport. By using AD to move the nodes to their optimal position, the resulting networks effectively adapt to the mismatch between the network lattice and the domain boundary. This results in configurations where the nodes are no longer evenly spaced but instead are positioned to account for the shape of the domain and the direction of the flow, yielding vein patterns that mimic better those found in actual biological systems.

Chapter 2

On tracking many worms

This chapter contains the following paper:

Fast detection of slender bodies in high-density microscopy data

Albert Alonso and Julius B. Kirkegaard

Niels Bohr Institute, University of Copenhagen, Copenhagen 2100, Denmark

Published in: *Nature Communications Biology*
DOI: https://doi.org/10.1038/s42003-023-05098-1
Pre-print server: https://arxiv.org/abs/2301.04460

Author Contributions:
Methodology, Validation, Investigation, Data Management, Visualisation, Formal Analysis, Software, Writing Original Draft, Writing Review & Editing. Apparently that's not much – at least according to that article the Novo Foundation did about Julius...

Manuscript reformatted to fit the style of this thesis.

Abstract

Computer-aided analysis of biological microscopy data has seen a massive improvement with the utilization of general-purpose deep learning techniques. Yet, in microscopy studies of multi-organism systems, the problem of collision and overlap remains challenging. This is particularly true for systems composed of slender bodies such as swimming nematodes, swimming spermatozoa, or the beating of eukaryotic or prokaryotic flagella. Here, we develop a end-to-end deep learning approach to extract precise shape trajectories of generally motile and overlapping slender bodies. Our method works in low resolution settings where feature keypoints are hard to define and detect. Detection is fast and we demonstrate the ability to track thousands of overlapping organisms simultaneously. While our approach is agnostic to area of application, we present it in the setting of and exemplify its usability on dense experiments of swimming *Caenorhabditis elegans*. The model training is achieved purely on synthetic data, utilizing a physics-based model for nematode motility, and we demonstrate the model's ability to generalize from simulations to experimental videos.

2.1 Introduction

Large-scale, high-throughput quantification of microscopy data has increasingly become possible with the aid of computer vision [150–155]. In particular within the last decade, deep learning techniques [42, 46, 57] have improved and enabled accurate image analysis of microscopy data in a broad range of areas including cell counting [156, 157], cell segmentation [158–160], nucleus detection [155, 161], sub-cellular segmentation [162], drug discovery [163], cancer detection [164–166], and the identification of infectious diseases [167, 168]. Detection models serve as the fundamental operation in tracking procedures, and combined with suitable tracking algorithms, these can achieve morphologically resolved organism tracks that can accurately quantify organism motility [169], the application of which ranges from fundamental neuroscience [170–172] and the circuitry of simple organisms [48, 173–175] to drug discovery [176–180].

Multi-organism detection can be achieved at increasing levels of fidelity: at the crudest, only center-of-mass locations or bounding boxes are predicted [181] which does enable tracking of organisms but provide little morphological information. In contrast, pixel-wise segmentation models [158] and pose estimation using keypoints [62] reveal accurate shape dynamics when employed on high-resolution data. However, these methods rely on high definition objects, as segmentation and prediction are highly sensitive to noise. In particular for organisms that are long and slender, pixel-wise segmentation fails at low resolution as correct predictions require sub-pixel accuracy. Moreover, at high densities, these methods may fail due to their inability to properly handle overlap between organisms.

Here, we consider the problem of studying slender organisms at low resolution and high density with the goal to enable both accurate identity tracking and quantification of shape dynamics. This problem has traditionally been approached by employing

pixel-wise segmentation and subsequent skeletonization procedures [52–54, 182–184], an approach that requires model-based approaches [60, 185] or ad-hoc procedures [63] to solve the problem of correctly identifying overlapping organisms, the combinatorial complexity of which blows up at high densities. To this end we abandon pixel-wise output and instead construct a neural network architecture that predicts, potentially overlapping, centerlines directly [186–188]. Our method enables both accurate shape prediction and tracking in dense experiments of slender objects, a key challenge for a broad class of systems [Fig. 2.1], including tracking of nematode worms [189–191], spiral or elongated bacteria [192–195], spermatozoa [196, 197], the flagella of both eukaryotes [183, 184] and prokaryotes [198], and freely swimming flagella such those of microgametes [199].

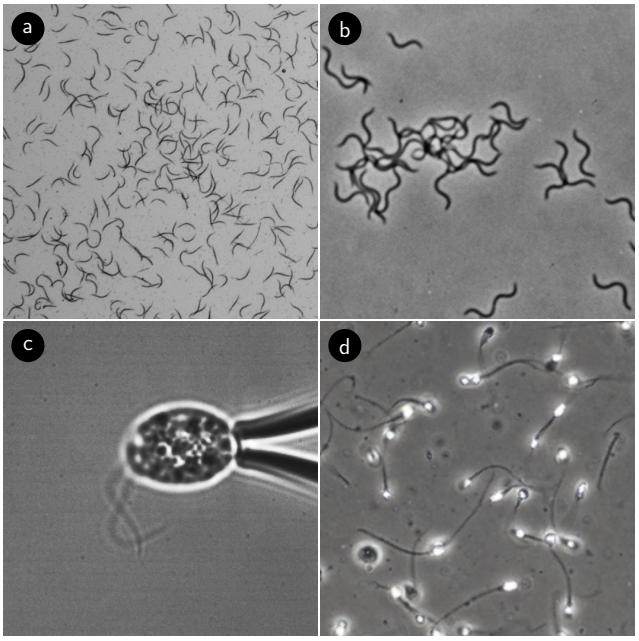


Figure 2.1: **Microscopy images of different microorganisms whose slender structure and frequent overlaps makes them hard to detect using classical approaches. a.** *C. elegans* motility experiment from the dataset of this paper. **b.** Motile, flexuous, thin, spiral-shaped *B. pilosicoli* bacteria. Still from Ref. [194], with permission. **c.** Beating flagella of the green alga *C. reinhardtii*, provided by Kirsty Wan, University of Exeter. **d.** Swimming human *spermatozoa*. From dataset in Ref. [197].

Our method relies on recent advances in deep learning [24, 200–203] and extends these by a few simple ideas: In still micrographs, the identities of individual worms can end up being obscured by overlaps making them impossible to accurately identify, and only by relying on the adjacent frames can they be correctly resolved. Thus, to allow the neural network to encode the identity of individual bodies as a function of their motion, the input to our neural network is taken to be short video clips rather than single frames. Our network outputs multiple independent predictions, and for each produces (1) the centreline of the organism, (2) an estimated confidence score for the prediction, and (3) a latent vector, the space of which we induce a metric on that measures whether

two predictions are trying to predict the same body. To train the network, each output quantity is associated with a specific loss term, where, importantly, the centerline loss term is permutation-invariant in the labels. To resolve overlap, we do non-max suppression [181], but rather than measuring distances between curve predictions, we use the latent space output, which allows two predictions to be kept even though they are close in physical space. This enables correct predictions for data in which objects overlap very closely. Our method is further tailored to support the subsequent tracking process, which must link uniquely predictions from frame to frame. To that end, we not only predict the object location at a single timepoint, but also predict consecutive past and future centerlines. Using these time-resolved predictions in the linking process enables high-precision tracking even through dense regions.

Our method is in-principle applicable to all microscopy datasets that involve slender bodies, but we do not develop its general applicability here. Instead, we focus on its applications for tracking dense experiments of swimming *C. elegans* worms, a popular model system in neuroscience [204], human diseases [205], drug discovery [177], motor control [206], memory [207], and ageing [208]. Studies of *C. elegans* often rely on phenotypic assays that measure the motility of the nematode worms as a function of some environmental condition or treatment [55, 180, 209–220], the throughput of which can be massively increased if overlap between organisms can be tolerated. Likewise, resolving identities of organisms during overlap is crucial for studies of interactions between organisms [132]. Previous work on tracking *C. elegans* have generally employed classical computer vision approaches to accurately track single or a few high-definition worms [182, 221–225], or many low-resolution worms at non-overlapping densities [54, 226, 227], in some cases by utilizing a computational model of the worm motion for hypothesis tracking [60, 182, 185, 221].

Recently, deep learning techniques have been utilized to track *C. elegans* worms using e.g. bounding box predictions [61, 228, 229] and fully resolved centreline in the case of isolated worms [230], allowing for detection also during periods of self-overlap.

With this paper, we publish a dataset of videos of motile *C. elegans* worms imaged at a wide range of densities. The dataset includes $\sim$ 1,500 labelled midlines that we use to evaluate, but not train, our detection model. We demonstrate that our model can be trained exclusively using synthetically generated data and yet generalizes well to real videos. Our method leverages the parallel capabilities of convolutional neural networks and is thus able to handle thousands of detections in a single pass, resulting in real-time detection at $\sim$ 90 Hz at 512×512 resolution on a single GPU. The code is open source and available at `https://github.com/kirkegaardlab/deeptangle`.

2.2 Methods

2.2.1 Model structure

Centerline predictions We choose to represent the centre-line of the slender bodies of interest by arrays consisting of k equidistant points [Fig. 2.2d]. These coordinate arrays,

which we refer to as centerline, become high-precision descriptors even for complex shapes when k is chosen large. To reduce the complexity of predicting k points, we embed the centerline representation with a principal component (PCA) transform $\mathbf{A}$, the dimension κ of which can be much smaller than k [189]. The PCA components $\boldsymbol{\lambda}$ represent shape, and in addition hereto, the network also predicts the offset $\boldsymbol{x}_0$ of the centerline, the internal calculation of which is done in a local coordinate system defined by the anchor points. Thus, instead of predicting $2k$ floating point values per centerline, the network needs only output $\kappa + 2$.

The temporal context of the input image stack permits output centerline prediction also for the non-central images. In our approach, we predict a set of three centerlines $\boldsymbol{z} = [\boldsymbol{x}^-, \boldsymbol{x}, \boldsymbol{x}^+]$ corresponding to the three central frames $[I^-, I, I^+]$ of the input stack [Fig. 2.2c]. We consider the middle centerline $\boldsymbol{x}$ the main output, whereas the past $\boldsymbol{x}^-$ and future $\boldsymbol{x}^+$ centerlines are considered auxiliary predictions whose main purpose lies in their use during the latent space encoding as well as the tracking procedure.

We define the similarity measure between two centerlines by the standard Euclidean distance. In the case of detections that look symmetric from either end, we exploit this symmetry and employ the flip-invariant distance defined by

$$d^2\big(\boldsymbol{x}, \boldsymbol{x}'\big) = \min\Big[\sum_{i=1}^{k}(x_i - x'_i)^2, \sum_{i=1}^{k}(x_i - x'_{k-i+1})^2\Big], \tag{2.1}$$

as illustrated in Fig. 2.2e.

Likewise, we define a distance between two collections of consecutive centerlines $\boldsymbol{z}$, $\boldsymbol{z}'$ by their weighted average $d_s^2 = \sum_t \omega_t\, d^2(\boldsymbol{z}_t, \boldsymbol{z}'_t)$, where the weights can be adjusted to give focus to central predictions, and for the present case we choose $\omega = 2\omega^- = 2\omega^+$.

The neural network is trained to minimize the distance d_s^2 between predictions and labels. To do so, we let the independent predictors specialize for different shapes. This is achieved by using a permutation-invariant loss such that the total loss is computed as a sum over the labels only, each using the predictor that best match the labels. Thus many centerline predictions will not contribute to the detection loss.

Confidence scores Each independent prediction of the network includes a confidence score s, which is used to filter out bad candidates. In bounding box or mask detection, intersection over union (IoU) is commonly used to evaluate the accuracy of a prediction, however, this metric does not generalize well to centerline predictions when there is overlap. Instead, we introduce a custom metric to define the goodness of a centerline set $\boldsymbol{z}$ by comparing it to its label $\hat{\boldsymbol{z}}$,

$$\hat{s} = \exp\big(-d_s^2(\boldsymbol{z}, \hat{\boldsymbol{z}})/\sigma_s^2\big). \tag{2.2}$$

Here, σ_s is a parameter that sets the scale over which the score varies. The metric is sensitive to perturbations on accurate predictions, i.e. predictions close to labels where

$d_s \to 0$, but loses sensitivity the worse the predictions are. This is a useful feature as correct scoring for good predictions is crucial for choosing the best one, whereas low-scoring predictions are discarded in any case and their relative scoring therefore unimportant.

The score prediction is trained using L2 loss. To avoid conflicting backwards error propagation between this task and that of centerline prediction (as scoring bad predictions is easier), we stop the gradient flow in the computational graph on the last layer of the score-predicting part of f_θ [Fig. 2.2a] such that it does not interfere with the accuracy of the predicted centerlines.

Latent space for candidates suppression Finally, we need to ensure that there is only one prediction per object. Bounding box detectors let the user decide the fraction of overlap between prediction boxes of the same class that should be considered to be targeting the same object. As our method must work at high densities, this task is complicated by the fact that two predictions might be very close, even completely overlapping in the central frame, and yet represent different objects. The task of choosing a suitable cutoff distance is therefore difficult, and we make this a trainable task. We do so by embedding each prediction in a low-dimensional latent space in which comparison between predictions is cheap, thus allowing efficient and fast candidate suppression also at high densities.

Our method computes the latent vectors $\boldsymbol{p}$ for predictions using an auxiliary neural network, q_ϕ which acts directly on the eigenvalues $\boldsymbol{\lambda}$ and offsets $\boldsymbol{x}_0$ rather than the more redundant centerline coordinate points. We induce a Euclidean metric on the latent space with the interpretation that two predictions i, j are predicting the same object with probability

$$\mathbb{P}(i \leftrightarrow j) = \begin{cases} \exp\left(-||\boldsymbol{p}_i - \boldsymbol{p}_j||^2\right) & \text{if } ||\boldsymbol{x}_{0i} - \boldsymbol{x}_{0j}|| \leq \sigma_l, \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

Here, σ_l is a real-space visibility cutoff that prevents far predictions to interact in the encoded space, thus avoiding the need to scale the dimensionality of the latent space with the number of candidates or the input size. We note that when using the flip-invariant metric d_s on centerlines, we explicitly construct the latent space encoder to likewise be flip-invariant.

To train the latent space, we assume that during training predictors are 'trying' to predict the label closest to the prediction centerline. Combined with the probability interpretation, this allows us to use binary cross entropy as a loss function for the probability defined in Eq. (2.3). To avoid wrong clustering between undefined close-by predictions, the loss contribution of each prediction is scaled by the product of their real scores $\hat{s}_i \hat{s}_j$, thus ensuring that the network focuses its attention on good predictions that will not be filtered out. Finally, since the encoder should not alter the performance of the centerline suggestions, the loss on the latent space representations only updates the weights q_ϕ of the encoder, but is trained concurrently with the main model.

We employ non-max suppression to choose the best prediction of each object, but with distances measured in latent space, as illustrated in Fig. 2.2b. Concretely: Once all the predictions whose score is lower than a threshold τ_s have been discarded, multiple candidates are likely to still remain for each target object. The lack of low score predictions exposes clusters in the latent space that correspond to single objects. We sort the remaining predictions by their score, automatically accepting the highest-scored one. Once a prediction i is accepted, all predictions j that have a high probability $\mathbb{P}(i \leftrightarrow j) > \tau_o$ of being the same object are removed. This is equivalent to setting an exclusion radius r_l in the latent space as shown in Fig. 2.2b. We keep iterating on the remaining predictions, pruning the latent space until all candidates have been iterated. The final number of accepted predictions should equal the number of objects in the frame.

2.2.2 Neural Network Architecture

Convolutional neural network Most of the weights of the network are at the feature detection convolutional network whose backbone is made of four ResNet groups consisting of $2, 4, 4, 2$ blocks with strides $1, 2, 1, 2$, respectively. We modify the original ResNet architecture by replacing the initial max-pooling layer with an average-pool layer to avoid translational invariance. The final shape of the feature space is $[H/16, W/16, C]$, with C being the number of candidates each cell proposes. We have set $C = 8$ for this project in order to fulfill the condition of the number of predictions being larger than the number of bodies even at high densities. All in all, there will always be C candidates per cell regardless of input size, which leads to a large number of candidates to be sorted in the filtering process. The head of the convolutional neural network is composed of two fully connected layers of 512 and $C \cdot (3(m + 2) + 1)$ cells, respectively, with batch normalization in between. Due to the orientation invariance of the loss function on the centerline predictions, it is possible that the centerlines in the predicted set $\boldsymbol{x}^-, \boldsymbol{x}, \boldsymbol{x}^+$ are not aligned. To remedy this, we aligned them by comparing them with the eigenvalues of the flipped centerline. In order to get the *flipped eigenvalues* λ_f, we use

$$\lambda_f = \mathbf{A^{-1}JA}\lambda \tag{2.4}$$

where $\mathbf{A}$ is the PCA transformation matrix and $\mathbf{J}$ is the exchange matrix.

Latent space encoder The encoder q_ϕ is composed of two fully connected layers with batch normalization in-between. The input of the encoder is the vector of size $3(m + 2)$ characterizing the centerline predictions and the output is D floating point values, corresponding to the coordinates of p in the D-dimensional latent space. We have found $D = 8$ to be a well-performing dimension in our experiments. Due to the orientation invariance of the centerlines predictions, we need to construct the encoder to cluster those centerlines regardless of orientations as well. To do so, the input values are expanded to include those of the flipped centerlines $\lambda \to (\lambda, \lambda_f)$ and both are fed to

the same layer. To ensure symmetry, the output is then summed before passing through the last layer. In doing so, the encoder becomes independent of centerline orientation.

2.2.3 Training

Simulation-based training Our in-silico data generator has two main components: a physics-based model for the organism and a synthetic frame generator.

In-silico worms are generated on demand every training step which removes the possibility of overfitting to the generated frames. In order to train the model to work effectively with a range of worm densities, we generate batches with different numbers of worms in a uniform manner, without bias towards low or high worm counts. This teaches the model to handle a variety of densities without overfitting to any specific case. And to make the model more robust, training also happens on densities whose manual annotation would be extremely challenging. The simulation and video synthesis are implemented in a GPU framework which enables fast end-to-end training without the performance penalization of data transferring between the accelerator and the host machine.

We base the worm simulation on resistive force theory, as it has previously been shown to correctly predict the position of the skeleton for short spans of time [231]. Since the network only perceives the frames surrounding the target frames, we found the total duration of the clip to be short enough that a linear swimming model approximation fits our needs. The physics-based model should encapsulate all types of organism behavior. This can be achieved by oversampling the behavior, i.e. by making the simulations more diverse in the behavior than reality and thus hope to include all types of real behavior as well. Details on the worm simulation and video synthesis can be found in the in-silico dataset section of the methods.

Despite the potential for physics-based simulations to be used for synthetic training data, discrepancies with real data may lead to inaccuracies when applied to real microscopy images. This reality gap can be the result of an overly simplified motility model or physics model, or the result of imprecise video synthesis. The gap may be further increased by the fact that the model relies on the PCA transformation matrix $\mathbf{A}$ obtained on synthetic data, where the number of PCA components used have been chosen to accurately reproduce all synthetic patterns, but not necessarily to generalize to out-of-sample videos. Thus we find that our model is limited to accurate skeleton predictions only on shapes that resemble those produced by our simulations, and the goal of the simulations is therefore to reproduce a broad spectrum of possible motility patterns. Likewise, we find that our model is susceptible to the brightness of the videos, and accordingly we adjust the real videos to increase their resemblance to the training data.

Loss functions Centerline descriptors are trained as a regression problem. Thus, the loss contribution is given by the custom distance defined in Eq. (2.1). To enforce specialization on the predictors, and due to the number of predictions M being considerably

larger than the number of bodies N, only the best predictors are accounted for in the loss. Nevertheless, there may be labels $\hat{\boldsymbol{x}}$ completely or partially outside the frame at t_c, despite being inside at t_0. To make sure not to punish bad predictions at the boundaries for not matching *invisible* centerlines, instead of using the number of simulated bodies N, the subset of bodies completely inside the frame N_v is used and the final loss expression is given by:

$$l_x = \frac{1}{N_v} \sum_{i}^{N_v} \min_m d_s^2(\boldsymbol{z}_m, \hat{\boldsymbol{z}}_i) \tag{2.5}$$

The score L2 loss is computed as the difference between the values predicted and the score the centerline proposals should have. Thus, using Eq. (2.2), we train the predicted score of all predictions using:

$$l_s = \frac{1}{M} \sum_{i}^{M} \left(\exp\left(-\min_n \frac{d_s^2(\boldsymbol{z}_i, \hat{\boldsymbol{z}}_n)}{\sigma_s} \right) - s \right)^2 \tag{2.6}$$

Finally, the loss function for the latent space encoder is a modified cross entropy loss scaled by the product of scores. Denote $\mathbb{P}_{i,j} = \mathbb{P}(i \leftrightarrow j)$ as defined in Eq. (2.3), then the encoder loss is defined as an average over all pairs of predictions $\langle i, j \rangle$ that are physically within the cutoff σ_l,

$$l_p = \frac{1}{S} \left\langle \hat{s}_i \hat{s}_j (t_{ij} \log(\mathbb{P}_{i,j}) + (1 - t_{i,j}) \log(1 - \mathbb{P}_{i,j})) \right\rangle_{\langle i,j \rangle}, \tag{2.7}$$

where $S = \sum \hat{s}_i \hat{s}_j$, and $t_{i,j}$ indicates whether i and j are targeting the same label k, and is set by

$$t_{ij} = \begin{cases} 1 & \text{if } k_i = k_j \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

with k_i, k_j being the closest labels to the predictions $\boldsymbol{z}_i, \boldsymbol{z}_j$ respectively.

Training details Training has been done from scratch, i.e. without the use of a pre-trained backbone. During training, the frame size for the input clips used was 256×256, but due to the anchored approach, this does not constrain inference to happen at the same resolution. Synthetic input is generated on demand and on device rather than using a fixed pre-generated dataset. Thus, the network never sees the same frame twice and there is no host-to-device data transfer. As mentioned in the main text, all networks are trained simultaneously, despite the weights of each one depending on different cost functions. The code has been written in Jax using Haiku and training has been carried out on a cluster of 8 $\times$ NVIDIA A5000's.

2.2.4 Inference

Inference happens at any resolution whose dimensions are multiple of 16. The input frames need to be slightly pre-processed as mentioned in the previous sections. Candidate predictions are chosen using a score threshold, and non-maximum suppression in latent space is used for filtering. Due to the sequential nature of the filtering process, the implementation is written to use the CPU using NUMBA.

Input clips pre-processing The images used to train the model have dark (small pixel intensity) backgrounds, as we employ zero-padded convolutional layers. This is relevant for real recordings, where a negative flip may be necessary to match the network requirements. During training, generated clips are normalized using a 199 percentile normalization. For real clips, to accommodate uneven lighting conditions and potential obstructions we apply contrast limited adaptive histogram equalization (CLAHE) and subsequently correct the intensity of videos to match the variations of the simulated data (see SI figure). Note that we match real data to the synthetic as this avoids the need to retrain the network for different experimental setups.

2.2.5 In-silico dataset

Worm simulation Worm trajectories are computed by employing a resistive force theory swimming model used to predict rigid body motions of *C. elegans* from the undulations [231]. Thus, we ensure that from a given set of generated undulations, the produced motions will match those of real worms. From empirical observations, we propose a simple Eq. (2.9) to generate the undulation of swimming adult worms. The set of equations is specifically targeted to the dataset of dense swimming C. elegans, but we expect it to also apply to other life stages by modifying the parameters of the sampling distributions. Similarly, the undulations proposed do not take into account self-coiling, as it is rare on free swimming nematodes, but changing Eq. (2.9) appropriately would allow the system to learn to detect them. We define the motions by the centerline angle $\psi_k(s)$ with $s \in [0, 1]$ [Fig. 2.2d], and decompose this into a linear combination:

$$\psi(s) = \psi_u(s, t) + \psi_s(s, t). \tag{2.9}$$

This logically separates the worm undulations into two types of motion: one corresponding to a sinusoidal motion ψ_s and one in which the whole body bends ψ_u. These we define by

$$\psi_u(s, t) = A \cos\left(\frac{2\pi}{T}t + \rho_1\right) \cos\left(k_u s_k + \rho_2\right) \tag{2.10}$$

$$\psi_s(s, t) = \tilde{A} \cos\left(\frac{2\pi}{T}t + k_s s_k + \rho_3\right) \tag{2.11}$$

where $\tilde{A} = \frac{1}{2}\left(1 + |\sin(2\pi t)|\right) A$ and the rest of parameters are sampled from random distributions. Although many improvements for the above equations can be suggested, we prefer to keep the model simple.

Once the values of the parameters for ψ are generated for all the timesteps of the simulation, the positional coordinates are obtained using

$$\vec{x}(s,t) = L \int_0^s \begin{pmatrix} \cos\left(\psi(s',t) + \gamma\right) \\ \sin\left(\psi(s',t) + \gamma\right) \end{pmatrix} \mathrm{d}s' \tag{2.12}$$

where γ is a random orientation and L is the length of the worm (also sampled). Once the skeleton is defined, the rigid body motions are predicted by solving [231]

$$\vec{F} = \int_0^L \vec{f}\,\mathrm{d}s = 0, \tag{2.13}$$

$$\vec{\tau} = \int_0^L (\vec{x} - \vec{x}_{\mathrm{CoM}}) \times \vec{f}\,\mathrm{d}s = 0, \tag{2.14}$$

where the force $\vec{f}$ can be calculated from the centerline velocity $\vec{U} = \partial_t \vec{x} + V + \Omega \times (\vec{x} - \vec{x}_{\mathrm{CoM}})$ by

$$\vec{f} = \alpha_t \left(\hat{t} \cdot \vec{U}\right) \hat{t} + \alpha_n \left(\hat{n} \cdot \vec{U}\right) \hat{n}. \tag{2.15}$$

Here, V and Ω are the center-of-mass velocity and rotational velocity (that we are solving for), and α_t and $\alpha_n = \alpha\,\alpha_t$ is the tangential and normal drag coefficients, which is also sampled for ($\alpha > 1$). We did not find a need for using a non-linear force theory. The simulation is run with Python 3.9 using the Jax library.

Video synthesis Given the labels for the centerlines positions, synthetic videos are generated to be used as input during training. In order to add width to each worm, we vary the local body radius r by a function of the form

$$r(s) = \tilde{R}\,|\sin(\arccos(as + b))| \tag{2.16}$$

The pixel values of those circles are calculated with anti-aliasing. Once the worms have been rendered, noise artefacts such as uneven background, blurring, Gaussian noise, etc. are added to replicate the observed conditions of real experiments. During training, standard augmentation techniques are applied as well. In the same manner as the simulation of the motion and the neural network training, frame generation is also written in Python using the Jax library in order to leverage GPU capabilities.

2.2.6 Evaluation

Experimental dataset Videos of swimming *C. elegans* were filmed using the protocol described in Ref. [54].

Manually annotated dataset The evaluation dataset is annotated using a custom tool that can be found at `https://github.com/kirkegaardlab/deeptanglelabel`. Around $\sim$ 1,500 centerlines have been annotated (see data availability)

Asymmetric dynamic time-warped error metric We introduce a custom metric to suitably compare the densely defined centerlines of the predictions to labels that are defined using only a few labelled points. The metric used must be shift-invariant, as having points anywhere along the centerline should yield zero error regardless of whether the label points precisely coincide with the prediction points or not. Likewise, label points should be monotonically assigned along the centerline in order to avoid artificially reducing the error for strongly bent or self-coiling worms. Finally, it must be robust against the subjectivity of the labellers, as manual annotations might miss or avoid spots where visibility is low such as the end-points of the worms.

To satisfy all these requirements, we introduce a metric based on the dynamical time warping (DTW) distance used to measure the similarity between temporal curves. In our modified version, asymmetric DTW, summation only runs over label points. Thus, the metric δ_{adtw} is defined as follows: Let $d(i, j)$ be the Euclidean distance between label point i and prediction line segment j, then

$$\delta_{\mathrm{adtw}} = \min_{\alpha} \frac{1}{N} \sum_{i=1}^{N} d(i, \alpha(i)), \tag{2.17}$$

where $\alpha : [1, N] \to [1, M]$ is a monotonic (non-decreasing or non-increasing) assignment of the N label points to the M prediction line segments. A visual representation of the metric is shown in Fig. 2.4b, and the $\mathcal{O}(NM)$ algorithm for its calculation is detailed below. We note that, just as is the case for the dynamic time warping distance, this is not a true distance in the mathematical sense.

2.3 Results

2.3.1 Architecture

Our model is based on single-stage detection models [181, 200] that output many candidate predictions per target in a single forward pass and rely on a score system to prune until a single candidate is left for each target object. The performance of such single-stage models has been shown to enable accurate real-time bounding box detection [202]. Fig. 2.2 illustrates the overall structure of our approach. The backbone of our neural network [Fig. 2.2a] consists of convolutional residual networks [24] and the output of our model is composed of a set of centerline predictions $\boldsymbol{z} = [\boldsymbol{x}^-, \boldsymbol{x}, \boldsymbol{x}^+]$ representing the past, present, and future motion of the bodies. We represent the centerlines by k equidistant points along the center of the body [Fig. 2.2d]. The centerlines contained within the set maintain alignment with a consistent head positioning across the three

predictions. In addition, the model outputs confidence scores s and latent vectors $\boldsymbol{p}$ that are used for subsequent filtering [Fig. 2.2b] (see Methods).

We take the input to our model to be a stack of consecutive frames in order to provide the model with a temporal context [Fig. 2.2c]. In the present case of motile slender objects where dynamic crossings and overlap between objects are very common, a temporal context can provide the necessary information to resolve the problem of correct identification. Furthermore, the temporal context allows the output of our model to include information on the motion of the centerlines, which we will further exploit for tracking purposes.

The backbone of our neural network performs a 16^2-fold reduction in resolution when mapping the input images to feature space, from which the network outputs multiple anchored predictions. This anchored approach means that the only restriction on input size is that its dimensions be divisible by 16, and, in particular, it allows training at a certain resolution $H \times W$ and subsequent inference at another $H' \times W'$ without loss of accuracy. We choose the resulting number of candidates to be considerably larger than the number of objects in the frame, thus ensuring that all objects have suggestions.

2.3.2 Detection on dense *C. elegans* experiments

To evaluate our approach, we study microscopy videos of swimming *C. elegans* worms. We are particularly interested in videos captured at much higher densities than those typically used in motility experiments. Thus we evaluate our model on wide-field videos captured under approximately uniform illumination [54], exemplified in Fig. 2.3a. In our dataset, the number of nematode worms varies ranging from ~ 400 with a small probability of overlap occurring (≈ 0.05 average overlaps per worm) to extremely densely packed plates with up to $\sim 6{,}000$ nematodes, where there is, on average, one overlap per worm. This means that in the dense plates, detection methods that stop tracking after contact between worms happens are rendered completely ineffective.

Defining worm density ρ as the number of worms in a region per square millimeter, we find, as expected, a linear relation between the average amount of overlap per worm and the density [Fig. 2.4a]. Due to the spatial heterogeneity of the worm distribution inside the plate, higher densities can be observed when considering small regions. On $100\,\mathrm{mm}^2$ scales, the highest density in the dataset is $\rho \sim 2.5\,\mathrm{mm}^{-2}$, but this jumps to an extreme $\rho \sim 3.5\,\mathrm{mm}^{-1}$ when considering $10\,\mathrm{mm}^2$ regions, where humans begin to struggle to correctly identify worms. For quantitative evaluation of our model, ~ 200 random regions of the videos were sampled and hand-labelled resulting in $\sim 1{,}500$ labelled worm centerlines. A sample of frames is shown in Fig. 2.3b to provide a sense of the different densities encountered in the evaluation dataset, with the predictions of the model overlaid.

To train our network, we implement a physics-based synthetic dataset generator to exploit perfectly defined labels (see Methods). This approach removes the need for a supervised dataset, and also allows labelled videos in situations where manual labeling may not be reliable, or where the subjectivity of the human labellers can result in inconsistent labels. Physics-based synthetic datasets have successfully been used to

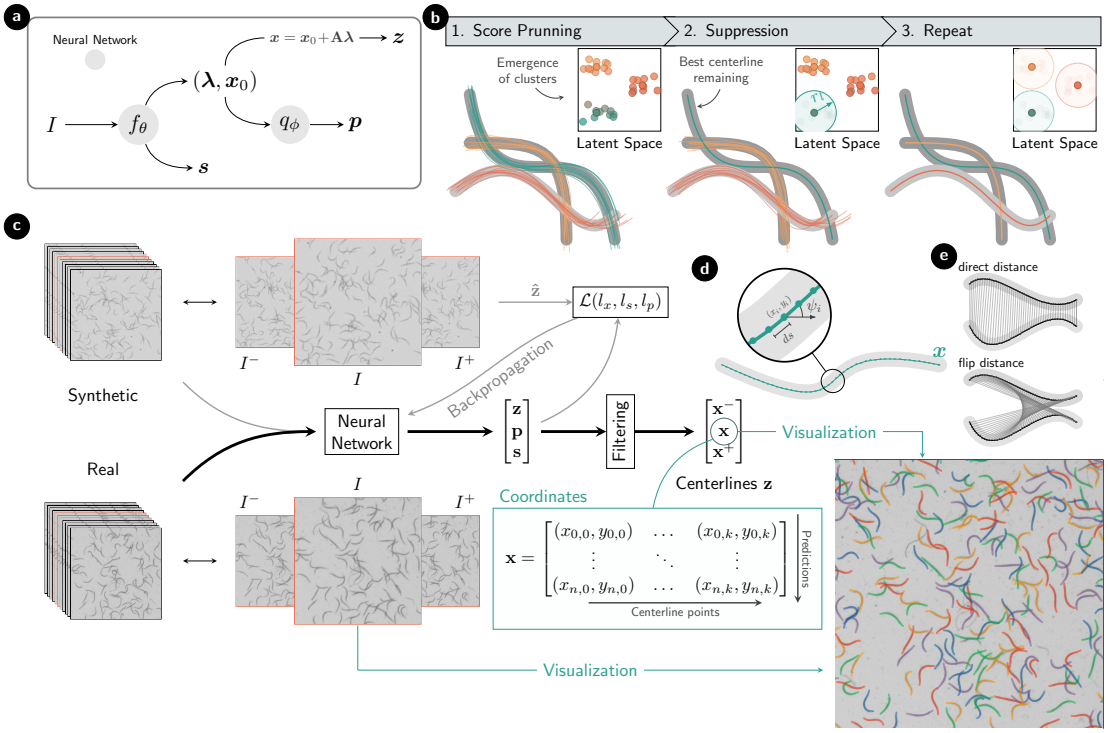


Figure 2.2: **Method workflow.** **(a)** Structure of the detection method. Trainable neural networks are colored in gray, and represent the convolutional neural network $f(I;\theta)$ and the latent space encoder $q(\lambda, x_0; \phi)$. **(b)** Procedure to prune unfiltered predictions to final detections with the use of the encoded latent space vectors. **(c)** Method overview from the input clip I (we use a stack of 11 frames in this work) to the final matrix of centerlines $\boldsymbol{x}$. The target frames $[I^-, I, I^+]$ (center frames from the clip, orange) are explicitly shown for both the synthetic and real videos. Additionally, the training setup is represented using lighter color arrows; from synthetic data to loss backpropagation. After detection, direct visualization of the predicted centerlines $\boldsymbol{x}$ is possible. **(d)** Diagram with a centerline descriptor composed of k equidistant points along the skeleton of the nematode. **(e)** Visual representation of the two distances used in Eq. (2.1), the minimum of which corresponds to correct head-tail alignment and is the one that will be used in the model.

train systems on similar conditions, for instance where manual labelling may introduce unnecessary noise or bias to the model [162]. Naturally, this requires the formulation of a physical model that is accurate on the relevant time scales. Furthermore, dependence on synthetic datasets can result in a divergence between the target and the training data, potentially leading to inaccurate predictions during inference a phenomenon that is avoided if the model is trained on real data.

Performance Despite being trained exclusively on synthetic data, the model's inference performance is very good on real clips. From visual inspection, no immediate discrepancies are observed between detections in low density clips and at high density [Fig. 2.3b]. Likewise, per design, the network accuracy is independent on the input clip dimensions, and the parallel structure of convolutions permits the use of large videos

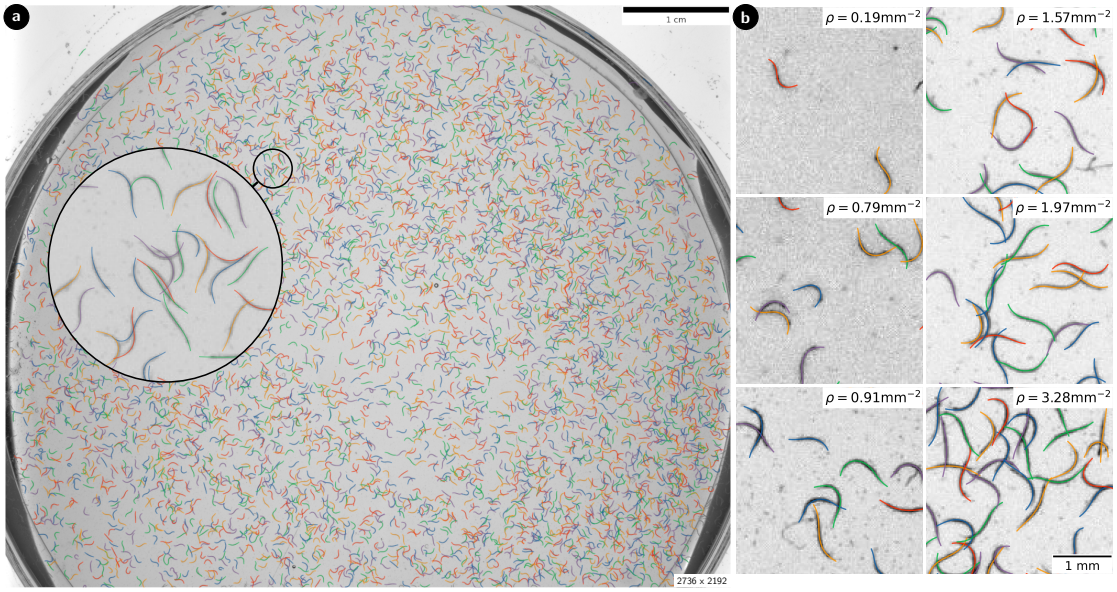


Figure 2.3: **Qualitative showcase of the capabilities of the model. (a)** Detected centerlines predicted on an entire densely populated well plate with a single forward pass through the neural network. Inset shows a zoom-in section to demonstrate the accuracy of detection across the entire plate (except near borders, where the plate interferes). The total plate contains around 6,000 detections. **(b)** Close-up evaluation of different experimental clips with different densities of worms.

covering thousands of nematodes to be processed simultaneously in a single forward pass [Fig. 2.3a]. We note, however, that even though no quality impact on detections is observed when using large fields-of-view clips, there can be a dependency if non-uniform illumination is used as different sections of the frame may have different requirements for preprocessing.

For a quantitative assessment of the method accuracy, we compare to the manually labelled dataset, an example of which alongside the model predictions can be seen in Fig. 2.4c. As the predictions are densely defined centerlines (here, ~ 50 points), we used an asymmetric version of dynamic time warping δ_{adtw} (defined in Methods and illustrated in Fig. 2.4b) to evaluate the accuracy of the predictions using labels with lower fidelity.

The results of evaluating the trained model on the labelled dataset are shown in Fig. 2.4. For reliable comparisons, we first solve the assignment algorithm for the label-prediction pairs. This means that in the case of two completely overlapped worms, two predictions need to be present to not count as a miss, and likewise, two predictions cannot be considered to target the same label. We find an average error of $\delta_{\text{adtw}} \approx 0.54\,\text{px}$ with no strong dependency between accuracy and density of worms [Fig. 2.4d], with the exception of a slight increase in error for extremely dense clips ($\sim 3.5\,\text{mm}^{-2}$). The average error corresponds to less than the width of a worm ($\approx 2\,\text{px} \approx 50\,\mu\text{m}$), and part of this can be attributed to the fact that human accuracy is also near the half-pixel level [Fig. 2.4c]. Some outliers can be seen however, which can mostly be attributed to an artefact of the model, where the network mistakes a single long worm for two

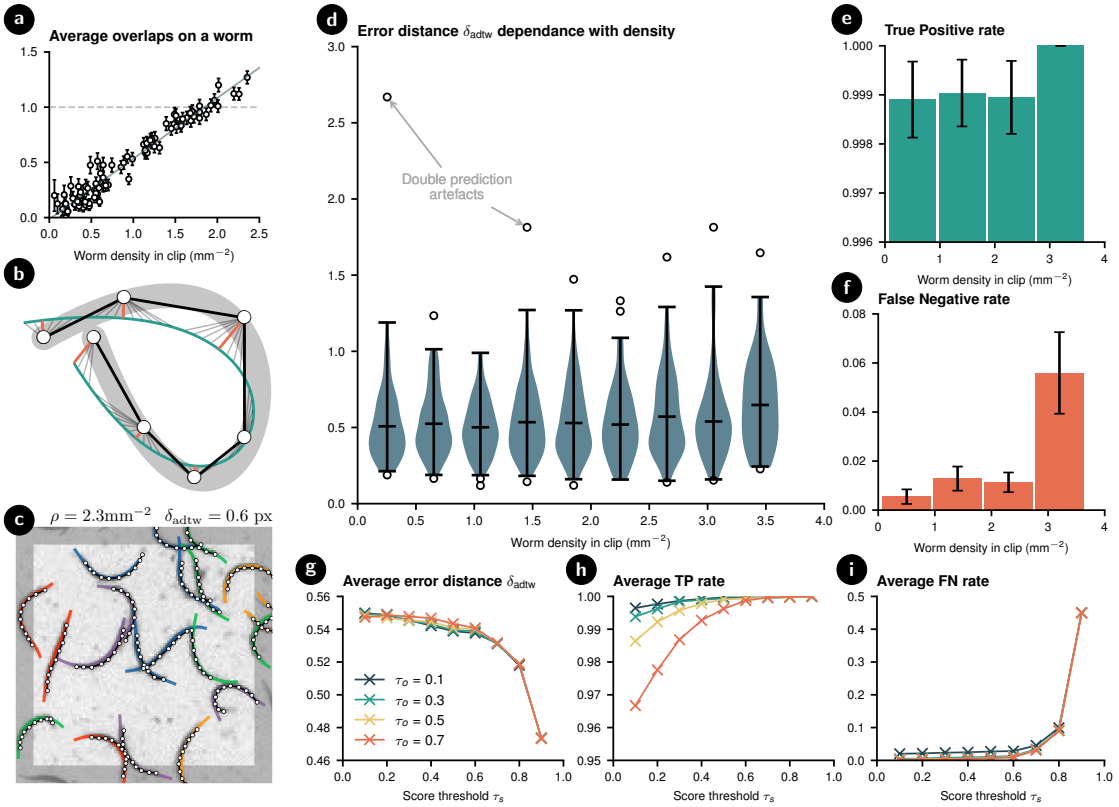


Figure 2.4: **Quantitative performance metrics on the detection of slender bodies in dense experiments of swimming nematodes. (a)** Average number of overlaps counted on frames of pixel size 512×512 with different densities of worms ($N = 90$). **(b)** Illustration of the asymmetric dynamic time warping distance error corresponding to the average value of the orange euclidean distances between the prediction (green) and the labelled points (white). **(c)** Example frame with manually labelled points (white) and models predictions (colored). The metric is only evaluated in the lighter area of size 100×100. **(d)** Quantified accuracy of the detections by showing the distance to the manually labelled centerlines. Distributions for different densities are shown. The violin plots represent the 99 percentile of the data whereas outliers are plotted individually. **(e–f)** Rates for True Positive and False Negative on the manually annotated dataset. **(g–i)** Performance of the model with different combinations of score (τ_s) and overlap (τ_o) thresholds. $N = 1{,}420$. Error bars indicate standard error.

overlapping shorter predictions. This effect seems particularly sensitive to incorrect intensity normalization of the videos.

Let σ_ϵ be a cutoff distance above which we no longer consider the predictions to be targeting the closest label. For all the figures in Fig. 2.4, this cutoff is assumed to be $\sigma_\epsilon = 3.0\,\text{px}$, and we observe no significant changes by tuning it within the range of sensible values. We define the True Positive (TP) rate as the fraction of predictions that both get assigned a label and this label is within the distance σ_ϵ of the prediction. Fig. 2.4e shows that the model rarely predicts a centerline where there is nothing with a TP rate of 0.999. Nevertheless, there are some predictions that do not get assigned a label which can be attributed to the double-prediction artefacts just mentioned. The

likelihood of this happening decreases with density, but the rate is so low that it is almost negligible. Similarly, we define the False Negative (FN) rate as the fraction of labels that are not assigned a prediction closer than σ_ϵ. Fig. 2.4f shows that the model in general manages a low FN rate at around ~ 0.015, but that this increases to a rate of ~ 0.06 at extreme densities such as $\rho \geq 3.0$ mm^{-2}, where clusters tend to be densely packed and manual labeling likewise becomes challenging.

The filtering process depends on two user-defined thresholds: the score threshold $\tau_s \in [0, 1]$ is used to prune predictions with low confidence scores [Fig. 2.2b(1)] and the overlap threshold $\tau_o \in [0, 1]$ is used for filtering by setting the maximum probability of two independent predictions to be targeting the same object [Fig. 2.2b(2)]. Throughout this paper, we have set these to $\tau_s = \tau_o = 0.5$. We evaluate how different combinations of thresholds may alter the performance results. Fig.s 2.4g–i show the average performance obtained across all densities when filtering the predictions with variable thresholds. In spite of some dependency between worm density and TP/FN rates, we consider the average metric to be a good indicator of the performance for each case.

Fig. 2.4g shows the effect of the thresholds on accuracy. No significant dependency on the thresholds is observed. This can be explained by the fact that accuracy is determined by the best predictors only (through assignment), which are not discarded until a high τ_s is used, and once those are removed, τ_o becomes irrelevant. Further, the fact that there is no notable difference between different values of τ_o indicates that the clusters are highly compact.

In contrast, Fig. 2.4h shows that the TP rate has a stronger dependency on τ_o at low τ_s because low score predictions do not form compact clusters, and therefore a larger exclusion radius is required to discard them. Finally, Fig. 2.4i shows that misses only begin to occur once the best predictions are discarded, and a strong dependence on the τ_s is not observed before that point.

2.3.3 Tracking from consecutive detections

Motility assays require not only accurate detections but also the ability to link these across frames to form time-resolved tracks of individual organisms. This is challenging at high densities where we have the breakdown of the assumption that the closest detected object to the previous frame corresponds to the same identity. In general, greedy approaches to particle tracking such as assigning directly the closest particle in consecutive frames frequently leads to failed tracks. Instead, the process of tracking can be efficiently formulated as a set of linear assignment problems [232]. Naturally, here we can expand upon particle tracking by using a metric that measures distances not between center-of-mass of the worms, but between the full centerlines [Fig. 2.2e]. This works well for most predictions but can fail for fast-moving worms or in dense clusters.

A separate approach to tracking is Kalman filtering. This would require separate detection of entry and exit events of worms, as well as a probabilistic model for worm motility, which would most likely have to be highly non-linear. Kalman filtering is viable for the tracking of few organisms, but for present large-scale systems we require a more efficient approach. As our model also outputs centerlines from adjacent frames (to embed

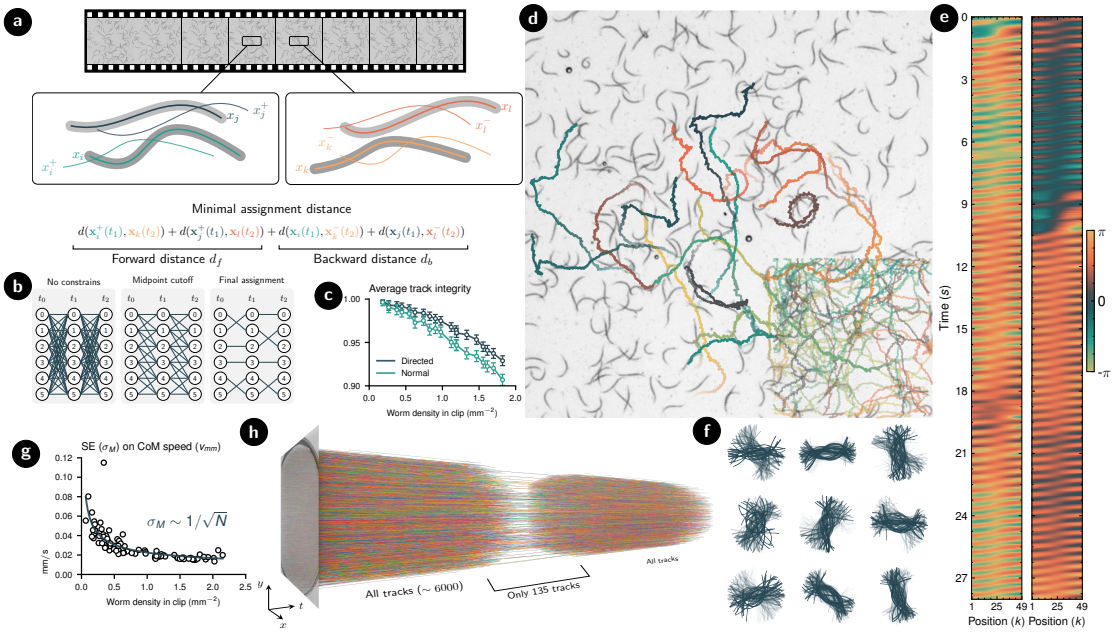


Figure 2.5: **Tracking methodology and results.** **(a)** Illustration of the directed distance used to assign consecutive detections of the same identity. The simplified drawing shows two independent predictions at adjacent frames and showcases how the assignment scheme computes the identity by comparing future-present and past-present distances and choosing the assignment that minimizes their sum. **(b)** Diagram showcasing how using a location cutoff simplifies the assignment problem. Nodes represent independent detections at each frame whereas edge values are given by the directed distance measure. The assignment happens by minimizing the sum of edges at each timestep. **(c)** Comparison of using the straightforward centerline distance and the proposed directed approach. The accuracy is evaluated by measuring the integrity of the tracks. In contrast to other metrics in this paper, this plot has been obtained using synthetic worms as long-term, accurate tracks are required to evaluate the tracking integrity (See Methods for details on Tracking integrity). Error bars indicate standard error ($N \sim 300$ at lowest density to $N \sim 3,000$ at highest density). **(d)** Qualitative example of 30 s trajectories of the center of mass of the nematodes in a dense experiment. The still background image represents the last frame of the video. To improve the visualization, a small subset of the trajectories is shown. In contrast, a corner of the frame is used to display all the trajectories to showcase the density of simultaneous tracks. **(e)** Two samples of the centerline angle ψ of two randomly sampled nematodes from (d). **(f)** Undulations corresponding to 30 s of the detections relative to the center of mass coordinate of nine randomly sampled nematodes from (d). **(g)** Standard error value of the measurements of the center of mass speed as a function of density. **(h)** Showcase of the possible throughput of the method, by simultaneously tracking more than 6,000 tracks from a full dense plate. A small window on the tracks is shown to showcase their continuity.

temporal information into the latent vector, see Methods), we propose a directed metric that leverages both past x^- and future x^+ centerlines predictions [Fig. 2.5a]. Thus to find a mapping σ from one frame to the next, we solve

$$\sigma = \arg\min_{\sigma} \left[\sum_i d(\boldsymbol{x}_i(t), \boldsymbol{x}^-_{\sigma_i}(t')) + d(\boldsymbol{x}^+_i(t), \boldsymbol{x}_{\sigma_i}(t') \right]. \tag{2.18}$$

Identity assignment can be seen as a network flow global optimization where nodes

represent detections and edges carry the cost of assignment. To avoid having to perform all possible combinations of assignments, we include a physical distance threshold on the midpoint of the central line. This threshold significantly simplifies the assignment scheme and improves the runtime of the filtering process [Fig. 2.5b]. Notice that due to the flip-invariance of the distance metric, consecutive assignments of sets of centerlines are not necessarily aligned, hence a trivial alignment of the centerlines is carried out during post-processing before being analysed. Likewise, head-tail alignment with the real worm is not granted and a post-processing step would be required to guarantee the alignment, e.g. by using temporal information such as the direction of the undulation wave [Fig. 2.5e].

To quantify the performance of these methods, we define the *tracking integrity* ι as a scalar that indicates how consistent the assignment of a label to a prediction is along the tracked video. Perfect tracks have $\iota = 1$, whereas labels that get assigned two different identities for half of the duration of the video have $\iota = \frac{1}{2}$, and so on (see Methods for a detailed definition). We evaluate this on synthetically generated videos of 10 seconds (200 frames) that have perfectly labelled tracks, the results of which are shown in Fig. 2.5c. On videos with densities up to 2.0 mm^{-2}, we achieve an average integrity of $\iota \approx 0.97$. This is a ~ 30 % improvement of the error over using direct detection assignment defined in Eq. (2.1). We observe that the integrity is almost perfect at low densities, but drops to $\iota \approx 0.93$ at the highest densities.

When applied to high density videos of *C. elegans*, the tracking method is able to keep track of individual worms as they pass through clusters of other worms [Fig. 2.5d] (videos in SI). In contrast to pixel-level classification of worms, our approach outputs centerlines directly, and thus subsequent analysis is straightforward. For instance, one may directly study the worm undulations [Fig. 2.5f] or extract the worm centerline angle $\psi = \arctan(y(s,t) - y_0(t),\, x(s,t) - x_0(t))$ to provide insight into the movement patterns and kinematics of the worm [Fig. 2.5e].

One of the key advantages of our methods is its ability to collect a larger number of samples compared to traditional techniques, while still obtaining reliable results. As the standard error decreases with the number of samples, using our methods allows for metrics to be gathered with less uncertainty while still requiring the same experimental setup. For instance, Fig. 2.5g shows how the error of estimating the average speed of the center of mass of the nematodes decreases with density. This advantage can be extended to tracking large numbers of nematodes in crowded environments, such as extremely dense petri dishes where more than 6,000 concurrent tracks can be simultaneously computed [Fig. 2.5h]. Thus, with our method, we are able to collect a larger number of samples and obtain more precise and reliable results, even in challenging conditions.

2.4 Discussion

We have introduced a deep learning approach for detecting and tracking slender bodies, such as swimming nematodes, in microscopy data. The presented convolutional neural network architecture is capable of accurately detecting a large number of overlapping

organisms, a task that can be particularly challenging for standard methods such as bounding boxes and pixel-level classifiers due to the issue of occlusion and overlap. To address this, we have implemented a latent space encoding which allows us to filter by non-maximum suppression and effectively handle overlapping objects. Not only is our method capable of accurately detecting and tracking slender bodies, but it also demonstrates strong scalability, performing well across a range of input frame sizes and densities of bodies. This makes it an ideal tool for a variety of experimental settings where centerlines are useful descriptors, including studies of swimming nematodes, swimming spermatozoa and beating eukaryotic or prokaryotic flagella.

Besides a suitable detector model, labeled training data is also needed. We have demonstrated that relying on a physics-based model to generate synthetic data is adequate to train our network to perform well on real data. This is a key achievement as it means that applications of our system for different experimental studies do not require large datasets to be procured, but rather the implementation of a suitable simulation. Our approach for synthetic data generation relies on over-sampling the behavior of the worms. This is naturally a trade-off as too extreme behavior can lead to datasets that are too hard for the neural network to replicate. For our model, we found that we slightly undersampled certain worm shapes such as strong coiling, which the model therefore could struggle with identifying. Though we did not look into this here, an interesting avenue for future research would be to bootstrap synthetic motility models on small datasets of real organisms. In a similar fashion, the frame-generator procedure should oversample the textures, pixel intensities and noise of real videos. Here, it could be interesting to study whether style transfer [161] or diffusion models [233] could be used to further reduce the gap between training and inference data. We note that we have only developed and studied a simulation of swimming *C. elegans* worms, and the study of other slender-body systems with our framework requires corresponding synthetic models.

For tracking, we introduced a directed metric that employs past and future centerline predictions to link them across time. At very high densities this may still fail, in particular because the directed metric yields little advantage if predictions are missing in some frames. A potential way to improve on this could come from utilizing the latent space encoding as well. This would require temporal continuity in the latent space representation, which is achievable by modifying the associated loss function. This should enhance the integrity of tracking, as it could potentially be used to resolve issues such as switches by leveraging the separation of close physical predictions with different temporal behaviour that characterises the latent encoding. We believe that these suggestions might be fruitful avenues for further research for improving deep learning models for dense detection of centerlines. Furthermore, we note that high short-time scale tracking integrities can still, over longer times, lead to loss of identity. The tracking integrity measurement thus sets the time scale over which accurate statistics can be formed. For longer times scales, other methods are needed [234].

Our approach differs significantly from previous approaches to slender-body tracking. For *C. elegans* tracking in particular, previous trackers have focused on either accurate

single worm tracking [230], few worm tracking [182, 185, 221–225], or large-scale tracking [54, 226, 227]. However, we found that none of these existing approaches were designed to handle the type of data and densities that we have presented here, and we thus omit quantitative comparisons.

In this paper, we have proposed a new approach for fast and precise detection and tracking of slender bodies in microscopy data. Its speed and accurate performance across a range of densities and sizes, combined with the ability to handle overlapping objects, make it a valuable tool for a variety of experimental settings where precise tracking is essential for obtaining quantitative metrics.

Chapter 3

On memory and cell size

The following chapter contains the paper:

Learning optimal integration of spatial and temporal information in noisy chemotaxis

Albert Alonso and Julius B. Kirkegaard

Niels Bohr Institute, University of Copenhagen, Copenhagen 2100, Denmark

Published in: *PNAS Nexus*
DOI: https://doi.org/10.1093/pnasnexus/pgae235
Pre-print server: https://arxiv.org/abs/2310.10531

Author Contributions:
Conceptualisation, Methodology, Validation, Investigation, Data Management, Visualisation, Formal Analysis, Software, Writing Original Draft, Writing Review & Editing. Although it becomes hard to remember after a certain size...

Manuscript reformatted to fit the style of this thesis.

Abstract

We investigate the boundary between chemotaxis driven by spatial estimation of gradients and chemotaxis driven by temporal estimation. While it is well known that spatial chemotaxis becomes disadvantageous for small organisms at high noise levels, it is unclear whether there is a discontinuous switch of optimal strategies or a continuous transition exists. Here, we employ deep reinforcement learning to study the possible integration of spatial and temporal information in an a priori unconstrained manner. We parameterize such a combined chemotactic policy by a recurrent neural network and evaluate it using a minimal theoretical model of a chemotactic cell. By comparing with constrained variants of the policy, we show that it converges to purely temporal and spatial strategies at small and large cell sizes, respectively. We find that the transition between the regimes is continuous, with the combined strategy outperforming in the transition region both the constrained variants as well as models that explicitly integrate spatial and temporal information. Finally, by utilizing the attribution method of integrated gradients, we show that the policy relies on a non-trivial combination of spatially and temporally derived gradient information in a ratio that varies dynamically during the chemotactic trajectories.

3.1 Introduction

Chemotaxis, the directed motion of organisms towards or away from chemical cues, is a fundamental biological mechanism that spans biological kingdoms. For instance, prokaryotes rely on chemotaxis to find nutrients, avoid toxins, or even optimize oxygen and pH levels by sensing molecular cues [101, 235, 236]. Single-celled eukaryotes show similar chemotactic traits [237], and countless biological processes in multicellular eukaryotes are supported by chemotaxis such as the fighting of bacterial infections by white blood cells, the positioning of stem cells during early embryonic development, and formation of multicellular structures in slime mold development [237–239]. Likewise, a hallmark of cancer metastasis is the chemotaxis of tumor cells towards blood vessels [240].

However, the ubiquity of chemotaxis in biology does not imply uniformity in the mechanisms that underlie the navigation. At the scale of microorganisms, the fluctuations of the molecules that bind to the cells' receptors are non-negligible and impose physical limits on the accuracy of the measurements and, thus, navigation. Chemotaxis is typically dichotomized into *spatial* and *temporal* strategies (Fig. 3.1c) [7, 106, 107]. Larger cells, usually eukaryotes, primarily exploit spatial sensing, harnessing their size to directly perceive chemical concentration gradients [13], whereas smaller cells like bacteria are known to adopt temporal sensing, detecting alterations in chemical concentrations temporally to deduce information on the gradient's direction, as the fluctuations across their body render spatial sensing useless. Cells are able to internally calculate temporal information due to the prolonged presence of the chemical signals within their body, which, in the absence of spatial cues, can provide information about the concentration changes experienced by the cell over time, especially as the cell moves through the envir-

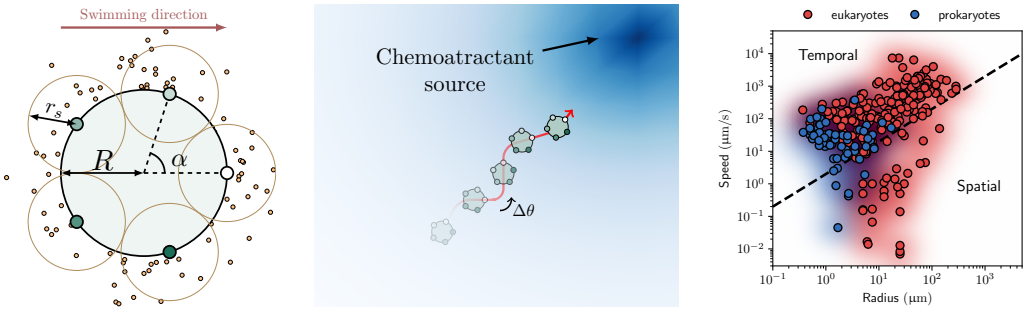
onment [51, 97, 241]. These differences in sensing mechanisms have direct consequences for the possible types of navigation decision processes.

This binary classification enables detailed analysis of the distinct forms of chemotaxis within each category. However, as the optimal strategy is dependent on continuously varying parameters such as the size and velocity of the organism as well as the chemoattractant concentration, it leaves the question of whether organisms can utilize an integration of both spatial and temporal sensing mechanisms in their chemotactic strategies [110], and whether such a combination would be preferential in intermediate ranges of these parameters. Interestingly, it has been shown that cells thought only to use spatial sensing also rely on temporal information during chemotaxis when given periodic waves of chemoattractant [242, 243]. Static temporal averaging of previous measurements has also been shown to reduce sensing noise on cells placed in shallow concentrations [13]; however, this does not take into account the effect of the motile cell itself reacting to the measurements. Previous work has proposed a more complex inclusion of both types of sensing to develop newer strategies without being able to outperform single sensing strategies [109], showcasing that efficient integration of both strategies is probably non-trivial.

Here, we employ deep reinforcement learning (DRL) to discover optimal chemotactic strategies that can combine spatial and temporal sensing. Previous work has successfully made use of DRL to find optimal strategies for self-propelled agents exploiting the flow in fluid environments [83] and for studying the tracking policies of flying insects relying on memory from noisy measurements to locate food or other insects [244]. Similarly, machine learning has been utilized for demonstrating the optimality of known chemotactic strategies [86, 245].

We propose a minimal chemotactic single-cell model and use modern policy optimization techniques [246] to identify the strategy that minimizes the time it takes the cell to reach a source of chemoattractant. The model cell is endowed with distinct sensors that enable spatial gradient estimation and is given an internal memory state that allows temporal information to be derived. Based on a combination of these inputs, the cell must modify its orientation, a mapping that we leave largely unconstrained by employing deep neural networks.

We demonstrate the existence of a better performant chemotactic strategy that non-trivially combines spatial and temporal sensing. Specifically, we pinpoint a range of cell sizes where a combined sensing strategy outperforms optimal single sensing strategies. We then concentrate our analysis on this interface, comparing it to analytical ones and offering both qualitative and quantitative insights on the internal dynamics of the optimal navigational policy.



(a) (b) (c)

Figure 3.1: Illustration of chemotaxis model components. (*a*) Representation of the model cell with five sensors surrounded by chemoattractant particles. Each sensor measures the number of particles M_i inside its sensing range r_s and transforms it as $m_i = log(M_i + 1)$. (*b*) Illustration of the simulation environment where the cell navigates towards the center of the chemoattractant source. (*c*) Phase space diagram of cell sizes and speeds showing the distribution of common unicellular prokaryotes and eukaryotes. The dashed line roughly indicates the binary division between temporal and spatial navigation strategies [7]. Data from Refs. [7, 247].

3.2 Methods

3.2.1 The simulation model

We study a exponentially decaying, two-dimensional distribution $C(\boldsymbol{x})$ of chemoattractant particles with a concentration peak at $\boldsymbol{x} = 0$,

$$C(\boldsymbol{x}) = C_0 \exp\left(-\lambda\,|\boldsymbol{x}|\right). \tag{3.1}$$

In Appendix A5, we further give examples of algebraic and Bessel function concentration profiles which can e.g. arise from decaying and diffusing particles emanating from a central static source,

$$D\,\nabla^2 C(\boldsymbol{x}) - \kappa\,C(\boldsymbol{x}) + \rho\,\delta(\boldsymbol{x}) = 0. \tag{3.2}$$

Here, D is the particle diffusion coefficient, and κ is a particle decay rate, which sets a length scale $\lambda = \sqrt{\kappa/D}$. We take $\lambda = 0.032\mu\text{m}^{-1}$ given typical values of $D = 100\mu\text{m}^2\,/\,\text{s}$ and $\kappa = 0.1\text{s}^{-1}$ [109], and study C_0 varying from $C_q = 16\mu\text{m}^{-2}$ to $10 \cdot C_q$, which sets the signal-to-noise ratio of the system and places the cell in the fundamental limit of sensing regime. ρ is the rate of particle release at $\boldsymbol{x} = 0$. Even though we assume a steady state profile for a static source, the cell will experience a change in concentration as it navigates the environment.

Our cell model consists of a circular disk of radius R, equipped with K sensors uniformly spaced around its surface (Fig. 3.1a), whose objective is to reach the source of the chemoattractant by controlling its direction of motion depending on the environmental measurements (Fig. 3.1b). The cell senses the environment through molecules binding to cell-surface receptors. Still, in the interest of keeping our model as simple as possible, we neglect the complex receptor dynamics of receptor binding and unbinding [10]. Thus,

we assume each of the K sensors to possess a detection area of radius $r_s = R\sin(\pi/K)$ such that the entire surface of the cell is covered. We fix $K = 5$ for all our experiments.

Our model cell never stops and moves forward at a constant speed, which we arbitrarily set to $v = 5\,\mu\text{m/s}$, with its trajectory orientation $\theta(t)$ being modified both by its own actions as well as due to rotational noise,

$$\mathrm{d}\theta = a_t\,\mathrm{d}t + \sqrt{2D_R}\,\mathrm{d}W. \tag{3.3}$$

Here, a_t is the output of the cell's navigational policy π, and the second term is Wiener noise with rotational diffusion coefficient D_R. Rotational diffusion forces the cells' navigation policies to react to the sensor signals at least on a time scale $1/D_R$ [99]. In our experiments, we use $D_r = 0.025/\text{s}$ as an average value on microorganisms of our sizes [7], and use a time-stepping of $\Delta t = 0.1\,\text{s}$ to solve the stochastic equations.

We model the cell receptors as *perfect instruments* [10], meaning that at each time step of our simulation, the sensors measure the exact number of molecules inside their sensor range instantly. This induces fluctuations in measurements with a signal-to-noise ratio that increases with concentration. Thus, nutrient-deprived environments with a low number of detected particles are noisy, and nutrient-rich environments are more deterministic.

We approximate the particle count within each sensor's area as a stochastic process sampling from a Poisson distribution. Exploiting the nearly constant particle density over the detection area, we use

$$\mathbb{E}(M_i) = \int_A C(\boldsymbol{x})\,\mathrm{d}A \approx C(d_i)\cdot\pi r_s^2 \tag{3.4}$$

$$M_i \sim \text{Poisson}(\mathbb{E}(M_i)), \tag{3.5}$$

d_i is the radial distance of the receptor center to the source of the chemoattractant.

Simulations are initialized at random distances d_0 from the source with random orientations θ_0 and crucially with a rate of particle release ρ, which we sample in the range ρ_0 and $10\cdot\rho_0$. These random initializations ensure that the cell agents cannot overtrain to specific molecule counts and specific trajectories but rather need to generalize across noise levels and become adaptable to varying concentration profiles.

Finally, we do biologically inspired preprocessing of the receptor input by transforming according to the Weber-Fechner law [248],

$$m_i = \log(M_i + 1)\,. \tag{3.6}$$

While this could have been learned directly from the data, it conveniently brings the neural network input to a tightly constrained domain that is more suitable for DRL, and also means that noise in m_i decreases not just relative to the signal but also in absolute numbers as ρ increases.

3.2.2 The policy

The internal mechanisms of a chemotactic cell involve a complex set of biochemical spatio-temporal reactions. Here, we do not model these reactions explicitly, but instead

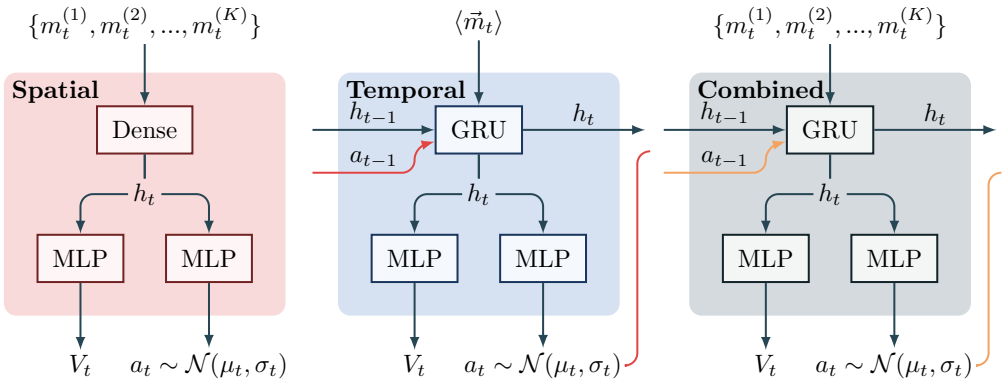


Figure 3.2: Our three neural network policies output the cell's action based on the measurements and hidden states. The combined policy has access to the individual measurement of its sensors and has a hidden state used in a recurrent neural network layer, whereas spatial and temporal only have one of these features. *Dense*: a linear NN layer connecting all inputs with all outputs. *MLP*: Multilayer perceptron, a sequence of dense layers with non-linear activations. *GRU*: Gated recurrent unit, a simple form of recurrent neural network module, which combines a hidden state with new input. The policy output of the model is both a mean value μ_t and a standard deviation σ_t, which defines a normal distribution from which an action a_t is sampled. In our experiments, $\sigma_t \to 0$ at the end of training results in deterministic policies

model directly an input-output approximator, *the cell policy* π. This policy maps an internal state s_t, which in the simplest case could just be the vector of instantaneous measurements, to an action a_t. We parameterize the function using artificial neural networks (ANN) to minimize expressive restrictions on the learned policy.

To estimate the cell policy, we assume that it is an optimizer of efficient chemotaxis, which we define as minimizing the time it takes to reach a certain distance from the source, as the faster a cell reaches a source, the less competition with other cells it will encounter. More precisely, at the end of a simulation, we calculate a *reward* by

$$\mathcal{R} = \frac{t_{max} - \tau}{t_{max}} + \max(-1, \frac{\delta - d}{d_0 - \delta}), \tag{3.7}$$

where d_0 and d are the initial and final distance to the source (which will be $d = \delta$ if the source has been reached), respectively, and τ is the simulation duration. In the case of not reaching the source, $\tau = t_{max}$. We include distance information as part of the reward reshaping technique to still gather information when the cell is not able to reach the source before t_{max}. The first term is a normalized reward for getting to the source fast, and the second is a bootstrapping reward that punishes cells that do not reach within the required distance δ of the source. The reward is normalized between $[-1, 1]$, as is convention in reinforcement learning. We perform episodic rewards instead of rewarding every action as we found the combined episodic rewards to converge to better solutions. Simulations terminate when the cell has reached δ distance to the source or the simulation time has exceeded $t_{\max}$, thus only one term of the reward expression is nonzero at the end of the episode; with the distance reward dominating early in training and the time reward at the end of training.

To find the optimal ANN policy, we employ Proximal Policy Optimization (PPO) [246], which adapts the policy π in order to maximize the average reward. We study three variants of the agents (Fig. 3.2): one policy we restrict to act purely on instantaneous *spatial* information. This is enforced by simply designing the neural network to be a pure feedforward network — from measurements $\{m_t^{(1)}, m_t^{(2)}, \cdots, m_t^{(K)}\}$ to output a_t. Likewise, we design a purely *temporal* network, which does not receive spatial information but rather the average of all receptors $\langle m_t \rangle$. Instead, this agent must rely on memory to provide temporal information on the particle gradients. This is achieved by introducing a recurrent layer into the policy neural network, which emulates the biochemical memory of real cells. Finally, we study a *combined* agent, which has access to both spatially resolved measurements and has memory that can be used to derive temporal information. This agent can execute pure spatial and pure temporal strategies but can furthermore act on any combination of this information. Network details are given in SI.

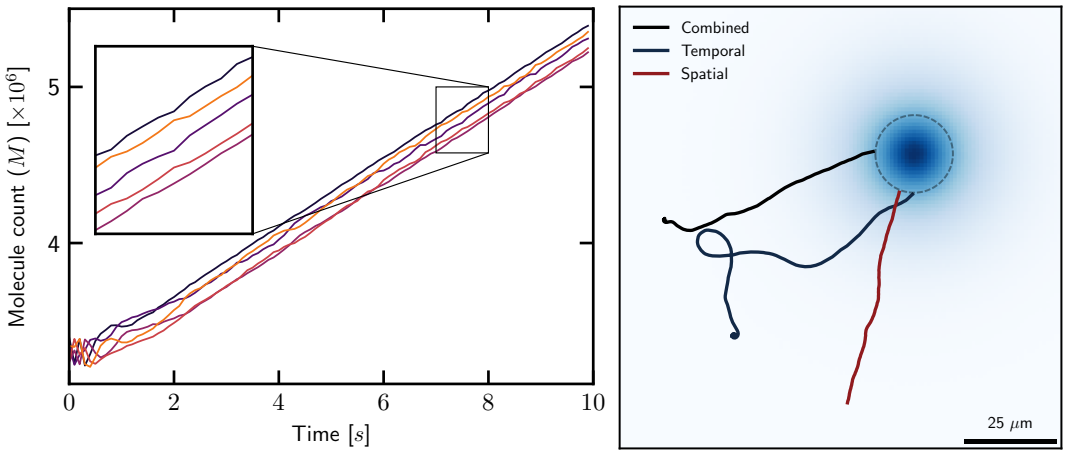
Our networks also output an estimate of the final reward V_t (Fig. 3.2), which the PPO algorithms use to speed up convergence but which does not influence the policy once trained. Further, as the nature of PPO's exploration strategy adds noise to the policy output, we also recurrently feed the cell's action back into the temporal policies, which aids the training in reaching a deterministic strategy without hindering stochastic exploration.

3.3 Results

3.3.1 Optimizing for noise-robust strategies

Our deep reinforcement learning approach is designed principally to work at all noise levels. In nutrient-rich environments, where the input to the agents is not corrupted by noise, our DRL framework converges quickly to effective temporal and spatial strategies. Resulting trajectories in these environments are close to deterministic as the noise from measurements gets reduced, and fewer mistakes in orientation corrections tend to occur. In those scenarios, spatial-based gradient estimation is effective in directly locating the source of chemoattractant, and noise due to rotational diffusion does not pose a challenge for the cell, which only needs to follow the strength of the sensors (Fig. 3.3a). Likewise, the optimal temporal sensing strategy at high concentrations is easily understood as it continuously measures the change in concentration and increases the turn when the concentration starts diminishing. As the temporal strategy contains no information about the sensors' positions, it has to spontaneously break its rotational symmetry, which is exemplified in the resulting left-turning shown in Fig. 3.3b, resembling, e.g., the chirality of sperm chemotaxis trajectories [249].

In contrast, in the low concentration limit, the input to the cell receptors is extremely noisy (Fig. 3.4a), and the identification of optimal strategies becomes less clear. Yet, our DRL approach is able to identify working strategies both using purely spatial and purely temporal sensing mechanisms (Fig. 3.4b). Qualitatively, we note that the identified low-concentration temporal strategy behaves very robustly against noise, as its trajectory



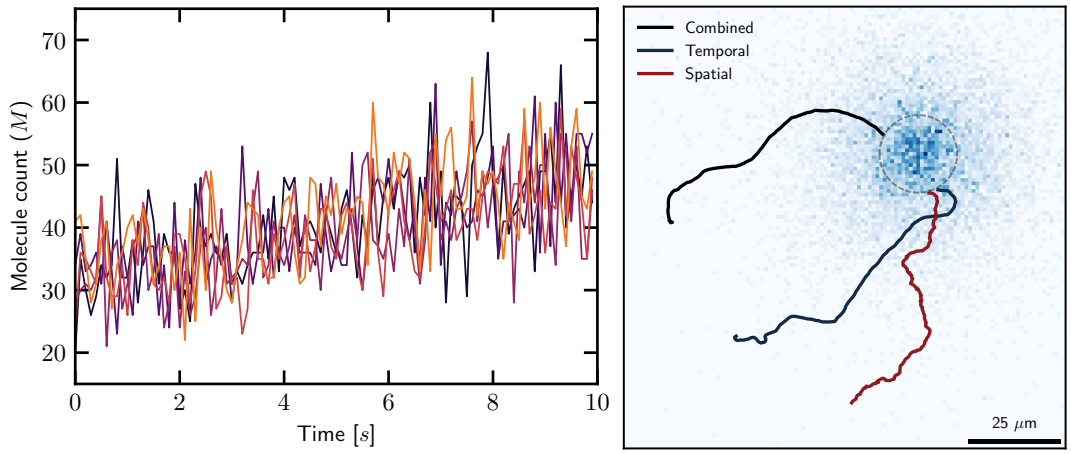
(a) (b)

Figure 3.3: Chemotaxis in nutrient-rich media ($C_0/C_q = 10^4$) (a) Measurement values of each sensor of the cell. Each color represents one of the $K = 5$ sensors used in the trajectories. The measurements correspond to those of the Combined cell. (b) Example trajectories of each variant found strategies at $R = 2\mu$m.

remains smooth despite its stochastic input. This can be interpreted as *low reactivity*, which also showcases itself as the temporal strategy only slowly adapts its trajectory as it nears the source. In comparison, the spatial strategy is very reactive, and while this makes it susceptible to the stochastic input, it enables it to quickly adapt its orientation once it nears the source and the concentration is relatively high. Finally, we observe the first hint that the combined strategy can outperform the two: it shows low reactivity when far from the source and high reactivity once in its proximity. Low reactivity in shallow and noisy concentration profiles allows the cell to rely on persistence to avoid getting trapped, increasing the likelihood of reaching a region where measurements convey more information. This strategy has also been observed in situations where cells are less likely to instantly turn around, enhancing their chemotactic response by reducing their reactivity [98].

While deterministic policies are fast to identify, the information that reinforces policies in the low concentration limit is much more stochastic, making the optimization process harder. To enable learning in this very noisy regime, our reinforcement learning steps rely on averaging the result of thousands of runs and require millions of simulations to converge to a solution (see Table 3.2). To make this feasible, we developed a custom end-to-end RL implementation that runs exclusively on GPUs (see Code Availability).

We note that DRL is not guaranteed to find the globally optimal policy. However, we find that independent runs of the DRL training procedure result in the same policies, which hints that the obtained local optima could be global.



(a) (b)

Figure 3.4: Chemotaxis in nutrient-depleted media ($C_0/C_q = 1$) (a) Measurement values of each sensor of the cell. Each color represents one of the $K = 5$ sensors used in the trajectories. The measurements correspond to those of the Combined cell. (b) Example trajectories of each variant found strategies at $R = 2\mu$m.

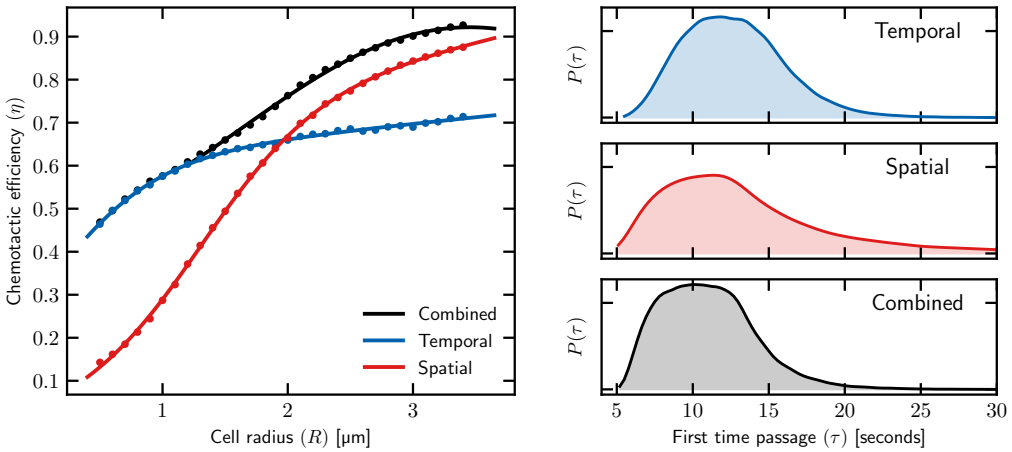
3.3.2 Smooth transition between a temporal and a spatial strategy

For evaluation, we define a strategy's chemotactic efficiency η by how fast the cell reaches the source compared to the minimal time a cell of speed v would take to reach it from the same initial position (note that this is independent of t_{max} which was used for training). Thus, the efficiency of a strategy is given by

$$\eta = \left\langle \frac{d_0 - \delta}{v \cdot \tau} \right\rangle , \tag{3.8}$$

where τ is the time it takes the cell to reach the source threshold distance δ and d_0 is the initial distance to the source, and the average is taken over all realizations.

We train our three variants, spatial (S), temporal (T), and combined (C), on the same simulation parameters at different cell sizes and proceed to calculate their efficiencies (Fig. 3.5a). At small sizes, where the positional sensor information becomes indistinguishable due to the noise, both T and C policies show the same performance. This is in accordance with previous studies showing that small cells are incapable of sensing gradients along their own body due to the fluctuations in measurements [110]. Nevertheless, as the cell size increases, C starts to outperform T, indicating that the tiny amount of available gradient information, as observed by the poor performance of S, can somehow be integrated into a temporally dominated strategy to improve its performance. At large cell sizes, S dominates T, and while a gap still remains between S and C at large R, it shows convergence towards the same strategy. Thus, the sensors need not rely heavily on old measurements to estimate the gradient accurately at the largest scales. At intermediate cell sizes, we find that the optimal strategy is not purely spatial



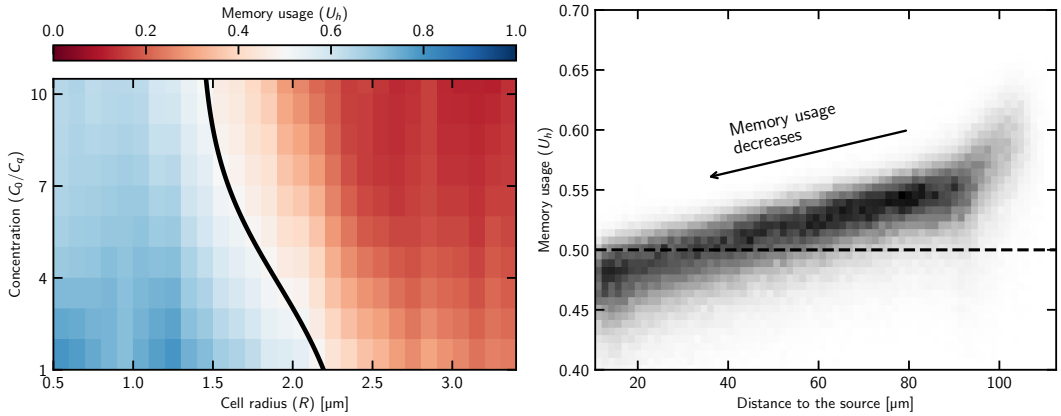
(a) (b)

Figure 3.5: Comparison of chemotactic performance between the combined policy and the control variants. (*a*) Chemotactic efficiency of each variant on reaching the source as a function of cell size. Each value is the result of training and evaluating the policies at that cell radius for sampled values of C_0. The average efficiency is evaluated on 2^{16} independent runs. A "blind" agent obtains efficiency $\eta \approx 0.02$. (*b*) Distribution of arrival times to the source of the three cell variants at $R = 2\mu$m. All evaluations use sampled concentrations.

or temporal. In detail, we observe a smooth transition between strategies, indicating that there is a continuous integration of information stemming from spatial input and memory. Despite being dominated by noise, as illustrated in Fig. 3.4a, C is capable of taking advantage of the measurement differences between the different receptors on the cell surface to improve its efficiency. To explore this integration, we now focus on this intermediate region where both S and T perform similarly yet are outperformed by C, at $R \approx 2\,\mu$m.

Inspecting the distribution of arrival times as shown in Fig. 3.5b for $R = 2\mu$m, we observe a clear difference in skewness between T and S. The distribution of arrival times in S has long tails since cells that start far away from the source are experiencing very low concentrations of molecules, which disproportionally affect the spatial strategy. In contrast, T shows very few cells that reach the source quickly, as this strategy relies on building memory. Interestingly, the cells that use C are both fast and do not get trapped, having both benefits of the other variants.

To evaluate the optimality of the found strategy C, we compare it against commonly proposed strategies that use memory kernels to integrate temporal information into spatial strategies. Likewise, we explore a switching strategy in which cells start by using the noise-robust T strategy and later switch to using the reactive S strategy at a set threshold. This incorporates the advantages of each variant as shown in Fig 3.5b. In all cases, we find that the RL learned strategy outcompetes these simpler explicit strategies (see Appendix A1).



(a) (b)

Figure 3.6: Reliance on temporal information, through the use of memory, of the action from the Combined strategy. (*a*) Average memory usage contribution to the steering output during the simulation runs at different sizes and concentration levels C_0. The line indicates $U_h \approx 0.5$, i.e. the transition from a memory-dominated strategy to a more reactive sensing-based policy. (*b*) Distribution of memory usage U_h values during individual trajectories, evaluated at different distances to the source. $R = 2\,\mu\text{m}$.

3.3.3 Integrating temporal and spatial information

Having established that C can integrate spatial and temporal information to outcompete both T and S, we move on to studying the internals of C directly. The policy π_{C} is a highly non-linear, recursive function which we have parameterized by deep learning neural networks — this at the cost of lack of interpretability. Nonetheless, numerous techniques have been developed to gain insight into the internals of a trained neural network, for instance, by estimating the importance of the input variables. One of the most elegant techniques to study this attribution problem is the method of integrated gradients (IG) [91], which calculates the importance of feature x_i as

$$\mathcal{I}_i = (x_i - x_i') \int_0^1 \frac{\partial \pi(x' + \alpha(x - x'))}{\partial x_i} \, \mathrm{d}\alpha, \tag{3.9}$$

where x' is a baseline, which we here simply take to be no input $x' = 0$. IG is *sensitive* meaning $\mathcal{I}_i$ is non-zero if and only if x_i contribute to the output, and satisfies *completeness* such that the attributions sum to the output, i.e. $a_t = \pi(x) = \pi(0) + \sum_i \mathcal{I}_i$.

We use IG to understand how the cell relies on previous measurements transmitted to it by the hidden state h_{t-1}, compared to current measurements m_t from the receptors. We define U_h as the relative importance of memory,

$$U_h = \frac{\sum_{i \in h} |\mathcal{I}_i|}{\sum_j |\mathcal{I}_j|}. \tag{3.10}$$

where the contribution of the hidden state inputs is normalized by the sum of contributions of hidden state components (memory) and instant molecule measurements.

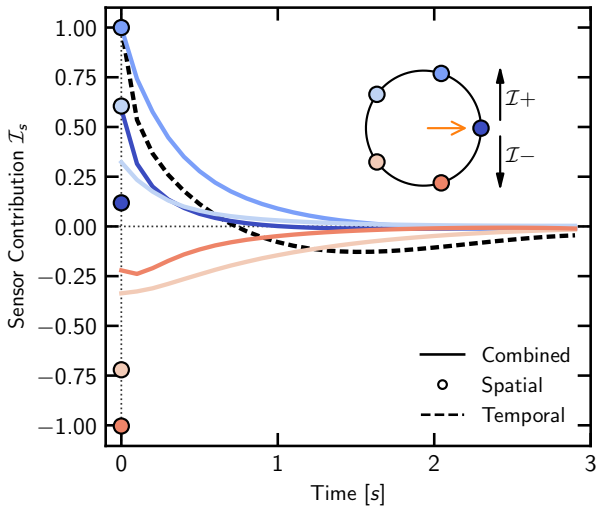


Figure 3.7: Temporal and spatial sensor influence on cell navigation. Contribution of each sensor from past time measurements to the current action. The three variants at $R = 2\,\mu\text{m}$ are shown, with data colored by sensor position as indicated in the cell diagram. For the temporal variant (dashed), only one sensor is shown, as all have the same profile as per the designed symmetry. The red arrow indicates the swimming direction. Curves are obtained by averaging over $\sim 10^5$ trajectories with initial conditions sampled similarly to previous plots.

Memory usage is well defined between 1, i.e., the cell only relies on memory for its decision without considering the current measures, and 0, where the cell disregards the memory information and instead only depends on instant measurements. Thus, a $U_h > 0.5$ value indicates that the hidden state contributes more to the output than the current values. Note that the definition sums over contributions from all hidden states and all measurements and is thus virtually independent of, e.g., the number of hidden states. Here, we also ignore the previous action a_{t-1} contribution, as it is negligible compared to the memory and measurements contributions for a converged policy, and it is only during training that it is essential.

Figure 3.6a shows how average memory usage U_h changes as a function of cell size and chemoattractant concentration. In accordance with previous conclusions, we observe a smooth transition of decreasing memory contribution as the cell gets larger. This transition occurs at smaller sizes the higher the concentration. Interestingly, when evaluating U_h within a single environment (Fig. 3.6b), we observe a decrease in memory usage as the cell approaches the source. Thus, the cell is adapting between temporally and spatially dominated strategies during a single trajectory, akin to a continuous version of the discrete switching strategy just considered.

Although the input to the neural network policy π is the current measurements m_t and the hidden state h_{t-1}, the output a_t can also be considered a function of all previous measurements $\{m_1, m_2, \cdots, m_t\}$, being processed recursively by a sequence of hidden states, i.e. $a_t = \pi(m_t, h_{t-1}) = \pi(m_t, m_{t-1}, m_{t-2}, \cdots, m_0)$. Applying Eq. (3.9) in this

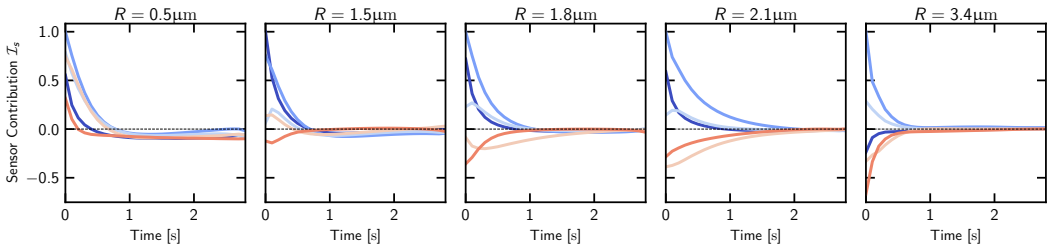


Figure 3.8: Temporal and spatial sensor influence on cell navigation. Sensor contributes to the combined policy for different cell sizes. Color code is the same as in Fig. 3.7.

formulation, we can attribute importance individually to all previous measurements on the current output.

Figure 3.7 shows the IG attributions of measurements for the purely spatial, the purely temporal, and the combined strategy at $R = 2\,\mu\text{m}$. As the cell diagram indicates, a positive IG value translates into a contribution for a positive reorientation and a negative value vice-versa. For our model, this translates positive and negative contributions to pushing the cell to turn left or right, respectively. On a pure spatial strategy, the sensors work in opposition, and previous measurements obviously do not contribute. In contrast, all sensors contribute the same on a pure temporal strategy, but previous measurements oppose current measurements. Curiously, the shape of the contributions highly resembles the bi-loped shape of the chemotactic memory kernel measured experimentally on the impulse responses of *E. coli* bacteria [99].

Similar to the spatial strategy, the combined strategy shows sensors working in opposition, but the left-right symmetry is broken and compensated by temporal variance, with one side dominating early and the other side contributing late. This sensor signature of the combined strategy makes explicit the non-trivial combination of information it is utilizing, and while these curves are merely IG components, they are indicative of a non-linear combination of information that asymmetrically merges spatial and temporal processing (Fig. 3.7). We observe a transition from temporal towards spatial information processing by looking at how measurements are integrated into the combined policy for different sizes (Fig. 3.8). This similarity is clearly observed when comparing trajectories of the purely temporal and pure spatial policies with the combined one at the respective extreme cell sizes.

3.4 Discussion

In this study, we have explored the theoretical possibilities in chemotaxis that arise when traditional limitations are relaxed, i.e. when spatial and temporal strategies are not studied in isolation. Our findings show that the borders of binary classifications of chemotaxis strategies can be blurred by suitable integration of spatial and temporal information. In particular, we have shown that for cells with the ability to sense across their bodies as well as having memory access, there is a navigation strategy that out-

performs those with only one sensing ability. Therefore, our results show that relying solely on unimodal chemotactic strategies to evaluate microorganisms' efficiencies may indicate only a lower bound of their actual chemotaxis performance, and integrating both sensing information instead may yield more accurate estimates. Without imposing any constraints on the policy, we have seen the optimal solution to converge to known policies (purely spatial, purely temporal) in the limits where it is known that one sensing mechanism clearly provides faster information on the chemical gradient. Here, we explored this as a function of cell size and found that for large cells, the emerged combined strategy converges to relying only on spatial information, whereas for small microorganisms, the gradient information is strictly obtained on temporal differences. In the intermediate range, we found no sudden switch in strategy, but instead, the transition between them is continuous and smooth, where information is slowly being integrated by the cell into its decision process. We expect our mapping of optimal strategies during this transition may provide some guidance to understand how larger, more complex cells exploit temporal information to improve their sensing capabilities.

Our general perspective on chemotaxis is achieved by employing artificial neural networks and optimizing these by reinforcement learning. The drawback to this is that the obtained strategies are difficult to interpret. Yet, by comparing analytical strategies and employing integrated gradients to study feature attribution, we find that the optimal strategy that employs both spatial and temporal information is not a simple combination of known strategies, nor is its integration of information types trivial. Our analysis reveals that memory usage varies with cell size and concentration and changes dynamically throughout trajectories. This is akin to the well-known phenomenon that cells adapt their measurement sensitivity to local concentration [250], but here, we find that in an optimal setting, the navigation strategy itself must also dynamically adapt.

Using DRL to study chemotaxis in the noise-dominated regime is computationally challenging, as it requires a large number of simulations that must dynamically be run during training. Our custom approach runs simulations and training on GPU, avoiding slow system-to-device transfers. Here, we have employed this approach to study a simple chemotactic agent in two dimensions. An interesting avenue for future research is the move to three dimensions, where the space of possible strategies is qualitatively different. Likewise, it could be interesting to consider the consequences of a non-static source of chemoattractant or heterogeneous environments and discover their effect on a combined chemotactic policy. Similarly, it is of interest to extend our minimal cell model to specificities of particular organisms, such as a thorough modeling receptor dynamics [98], the inclusion of stochastic tumbles of peritrichously flagellated bacteria [241], or more complex behaviors as the ones seen in *C. elegans* [251]. Concurrent to our work, agents capable of both spatial and temporal sensing mechanisms have been investigated using an information theoretical approach [108], deriving complementary insights.

Appendix

A1 Comparison with interpretable models

In this section, we evaluate the chemotactic efficiency of the trained policies by comparing them to simpler strategies that are straightforward in explaining the integration of memory and spatial gradient sensing.

We begin by defining a naive spatial policy $\tilde{\pi}$ where the steering of the swimming orientation is directly dictated by the strength of the receptor measurements. The naive optimal reorientation is given by

$$\phi = \operatorname{atan2}\left(\frac{\sin(\boldsymbol{\alpha}) \cdot \boldsymbol{\omega}}{\cos(\boldsymbol{\alpha}) \cdot \boldsymbol{\omega}}\right), \tag{3.11}$$

where $\boldsymbol{\omega}$ are the contributions of each receptor to the decision and $\boldsymbol{\alpha}$ are the angles of the receptor position on the cell's surface with respect to the swimming direction (Fig. 3.1a). This naive strategy is very susceptible to fluctuations in measurements and can sometimes be improved by restricting the reorientations to a certain ε. Thus, we consider policies of the form

$$a_t = \tilde{\pi}(s_t) = \frac{1}{\Delta t} \cdot \begin{cases} -\varepsilon & \text{if } \phi \leq -\varepsilon \\ \varepsilon & \text{if } \phi \geq \varepsilon \\ \phi & \text{otherwise.} \end{cases} \tag{3.12}$$

Integrating measurements over time reduces the fluctuations in concentration measurements, as has also been shown experimentally [13]. Thus, we explore the possibility of cells relying on the average of previous measurements to set the change in orientation. The contribution of each sensor is then averaged by previous measurements as

$$\omega_t^{(i)} = \int_0^\infty \kappa(t')\, \hat{m}^{(i)}(t - t')\, \mathrm{d}t'. \tag{3.13}$$

Here, $\hat{m}(t)$ are corrected measurements at time t. Directly using $m(t)$ completely ruins performance, as every time an action is performed, the information of previous measurements is no longer aligned with the cell orientation. To obtain optimal strategies, we use $\hat{m}(t)$, which is corrected by the action taken a_t, and thus only suffers from information decay due to rotational diffusion.

We begin by studying a uniform distribution, such as

$$\kappa(t) = \begin{cases} \frac{1}{T} & \text{for } t \leq T \\ 0 & \text{otherwise} \end{cases} \tag{3.14}$$

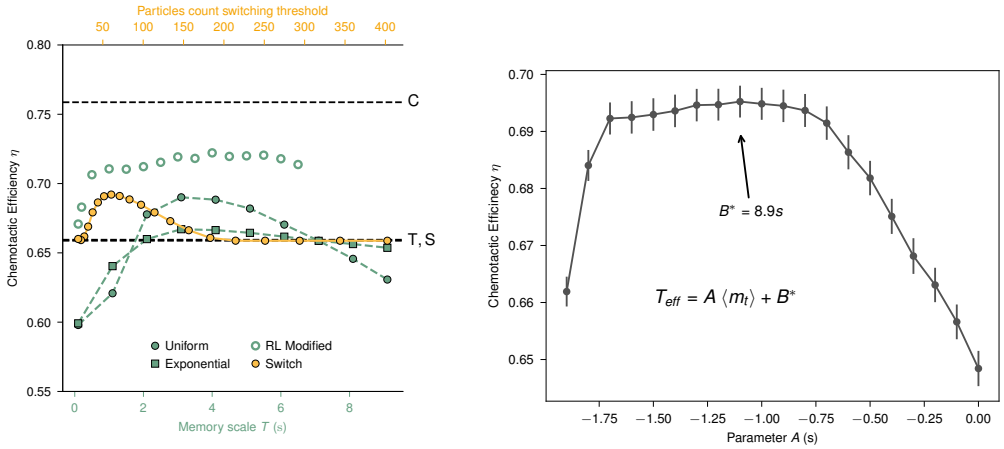
where all previous measurements contribute the same up to T. Moreover, we consider the use of an exponentially decaying kernel

$$\kappa(t) = \frac{1}{T} e^{-t/T}, \tag{3.15}$$

which gives more weight to newer measurements.

As seen in Fig. 3.9a, the chemotactic efficiency of these models outperforms S and T when some rudimentary use of memory is allowed. We note that each reported value on the analytical strategies is evaluated with different ε, and only the best-performing one is shown. Nevertheless, we observe that a large memory timescale becomes counterproductive as the movement of the cell makes previous measurements irrelevant and only contributes noise to the decision. Despite the gain in efficiency, the optimal timescale for the proposed models is far from reaching the chemotactic efficiency of C.

We note that as $T \to 0$, S outperforms the explicit models. This can be explained by the freedom of S to dynamically control a non-linear equivalent of ε depending on the measurements. With this in mind, we investigate a new RL agent using the same neural network as S, but whose input is given by Eq. (3.13). Thus the integration of memory is fully controlled, but any non-linear action can be taken based on this input. We note that this again requires correcting previous inputs and special attention is given to the early parts of trajectories, such that the policy only averages over known measurements. Fig. 3.9a shows that this indeed outperforms S and T, but cannot reach the performance of C. This suggests that C is not just combining a *temporal average* with a *spatial strategy* but is also using elements of a *temporal strategy*.



(a) (b)

Figure 3.9: (*a*) Chemotactic efficiency of proposed explicit policies compared to the neural network policies found using reinforcement learning, at $R = 2\,\mu\text{m}$. Green points are for policies that integrate measurements over time (lower axis), whereas orange points correspond to the policy achieved by switching between temporal and spatial strategies at a certain concentration threshold (upper axis). (*b*) Chemotactic efficiency of a policy that adjusts the memory time scale according to a linear dependency with the average strength of the measurements $T = A\langle m_t \rangle + B$. The efficiency is shown as a function of A, and B and ε are the optimal values for that A. The simulation parameters are the same as in (*a*).

Spermatozoa have recently been shown to exhibit a biphasic chemotactic strategy, in which there is a concentration-dependent switch between hyperactive phases, characterized by random changes in orientation, and more well-known chiral motion [252].

Presently, a switch between a temporal and a spatial strategy could achieve the best of the distinct time distributions of T and S in Fig. 3.5b. We implement this by setting a cutoff particle count at which we switch from T to S. As a function of this threshold, an increase in chemotactic efficiency is observed, as shown in Fig. 3.9a, but this also does not reach the efficiency achieved by C. Nevertheless, the increase in efficiency does suggest that the contribution of temporal and spatial may change dynamically with the concentrations.

Finally, we explore the possibility of designing an agent where the effective memory scale T is linearly dependent on the measurement concentration, as suggested in FIG.4B, such that

$$T = A\,\langle m_t \rangle + B \tag{3.16}$$

We evaluate for different parameters of A, B, and ε on a uniform kernel. Fig. 3.9b shows the chemotactic efficiency at different parameters A, with the best performant $B^*(A)$ and $\varepsilon^*(A)$.

The performance of this model is similar to that of a fixed uniform kernel. While the study of integrated gradients shows the amount of memory used, it does not reveal how this memory is used. In particular, here, we find that a simple uniform kernel is far from enough to reach optimal behavior.

A2 Simulation Parameters

Table 3.1 lists the default parameters used in the simulation environment for training and evaluation. These parameters govern the behavior of agents and the properties of the environment, including the chemoattractant field and cellular dynamics.

Description	Symbol	Value	Unit
Number of receptors	K	5	–
Cell speed	v	5	µm/s
Rotational diffusion coefficient	D_R	0.025	s^{-1}
Chemoattractant diffusion coefficient	D	100	$\mu m^2/s$
Chemoattractant decay rate	κ	0.1	1/s
Time step	Δt	0.1	s
Lower limit concentration levels	C_q	16	$1/\mu m^2$
Concentration levels	C_0	$\sim U(C_q, 10\,C_q)$	–
Threshold distance to the source (10^{th} percentile of N)	δ	$-\log(0.9)/\sqrt{\kappa/D}$	µm
Initial distance	d_0	$-\log(\sim U(0.3, 0.7))/\sqrt{\kappa/D}$	µm
Initial orientation	θ_0	$\sim U(-\pi, \pi)$	rad

Table 3.1: Default parameters used on the simulation of the environment during training and evaluation runs, if not stated otherwise.

A3 PPO training hyperparameters

Table 3.2 summarizes the hyperparameters used in training the Proximal Policy Optimization (PPO) algorithm for our implementation. These parameters were chosen to balance learning efficiency and stability while leveraging parallelized environments for large-scale simulations.

Description	Symbol	Value
Number of parallel training environments	N_{envs}	4096
Total number of simulation steps	N_{steps}	10^{10}
Simulation duration steps	$t_{\max}/\Delta t$	256
Maximum gradient norm clipping		0.5
Learning rate	η	$3 \cdot 10^{-4}$
Number of epochs	N_{epoch}	8
Number of mini batches	N_{batches}	8
Epsilon clipping	ε	0.2
Entropy coefficient	$\hat{\sigma}_s$	0.01
Critic coefficient	$\hat{\sigma}_c$	0.5
Advantage Discount	γ	1.0
Generalized advantage coefficient	λ	1.0
Number of hidden layers	K	2
Hidden layers size	L	64
Number of hidden state cells	M	25
Minimum variance in output	$\sigma_{\min}$	0.05
Maximum variance in output	$\sigma_{\max}$	1.0

Table 3.2: Parameters used during training to train using our Proximal Policy Optimization implementation

A4 Approximation of the concentration profile

Figure 3.10 compares the Modified Bessel function of the second kind, which is the solution to the 2D diffusion equation, with an exponential gradient used for simplicity. While the Bessel function provides the exact solution, the exponential gradient serves as an effective approximation in the regions of interest, offering a computationally efficient alternative.

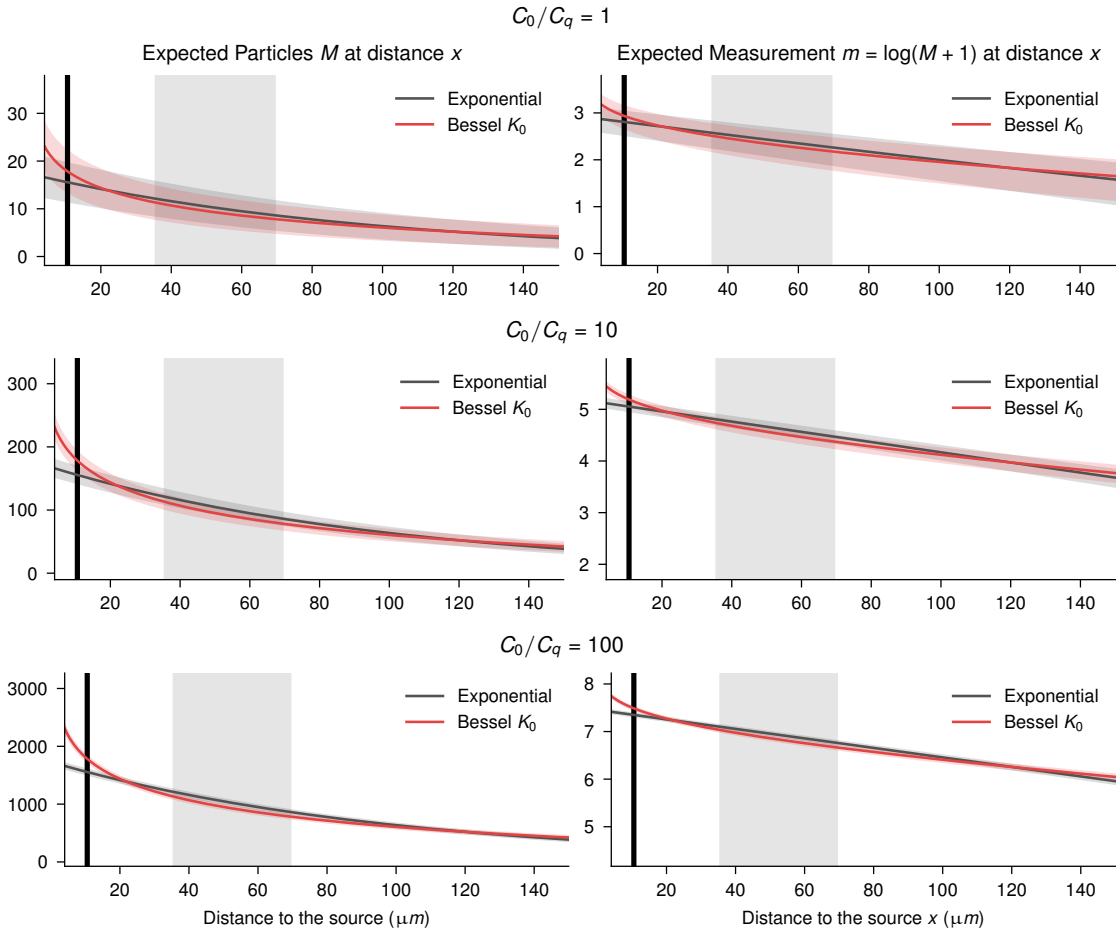
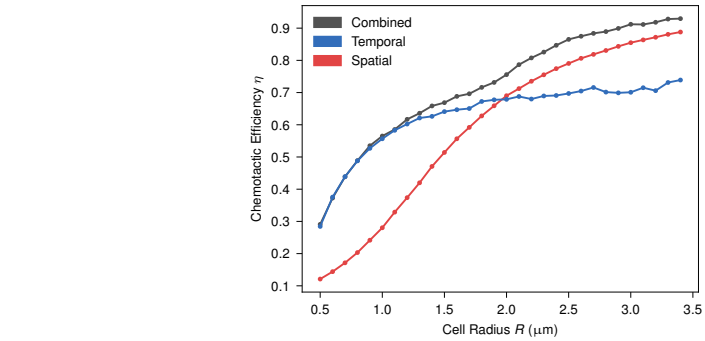


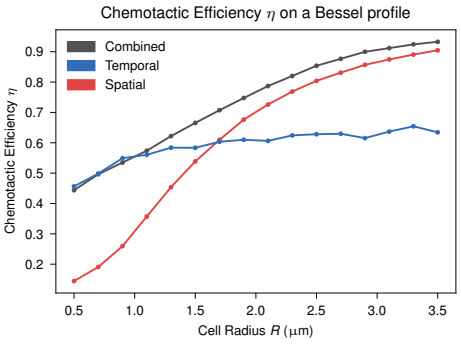
Figure 3.10: Concentration profiles at different amounts of concentration levels C_0/C_q. The standard deviation of the expected values at different distances from the source is also shown to indicate the noise in the measurements. The thick black line represents δ, the closest the cells will be from the source, and the gray region indicates the initial distance region where the cell may start an episode. Both the Modified Bessel of second kind function and the exponential are plotted to showcase their similarity in the region of interest.

A5 Universality of the results

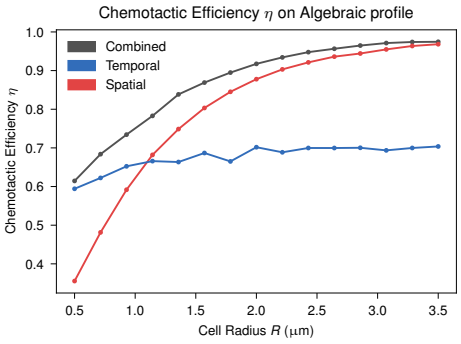
Figure 3.11a-3.11c demonstrates the universality of the observed transition between temporal and spatial chemotactic strategies across different gradient profiles and under varying assumptions about rotational diffusion. The crossing consistently occurs, and the combined strategy always outperforms the individual strategies.



(a)



(b)



(c)

Figure 3.11: Chemotactic efficiency as a function of cell size. (a) We model the rotational diffusion of the agent as $D_{rot} = \left(\frac{a}{R}\right)^3$, with $a = 0.315\ \mu m\ s^{-4}$. Similar results (transition) are observed, with the main difference being at smaller sizes where the D_{rot} worsens the efficiency of all strategies. (b) Chemotactic efficiency while using a concentration profile $c(x) \propto K_0(\lambda x)$. (c) Chemotactic efficiency with and algebraic concentration profile, i.e. $c(x) \propto 1/x$.

Chapter 4

On amoeba decision-making

The following chapter contains the paper:

Persistent pseudopod splitting is an effective chemotaxis strategy in shallow gradients

Albert Alonso[1], *Julius B. Kirkegaard*[1,2] *and Robert G. Endres*[3]

1. Niels Bohr Institute, University of Copenhagen, 2100, Copenhagen, Denmark.
2. Department of Computer Science, University of Copenhagen, 2100, Copenhagen, Denmark.
3. Department of Life Sciences and Centre for Integrative Systems Biology and Bioinformatics, Imperial College, London, United Kingdom

Under review in: *PNAS* (Proceedings of the National Academy of Sciences)
Pre-print server: https://arxiv.org/abs/2409.09342

Author Contributions:
Conceptualisation, Methodology, Validation, Investigation, Data Management, Visualisation, Formal Analysis, Software, Writing Original Draft, Writing Review & Editing. I reckon British me did a bloody quick job on this one, considering my living situation...

Manuscript reformatted to fit the style of this thesis.

Abstract

Single-cell organisms and various cell types use a range of motility modes when following a chemical gradient, but it is unclear which mode is best suited for different gradients. Here, we model directional decision-making in chemotactic amoeboid cells as a stimulus-dependent actin recruitment contest. Pseudopods extending from the cell body compete for a finite actin pool to push the cell in their direction until one pseudopod wins and determines the direction of movement. Our minimal model provides a quantitative understanding of the strategies cells use to reach the physical limit of accurate chemotaxis, aligning with data without explicit gradient sensing or cellular memory for persistence. To generalize our model, we employ reinforcement learning optimization to study the effect of pseudopod suppression, a simple but effective cellular algorithm by which cells can suppress possible directions of movement. Different pseudopod-based chemotaxis strategies emerge naturally depending on the environment and its dynamics. For instance, in static gradients, cells can react faster at the cost of pseudopod accuracy, which is particularly useful in noisy, shallow gradients where it paradoxically increases chemotactic accuracy. In contrast, in dynamics gradients, cells form *de novo* pseudopods. Overall, our work demonstrates mechanical intelligence for high chemotaxis performance with minimal cellular regulation.

4.1 Introduction

Mechanical intelligence is widespread in nature, by which information processing is deeply embedded in the architecture of living systems [7, 253]. For instance, the underlying mechanisms by which cells perform chemotaxis, the directed movement of an organism along a chemical concentration gradient during microbial pathogenesis, wound healing, and immune response, remains a subject of intensive research [2, 95, 131, 254]. Particularly relevant is the understanding of the tight coupling between sensory cues and cell locomotion mechanisms, as they provide insights into effective navigation methods at the microscopic scale for cell sensing at fundamental physical limits [255]. Here, we focus on studying the role of pseudopod formation as a cellular decision-making mechanism, which represents an important yet not fully understood aspect of cellular navigation [118, 121].

Amoeboid locomotion is characterized by the extension of pseudopods, temporary protrusions that allow the cell to explore its environment and move directionally in response to chemical cues [7]. Experimental evidence has shown pseudopod formation to occur at higher rates in shallow gradients and that they are more pronounced [112]. In particular, this phenomenon may constitute a way to reach the fundamental physical limit by minimizing the interference of sensing by movement [118]. The ultimate limit is reached by a cell that senses only previously undetected ligand molecules to gain new information, effectively corresponding to a ligand-absorbing cell [118]. Despite the observed importance of amoeboid cell migration, a comprehensive theoretical understanding of pseudopod splitting and its strategic role in accurate chemotaxis is still lacking.

Traditionally, chemotaxis is approached from microscopic receptor-ligand interactions at the cell surface, translating external signals into directional movement through a complex network of signaling pathways [49, 122]. Often, models assume an "all-knowing" cell, capable of optimally processing their signaling information for directional migration[2, 13, 108]. Alternative, more realistic models exist with a tight coupling between signaling, cytoskeleton remodeling, and cell-shape dynamics [118, 256], but from which insights are more difficult to obtain.

Here, we model pseudopod splitting through the dynamics of actin polymerization, wherein the competition for a finite resource between extending pseudopods determines the next cell movement direction. By quantitatively describing intracellular interactions within an interpretable model, we gain insights into the fundamental principles governing cellular decision-making and their implications for efficient chemotaxis in complex chemical landscapes. To generalize, we employ state-of-the-art deep reinforcement learning (DRL) [79, 246], allowing us to study how pseudopod suppression enhances the cell's ability to correctly choose the direction of movement faster. By optimizing a self-contained but unconstrained suppression policy at a level beyond what is possible with classical optimization, we achieve chemotactic strategies that cells have evolved over evolutionary time scales. One example is the experimentally observed alternation of cells between pseudopod splitting and elongation [118]. Deciphering the key physical principles of embodied computation in the living world allows us to understand the cell body as an analog machine for both information processing and motility. This approach may inspire entirely new classes of intelligent matter designs.

4.2 Methods

4.2.1 Decision-making model

In the context of cellular decision-making, pseudopods play an important role that extends far beyond mere movement [118] – we consider them fundamentally coupled with the sensing process. Experimental results show pseudopod formation originates mainly from two distinct mechanisms: splitting from existing pseudopods or by *de novo* formation (Fig. 4.1a) [112]. To simplify this complexity, our minimal model assumes a unified cell body and pseudopod, where the previous pseudopod serves as the origin for new pseudopod growths, which permits us to describe both mechanisms under the same dynamical framework. We refer to this as a *splitting event* (Fig. 4.1b). A splitting event is defined as a competitive process between n possible directions for the cell to move (or grow, in our case), where the winning candidate dictates the new cell orientation. We allow 12 directional options, providing a suitable number of choices for the cell to navigate its environment. In order to win, each candidate attempts to polymerize as many actin filaments as possible from a finite reservoir of actin monomers.

Actin polymerization. Previous work by Pais et al. [123] successfully modeled collective decision-making on honeybee swarms using stochastic differential equations for

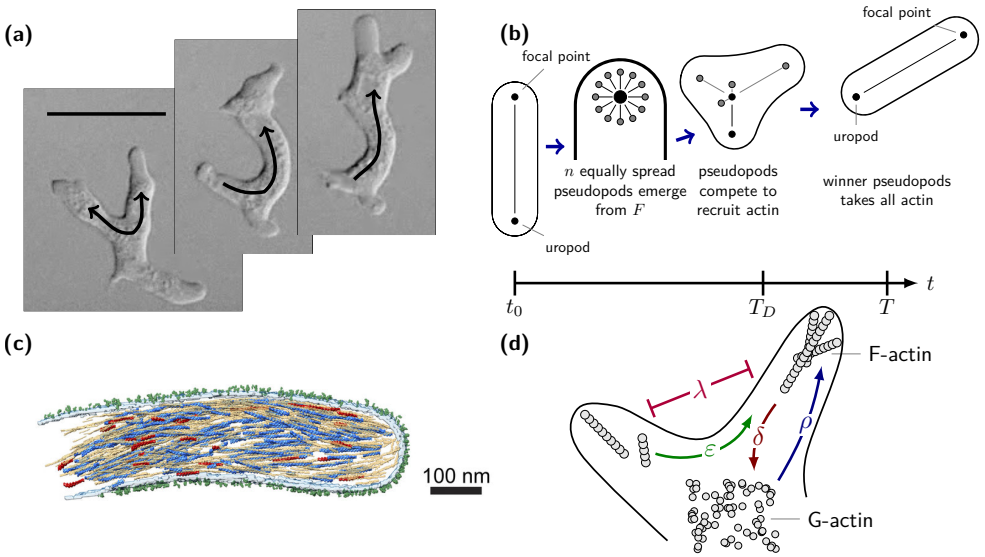


Figure 4.1: **Schematics of the decision-making model and actin dynamics.** **(a)** Pseudopod splitting in a *Dictyostelium* cell performing chemotaxis under agar (provided by Robert Insall) [121]. Arrows indicate likely actin flows, and the scale bar is 20µm. **(b)** Diagram of the cell morphology during a *splitting event*. Pseudopod formation occurs due to competition during actin recruitment. In this instance, only two pseudopods emerge even though $n = 12$ candidates start the competition. Finally, only one remains after decision-making time T_D, altering the cell orientation and advancing its position. The movement step is completed at time T. **(c)** Segmentation of actin filaments inside pseudopodia of small platelets with permission from Sorrentino et al. [116] based on high-resolution structural analysis. Actin filaments are in blue, red, and yellow, while receptors are shown in green. **(d)** Schematics of the simplified polymerization of actin into filaments (F-actin) at the internal membrane surface from actin monomers (G-actin), forming pseudopods. The diagram also shows how pseudopods suppress neighbors by redistributing their actin filaments and mutually inhibiting each other's growth.

valued-based decisions with a finite resource to distribute, e.g., swarm members to 2 or more potential nest sites. Similarly, our model describes actin polymerization on each pseudopod (Fig. 4.1c) with the following overdamped Langevin dynamics

$$\frac{dA_i}{dt} = \rho_i A_u - \delta A_i - \lambda A_i \bar{A}_i + \varepsilon(A_i - \bar{A}_i) + \eta_i(t), \tag{4.1}$$

where A_i are the cumulative proportion of actin monomers that have been polymerized into filaments at pseudopod $\mathcal{P}_i$, and $\bar{A}_i$ are the local sum of actin levels of the other pseudopods $\bar{A}_i = \sum_{j \neq i} A_j$. Assuming actin mass conservation, we set $A_u = 1 - \sum_i A_i$ as the uncommitted actin monomers (G-Actin) proportion inside the cell.

Fluctuations are assumed to arise by the Berg-Purcell noise in chemoattractant measurements [10], which we include by adding noise η with

$$\langle \eta_i(t) \rangle = 0, \qquad \langle \eta_i(t) \cdot \eta_j(t') \rangle \;=\; \sigma \cdot c_i \cdot \delta(t - t') \cdot \delta_{ij}, \tag{4.2}$$

where $\delta(\cdot)$ is Dirac's delta, δ_{ij} is Kronecker's delta, and $c_i \equiv c(x_i)$ is the dimensionless concentration level at the tip of pseudopod $\mathcal{P}_i$ (see 4.2.2 for details on non-dimensionalization). This assumes ligand noise as Poissonian due to the random arrival

of ligand molecules at the cell surface by diffusion, and internal noise due to fluctuations in actin levels is ignored.

The polymerization rate ρ_i of actin filaments inside a given pseudopod $\mathcal{P}_i$ is influenced by the local concentration of chemoattractant, which activates well-known signaling pathways that lead to increased actin polymerization [112]. Thus, we model the amount of G-actin recruited by a pseudopod as being proportional to the signaling activity of the receptors at the pseudopod end. The rate is defined as the average signaling activity of a receptor complex composed of many receptors, which can switch between an on and an off state [257], given by the Boltzmann probability P_{on} with a strength ρ_0

$$\rho_i = \rho_0 \cdot P_{on} = \frac{\rho_0}{1 + e^{\Delta F_i}}, \tag{4.3}$$

where we linearized the change in free energy for small ligand concentration changes, i.e., $\Delta F_i \approx -\kappa_c \, (c_i - c_0)$, with c_i and c_0 being the concentration value at the end of the pseudopod and at the original focal adhesion point, respectively. Note that in an environment without chemoattractant gradient, i.e., constant concentration profile, all pseudopods have the same intrinsic polymerization rate set by $\rho = \rho_0 \,/\, 2$.

F-actin constantly undergoes treadmilling, where individual monomers are removed from one end to be added at the other end of the polymer (Fig. 4.1d). We include this by adding a depolymerization rate (δ). In our case, however, once the monomer has left the filament, we consider it to be returning to the uncommitted actin pool and, thus, potentially being reused by other pseudopods. Furthermore, pseudopods may inhibit each other by sequestering shared resources and signaling crosstalk [258]. Hence, we also include a cross-inhibition (λ) term, where the size of the rival candidates will diminish the overall recruitment speed [123].

Finally, the actin exchange rate (ε) represents the transfer of actin between pseudopods (Fig. 4.1d), as it has been observed that cells can redistribute actin to prioritize certain directions [113, 115]. This term results in a *commitment to the winner* behavior where the cell follows the largest pseudopod the moment it has grown enough to collapse the other candidates back to the focal adhesion point.

Pseudopod growth. To model pseudopod growth, we assume actin to be the sole driver of membrane expansion, thereby linking sensing mechanisms to cellular motility. Hence, by focusing on its intrinsic coupling, we ignore some known effects of membrane mechanics on cellular motility, such as membrane tension, substrate interaction, or surface curvature. Due to the high fluctuations in the actin dynamics at elevated chemoattractant concentrations, we model pseudopod length as a timed-average linear response of F-actin levels, such as

$$\ell_i(t) \;=\; L \int_{t-1}^{t} A_i(t')dt', \tag{4.4}$$

such that the total dimensionless length of the cell is conserved, i.e., $L \;=\; \ell_u + \sum_i^n \ell_i$, with the time for the linear filter used as the characteristic timescale for the dynamics (see 4.2.2 for further details).

Decision time. At the beginning of a *splitting event*, we consider all the motility-associated actin to be unpolymerized $A_i(t_0) = 0$ for all $i \in \{1, \ldots, n\}$, and $A_u(t_0) = 1$. The event ends at $t = T$ when one candidate has gathered most of the actin ($A_i \approx 0.95$). However, the duration of the event contains both the decision-making process and the final growth of the winning pseudopod until it takes all remaining actin. Since we are interested in the decision time T_D, we define it as the time beyond which the length of the winning pseudopod is larger than the summed lengths of all remaining candidate pseudopods, i.e.

$$\ell_i(t) \geq \sum_{j \neq i} \ell_j(t), \quad \forall t \geq T_D. \tag{4.5}$$

Concentration profile. To simulate chemotactic environments, we assume a linear gradient of concentration profile similar to those observed in chemotactic chambers [118], such as

$$c(x) = g_x x + c_n. \tag{4.6}$$

where g_x is the concentration gradient and c_n is the background concentration at the origin. This is a common approximation to the resulting concentration profile based on Flick's second law of diffusion of a constant chemoattractant value that diffuses from the side of the chamber. We treat gradient magnitude and background concentration as freely varying initial conditions to investigate their impact on cell decision-making processes. Notably, the profile is defined with unitless variables as described in the Methods section.

4.2.2 Implementation

Numerical simulations. Actin dynamics are integrated using the Euler-Maruyama integration scheme, converging to the Ito solution. Discrete-time steps are set to $\Delta t = 0.1\ s$ during the simulations unless explicitly stated. Thousands of realizations are carried out for each numerical result by massively parallelizing the simulations using GPUs (see Code availability).

Non-dimensionalization of the dynamics. To simplify the equations, we non-dimensionalize the system by scaling all lengths relative to cell size a. Similarly, time is defined relative to τ_m, a characteristic timescale for the linear filter in Eq. (4.4), which relates to the mechanical properties of the membrane. Accordingly, the gradient and concentration levels are rescaled by length scale a, resulting in unitless concentration profile Eq. (4.6). Hence, length scale a changes the concentration profile while maintaining the same cell dimensions. The cell dynamics are described in the cell's frame of reference.

Suppression policy architecture. We model the suppression policy ρ_θ as a relatively small artificial neural network whose input is the logarithm in base 10 of the SNR and whose output is a vector of $(n, 2)$ values between 0 and 1 as the logits for the probability

of activating or suppressing each pseudopod $\boldsymbol{\rho_0}$. The final state is sampled from a categorical distribution. The network is a multi-layer perceptor (MLP) of 4 layers of 128 neurons each, with tanh activation functions between them. Since the policy also predicts the expected value V, we use an MLP with 4 layers and 128 neurons.

Optimizing the suppression policy. The algorithm used here is a modified version of the proximal policy optimization (PPO) algorithm [246] implemented in Alonso and Kirkegaard [2]. PPO is an on-policy optimization technique that iteratively improves its policy p_ω by collecting information from simulations between optimization steps. The results of these simulations are then used to perform stochastic gradient descent on the policy parameters ω. Simply, the algorithm maximizes the following clipped surrogate loss defined as:

$$L^{CLIP}(\omega) = \mathbb{E}\left[\min\left(\frac{p_{\omega_n}(\boldsymbol{\rho_0}|z)}{p_{\omega_p}(\boldsymbol{\rho_0}|z)} \cdot A_t, \right.\right. \\ \left.\left. \text{clip}\left(\frac{p_{\omega_n}(\boldsymbol{\rho_0}|z)}{p_{\omega_p}(\boldsymbol{\rho_0}|z)}, 1-\epsilon, 1+\epsilon\right) \cdot A_t\right)\right],$$

where ω_n represents the updated parameters of the policy, while ω_p indicates the previous policy parameters. The term $p_\omega(\boldsymbol{\rho_0}|z)$ is the probability of output $\boldsymbol{\rho_0}$ given the SNR, here indicated by z, under the new policy. The advantage function A_t quantifies the relative benefit of taking a particular action at a given state compared to the average action value. Notably, the clipping parameters ϵ control the size of the trust region, ensuring that new updates do not deviate significantly from the previous policy, which leads to more stable optimizations. In practice, the clipping is performed by defining upper and lower bounds on the allowed change in ratio between consecutive policies that contribute to the loss.

To promote exploration, especially given the discrete nature of our actions (active or suppressed candidate), we include an entropy term on the loss function as a regularization term set by

$$H = -\frac{1}{n}\sum_{i}^{n} p_\omega^{(i)}(z)\log(p_\omega^{(i)}(z))$$

which encourages the policy to maintain a certain degree of stochasticity, preventing collapses and premature convergence. Here, n is the number of actions, which is set as the number of candidate pseudopods.

4.3 Results

Effect of chemoattractant on actin dynamics. We evaluate the system at different concentration profiles and observe the difference in the actin dynamics during the

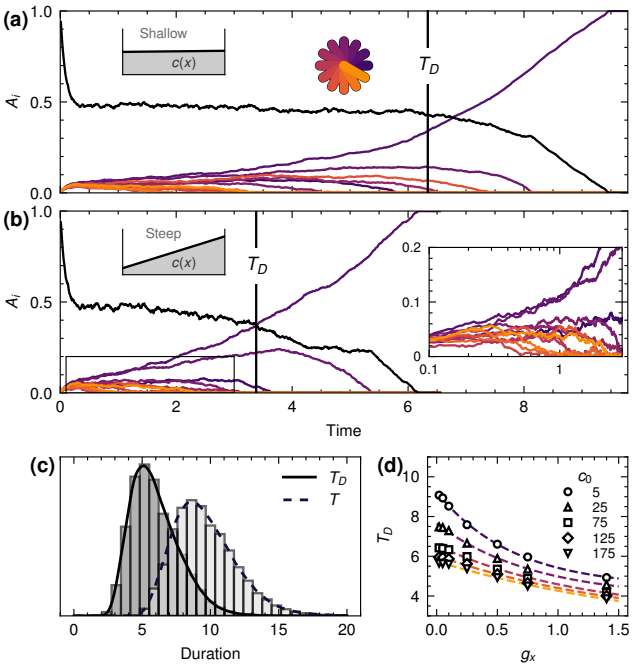


Figure 4.2: **Actin and pseudopods dynamics during decision-making. (a-b)** Sample trajectories of F-actin levels of each of the $n = 12$ candidate directions of the cell, shown in the circular diagram in (a), in a linear concentration profile Eq. (4.6) in a shallow (a: g_x=0.01, c_0=50) and a steeper (b: g_x=1, c_0=150) gradient environment, respectively. The inset in (b) shows a closeup of the initial dynamics of the competing pseudopods on log time. The proportion of G-actin is shown as a black line. The decision time T_D is also marked with a vertical line set by Eq. (4.5). **(c)** Duration distribution of the events T and the decision times T_D. **(d)** Decision time as a function of the chemoattractant gradient for different noise levels set by the concentration value.

competition event (see Fig. 4.2). The trajectories quickly start by equally recruiting monomers regardless of direction, which halves the pool of G-actin A_u. As shown in Fig. 4.2b, the F-actin levels fluctuate until some candidates are suppressed while others remain, forming pseudopods. This process is quicker when the gradient is stronger. The resulting distributions of decision time and total duration for many combinations of gradients and concentration values (Fig. 4.2c) indicate that decision time only accounts for half the duration of the event. This demonstrates that, despite cell movement and decision-making being coupled, the decision occurs before the final pseudopod grows and collapses all other candidates, directing the entire cell body toward the chosen direction. Interestingly, when studying the effect of the environment in the decision time of the cell, we observe an exponential decaying dependency with the gradient strength (Fig. 4.2d). Hence, the cell reacts faster in environments where the gradient (signal) information is stronger. Similarly, a larger concentration background level c_0 (noise) also causes the cell to decrease its decision time.

Emergence of Weber-like law. When examining cell decision success rate in response to chemoattractant gradients, we find the minimal signal strength required for a consistent movement up the gradient scales with noise strength due to measurement fluctuations (Fig. 4.3a). This phenomenon echoes Weber's law of just noticeable differences, which describes a linear relationship between signal strength and mean value. Notably, here, this property emerges without relying on logarithmic transformations of the signal, commonly used in chemotaxis studies [2, 257]. Instead, the difference between concentrations, e.g., at the pseudopod tip and at its original position at the focal adhesion point, suffices to produce Weber law-like behavior, implying that cells may use a simple, local comparison mechanism to make directional decisions. However, our model results in a square-root dependency on concentration, approximating linearity when noise contributions become significant (see Fig. 4.3a). The square-root dependency indicates that the cell success rate scales with the signal-to-noise ratio (SNR) instead, which is defined as

$$\mathrm{SNR} = \frac{g_x^2}{c(x_0)}. \tag{4.7}$$

This can be easily understood in our model, Eq.(4.1), as our cells make finite-difference estimates of concentrations across pseudopods, effectively measuring gradients, while the noise Eq.(4.2) scales as the square root of the chemoattractant concentration. This result also matches previous experimental observations [118] and is consistent with Weber's law observed in the chemotaxis of some cell types [259, 260].

Interestingly, adjusting ε, which determines how fast the cell commits to the winner pseudopod, modifies the slope of the dependency while preserving its linearity (Fig. 4.3b). This showcases robustness in the behavior of the cell, which, despite changing the decision accuracy and speed, conserves the overall dependencies on the environment information.

Pseudopod competition leads to indirect gradient sensing. A commonly used measurable observable for chemotactic performance is the *chemotactic index*, which was previously calculated using the physical limits of sensing by Endres and Wingreen [13]. The probability of estimating the gradient of a concentration by a perfectly ligand-absorbing cell (to avoid the noise from rebinding) is

$$P(\hat{g}_x, \hat{g}_y) = \frac{1}{2\pi\sigma_g^2} \exp\left[\frac{-(\hat{g}_x - g_x)^2 - (\hat{g}_y - g_y)^2}{2\sigma_g^2}\right], \tag{4.8}$$

where $(\hat{g}_x, \hat{g}_y)$ is the estimated gradient, and the real one is given by $\nabla c = (g_x, g_y)$, with the uncertainty of the measurement being

$$\sigma_g^2 = \frac{c_0}{12\pi DT}.$$

where T and D are the relative time the cell takes to measure the gradient and the unitless diffusion constant of the chemoattractant, respectively. From Eq. (4.8) and assuming a linear gradient on x such that $\nabla c = (g_x, 0)$, we obtain that the expected

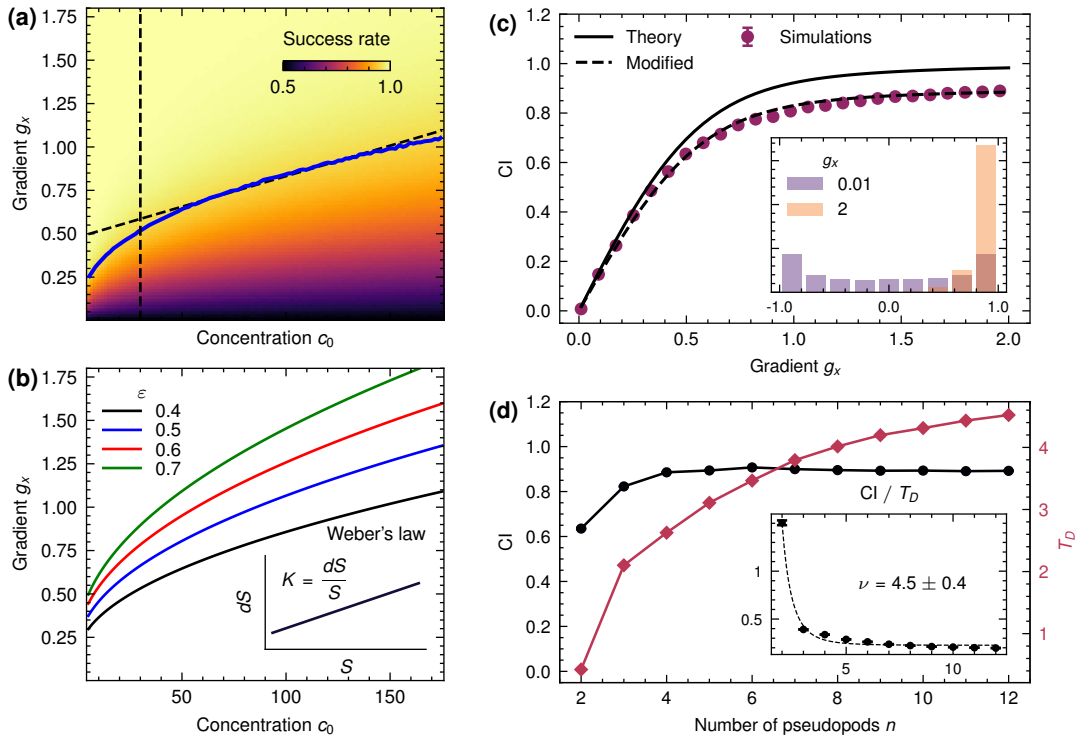


Figure 4.3: **Weber-like law and speed-accuracy tradeoff. (a)** A heatmap of the success rate at different values of gradient and background concentration. Each rate value (square) is calculated by averaging 10^5 independent splitting events with randomly oriented cells and considering it a success if the final movement of the cell has led it to a higher concentration, i.e., $c(x_T) > c(x_0)$. The blue line indicates the minimum gradient at which the accuracy of choosing the direction up the gradient reaches 0.95, the threshold rate at which we consider the cell to be making the correct decision unambiguously, as the perfect rate is subject to numerical fluctuations. The vertical dashed line indicates the threshold at which we consider the linear regime to begin ($c(x) > 30$). A linear fit is shown in a dashed line on top of the minimum gradient for the linear region. **(b)** The minimum gradient line changes for different actin exchange parameters ε. The inset showcases what is commonly understood as Weber's law, a scalar ratio between the perceived change in stimulus (dS) and stimulus value S. **(c)** Accuracy of aligning the cell body with the gradient, given by the chemotactic index (CI). The solid line showcases the optimal CI of a perfect absorbing cell set by Eq. (4.9), while the dashed line adjusts it by a factor 0.9. The inset exemplifies the resulting distributions of the alignment of the cell at low and high gradients, respectively. **(d)** CI (black) and mean decision time T_D (red) as a function of the number of candidate pseudopods at the highest gradient ($g_x = 2$). The inset shows the rate of alignment, fitted to the power law $n^{-\nu}$.

cosine similarity of the gradient and direction of movement. This is the chemotactic index given by

$$\mathrm{CI} = \langle \cos(\theta) \rangle = \sqrt{\frac{\pi z}{2}} e^{-z} \left[I_0(z) + I_1(z) \right], \tag{4.9}$$

where $z = 3\pi k$ SNR, θ is the orientation of the cell, and $I_{0(1)}$ are first (second)-order modified Bessel functions. The combination $k = DT$ can be thought of as a single fitting parameter. During the measuring time, it is assumed that the cell processes

the measurements by averaging their positional information before making a decision. Instead, Fig. 4.3c shows that this processing emerges from the competition between pseudopods until one candidate is chosen and the cell aligns itself with the estimated gradient. However, the resulting CI from numerical simulations saturates at a lower value due to the similarity in signal between neighboring candidates and the saturation of the receptor's signaling, set by Eq.(4.3). Agreement with the theory is only obtained when multiplying Eq.(4.9) by a 0.9 factor. Interestingly, this was also done in Endres and Wingreen [13] when comparing to data.

Pseudopods competition reveals speed-accuracy trade-off. Due to the pseudopods' fixed orientation, the larger the number of candidates, the more likely a pseudopod will align perfectly with the gradient direction. As shown in Fig. 4.3c, the closeness of the candidates, together with the low signal or strong fluctuations of the measurements, saturate the decision. Thus, we compare the resulting alignment at the highest signaling strength for an increasing amount of candidates (Fig. 4.3d) and notice that the alignment increases with n up to a saturation point at $n = 6$. Curiously, when estimating the average decision time $\langle T_D \rangle$, we observe a monotonic increase as well, without reaching a saturation state, pointing towards a diminishing return in the number of candidates in terms of efficiency. Furthermore, by evaluating the rate of alignment (Fig. 4.3d), we infer a power-law-like behavior $\mathrm{CI}\,/\,T_D \approx n^{-\nu}$, with $\nu = 4.5$, which clearly indicates that the lower the number of candidates, the lower the decision time, and the more efficient the decision making (despite a decrease in absolute accuracy).

Chemotaxis trajectories depend on SNR. The dynamics of the *splitting event* move the cell body towards its chosen direction. When completed, the winning pseudopod becomes the cell body in a new location with a new orientation, ready to start another event. Consequently, in our model, chemotaxis emerges as a sequence of consecutive pseudopod-splitting events. Similar to the decision success rate in Fig. 4.3a, by extending pseudopod splitting to many events, the resulting ensemble displacement is strongly affected by the SNR (Fig. 4.4a). The spread of the cell during the trajectory is highly correlated with the accuracy of each individual decision. At low SNR, the resulting distribution is almost that of a random walk, whereas, at high values, the cells show strong persistence in moving up the gradient.

In Fig. 4.4c, we compare the chemotactic index of our unsuppressed cell, i.e., with $n = 12$ possible directions of movement (yellow line), with those from experiments (symbols). To quantify the alignment of trajectories, we calculate the chemotactic index as the average of the cosine similarity S_C for binned values of SNR. This can be expressed as $CI = \langle S_C(t) \rangle_{\mathrm{SNR}}$. Due to the shape of our cell, S_C is defined as the weighted average of the orientation of each pseudopod (with respect to the uropod), proportional to their length. Thus, at each timestep, we have

$$S_C(t) = \ell_u \ \cos(\theta) + \sum_i^n \ell_i \ \cos(\theta + \phi_i), \tag{4.10}$$

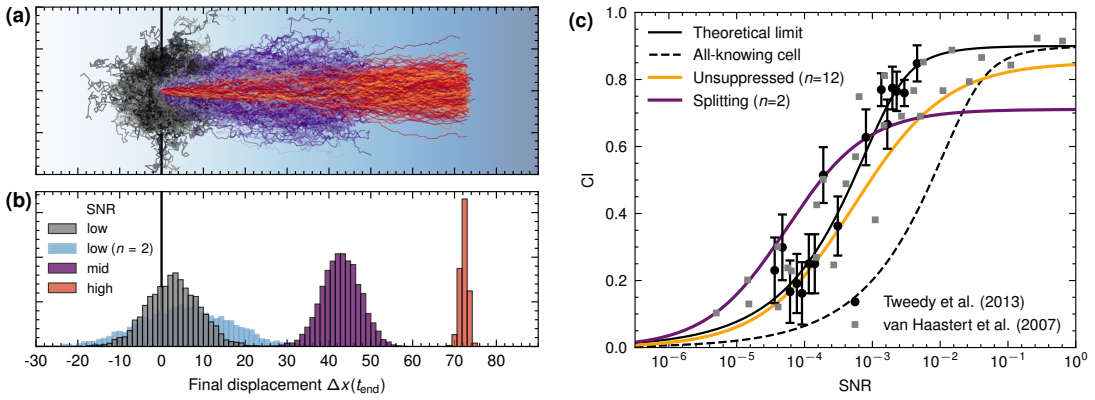


Figure 4.4: **Chemotaxis performance depends on SNR. (a)** Sample of $3\,10^2$ chemotaxis trajectories of our model, composed of 30 *splitting events*, at different levels of SNRs. The vertical black line indicates the initial positions of the cell. Since SNR changes along a trajectory, low here contains SNR $\in [10^{-5.874}, 10^{-5.875}]$, mid contains $[10^{-3.26}, 10^{-3.29}]$ and high $[10^{-1.87}, 10^{-2.06}]$. **(b)** Final displacement distribution ($N = 7\,10^3$) of the trajectories in (a). A distribution for the splitting configuration with $n = 2$ at low SNR is also included in blue. **(c)** Chemotactic index (CI) as a function of SNR, plotted on a $\log_{10}$ scale. The fundamental physical limit for a static spherical ligand absorber is given by Eq. (4.9) [13], fitted to the experimental data from [105, 118]. Additionally, the performance of an *all-knowing* cell that estimates the gradient based on ($n = 12$) sensors uniformly distributed across its body is shown in a dashed line (see Appendix A1 for details). In yellow is the resulting CI of 10^4 independent trajectories, from which at each timestep the CI is calculated using Eq.(4.10) and binned according to the SNR at the start of the event. Similarly, the purple line shows the results for a cell with only activated $\mathcal{P}_2$ and $\mathcal{P}_{10}$. For clarity, the lines are fitted to a logistic function using the numerical simulations binned by SNR.

where θ indicates the cell orientation and ϕ_i the relative orientation of the candidate $\mathcal{P}_i$ w.r.t. the cell. Interestingly, at lower SNR, when the gradient is shallow and difficult to infer, the average performance resembles the expected one of a perfect absorbing cell at the fundamental limit of sensing (black solid line). Note that for wide-ranging attractant concentrations, our cells with only two pseudopods perform significantly better than all-knowing cells, implemented by 12 sensors, sophisticated inference by least squares fitting, and uncertainty based on the Cramér-Rao bound (see Appendix A1 for details). Furthermore, as the signal increases, the model approaches the lower bound of the experimental data, ultimately saturating at ~ 0.9 chemotactic index. There is notably excellent agreement between our model and the data without any explicit fitting.

Pseudopod suppression to enhance chemotactic efficiency. Until now, unless explicitly specified, we assumed that the cell has n=12 evenly spaced distributed pseudopod candidates, sufficient for the cell to accurately choose the right moving direction. However, we observed that fewer candidates can drastically improve efficiency during decision-making (Fig. 4.3d). Experimental observations showed that the angle between pseudopods is affected by the chemoattractant gradient shallowness [112, 122]. Based on these observations, we suppress all pseudopods candidates except $\mathcal{P}_2$ and $\mathcal{P}_{10}$, which grow at a $\varphi = \pm\, 60°$ of the cell movement orientation, and observe their CI as the SNR

increases (Fig. 4.4c). The resulting chemotactic index shows that the cell is able to move up the gradient more robustly at small SNRs than when allowed to change directions freely ($n = 12$). This is also seen on the final displacement at Fig. 4.4b at very low SNR, where this configuration manages to reach further up the gradient. Nevertheless, as the quality of the signal increases, the cell with the wider range of options performs better, as the two pseudopods likely do not align with the actual gradient. This cross-over in the chemotactic index can be seen in the data of cell morphology, where amoeba use pseudopod splitting at small SNR and a broad-front polarization at high SNR [118].

In previous work [2], we observed the advantage of gradual orientation updates instead of full turns when optimizing a fully unconstrained spatial policy using deep reinforcement learning (DRL) at the fundamental limits of ligand sensing, confirming that relying on persistence is an effective strategy when navigating shallow gradients. What can DRL tell us about the ideal number of pseudopods and their orientation?

Optimal pseudopod suppression policy. Having observed a clear advantage in suppressing possible directions, we explore the possibility of an optimal configuration in which the cell may have learned to exploit suppressing certain directions throughout the course of evolution to enhance its chemotactic performance. Here, we optimize a mapping function $p_\theta : \mathbb{R} \to [0,1]^n$ that indicates the probability of not suppressing candidate i by setting the signal strength to either suppressed ($\rho_0 = 0$) or active ($\rho_0 = 1$). Due to the number of possible combinations of pseudopod states, classical optimization techniques are computationally unfeasible. Therefore, we rely on modern reinforcement learning approaches, specifically, proximal policy optimization (PPO) [246], to locate the optimal suppression policy, here constructed as an artificial feed-forward neural network, given the environment's state, set by the SNR at the cell location at the start of the event (see Fig. 4.5a and Methods for further details).

We define the reward function after each decision step during the chemotactic trajectory of the cell (i) as

$$R^{(i)}(x_T, T) = \cos(\theta_T) + \gamma \left(\frac{t_{max} - T}{t_{max}} \right), \tag{4.11}$$

where t_{max} is the maximum possible time for the cell to make a decision before a random one is selected, and γ is a time penalty to favor the configurations that lead to faster events. Subsequently, we optimize the weights on the network by maximizing the cumulative reward $\hat{R}^{(i)}$ during a trajectory, which we specify as the concatenation of 30 splitting events.

The resulting configurations further confirm the important role of persistence in accurate chemotaxis, as it enables the cell to place its receptors at the leading edge, and by relying on consecutive turns, the cell further aligns itself with the gradient. Despite exploring a wide range of SNR, only two final configurations emerge that surpass both of our previous proposed ones (Fig. 4.5b). At low SNR, two candidate pseudopods, closer than those we constructed from experimental observations, prove optimal, further demonstrating that relying on small changes reduces the likelihood of sudden errors.

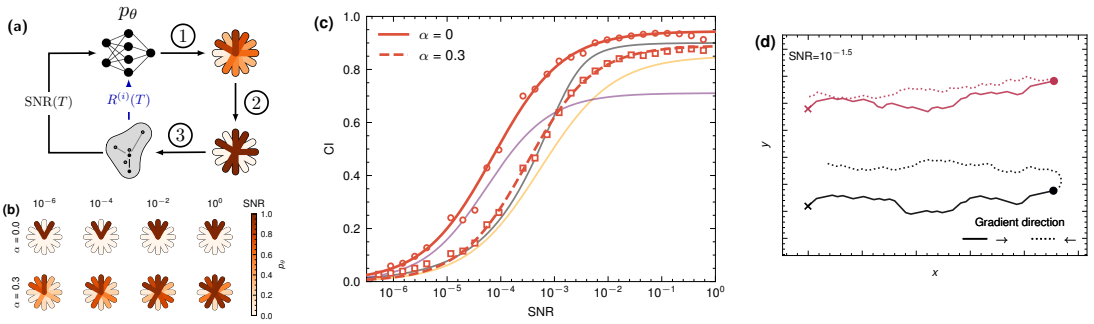


Figure 4.5: **Optimal pseudopod suppression strategy. (a)** Schematics of the DRL training process, where (1) the policy outputs activation probabilities of each candidate, (2) a sampling occurs to suppress certain directions, and (3) a splitting event is simulated, resulting on a new outcome which we use to optimize the weights of the policy, and a new SNR that will be used as input for the next step. **(b)** The cell diagrams show the resulting activation probability p_θ of the 12 candidates at different SNR for both $\alpha = 0$ and $\alpha = 0.3$. On the schematics, the cell direction movement is up. **(c)** Evaluation of the optimal policy p_θ using PPO with $\gamma = 0.2$, based on how well pseudopods align with the gradient compared to the results from Fig. 4.4c. The resulting points are fitted with a logistic function and are the result of 10^4 independent simulated trajectories, from which, at each time step, the CI is calculated and binned by SNR. Both the policy strained on static profile concentrations ($\alpha = 0$), and the one trained with a high rate of changing gradient direction ($\alpha = 0.3$) are shown despite being evaluated on a static gradient. **(d)** Trajectories of the suppression policy trained in a static gradient (black) and in a dynamic gradient with a high rate of switching (red). The initial position of the cell is marked with a cross, while the point of the trajectory at which the gradient sign is changed is indicated by a dot.

Given that the limited information the cell gains from the environment is not enough to distinguish forward and side directions, the found policy optimizes for time, which results in a configuration with a minimal number of candidates ($n = 2$), echoing our results on alignment efficiency from Fig. 4.3d. However, as the information from the environment increases, the subtle difference between neighboring candidates becomes apparent. The resulting policy at high SNR activates the forward pseudopod candidate while still minimizing the number of candidates to yield better alignment during long trajectories, where the concentration profile remains static. When comparing the CI of the trajectories as a function of the SNR, we obtain an envelope curve, resulting in an upper limit to previous performances (Fig. 4.5c).

The ability of a cell to react and adapt to sudden changes in the environment is another criterion of successful chemotaxis. Specifically, Aquino et al. [261] investigated how microorganisms respond to abrupt changes in the chemoattractant direction, demonstrating that cells can adapt when the direction is suddenly reversed, at least in steep gradients. We introduce the possibility, set by α, for the gradient direction to randomly change during a chemotaxis trajectory, which we include while training the policy. Given that our previous policy exploited the persistence of movement in a static profile, we examine the performance of both the old policy ($\alpha = 0$) and the new one ($\alpha = 0.3$) in enabling the cell to adjust its trajectory in response to a sudden change in the gradient sign. Interestingly, when $\alpha = 0.3$, the resulting pseudopod configurations

are more spread, showing the possibility of activating the rear pseudopod candidates at low SNR and consistently activating them at high SNR, where the frontal candidates are likely to win when the difference in signal is notable (Fig. 4.5b). Observing example trajectories, a forward-facing configuration reacts slower to changes in the gradient, performing U-turns to realign with the source direction on both low and high SNR scenarios (Fig. 4.5d). This contrasts with the adaptive policy, which can immediately reverse direction when the change occurs at high SNR, reproducing previously observed behaviors [261] and the cells' ability to form *de novo* pseudopods [112]. Despite the benefit of a better reaction time to sudden changes in the environment information, when evaluated on a static concentration profile as those more common on cellular chemotaxis, we recover a similar performance as the one set by the limits of sensing (Fig. 4.5c). Thus, cells lose the advantage persistence yields for overcoming those limits. Combining DRL, an effective optimization technique for non-trivial problems, with the interpretability of classical systems biology, we obtain insights into how different evolutionary traits may have altered the navigation strategies of motile, chemotactic cells.

4.4 Discussion

We introduced a minimal model for cellular decision-making based on the competition between pseudopods with stimulus-dependent growth. Instead of relying on an all-knowing cell that evaluates the measurements of the chemical gradient to decide its orientation of motion, we propose that pseudopod dynamics, driven by the stochastic processes of actin polymerization, simple G-actin conservation, and mutual inhibition, are responsible for the cell's emergent decision. Hence, no direct spatial gradient sensing is required. Despite its simplicity, we have shown that our model agrees with the theoretical limit while providing insights into the time costs of moving up the chemoattractant gradient. Furthermore, our model captures the key features of pseudopod dynamics as observed in experiments, such as the emergence of multiple pseudopods on shallow low-SNR gradients and the scaling of the signal with noise to move up the gradient unequivocally, reminiscent of Weber's law.

When concatenating consecutive decision steps, i.e., splitting events, we modeled chemotaxis trajectories capable of reproducing experimental data, where our results showed a characteristic dependence of the chemotactic index on the SNR, previously postulated in Endres and Wingreen [13]. We then extended the model by incorporating a learnable suppression mechanism, allowing us to explore how cells might have optimized their polarization for efficient chemotaxis, particularly at low SNR, where gradient information is especially limited. By employing deep reinforcement learning, we found a mapping between the SNR and pseudopod suppression. Interestingly, the learned suppression policy converges towards a behavior where the cell preferentially suppresses pseudopods at small angles, leading to a more focused forward distribution of directions of motion. These findings align with experimental observation in shallow gradients, whereby cells exhibit a fixed angular spread of forward-facing pseudopods [112].

Spread-out cell protrusions are optimal for instantaneous sensing, as they increase

the spatial information of the environment around the cell [253, 262]. Our results suggest that cell polarization promotes forward-facing pseudopods, which, while clearly suboptimal for an instant decision, prove advantageous during chemotaxis. By leveraging persistence, which bypasses the need for a memory of prior information, the cell is capable of maintaining a consistent direction of movement, thereby improving its ability to navigate gradients over longer trajectories. Thus, our findings highlight the tradeoff between instantaneous, accurate sensing and overall chemotactic performance.

The model proposed here does present certain limitations, as assumptions were made to facilitate interpretability and simplification. Thus, future research may want to expand on this work by increasing its realism and complexity. Notably, we have ignored the physical effects of the moving cell, e.g., the "windshield effect" of a ligand-absorbing cell [13, 118] and the mechanical properties of the cell membrane as the cell crawls on the substrate, as well as molecular details of the spatial positioning of neighboring pseudopods with mutual inhibition. Such positioning and regulation could be achieved by simple reaction-diffusion mechanisms of activators and inhibitors [118]. Furthermore, our simplification of concentration sensing in terms of stimulus-dependent growth of pseudopods could be investigated in more detail to capture the effects of ligand binding and unbinding times or ligand-induced receptor internalization. Further experimental validation of our model may include tracking the dynamics of pseudopods in testable decision-making scenarios according to the SNR. We hope further research into the mechanistic foundation of decision-making may yield novel insight into cellular behavior.

Amoeboid cell migration based on pseudopods and other cell protrusions is not unique to *D. discoideum* but also occurs in neutrophils [121] and even in the spermatozoa of *C. elegans* [263]. Notably, the latter achieves motility without actin, emphasizing shape and behavior as fundamentally important and that the biochemical details might be secondary [50]. Instead, we suggest that amoeboid shape and behavior are evolutionarily conserved traits, providing advantages in fast and accurate chemotaxis. Even the syncytial plasmodia of the slime mold *Physarum polycephalum* forages and grows as a macroscopic network [264]. Not unlike our proposed stimulus-driven pseudopod extension in Dicty, nutrient uptake on one end of the network drives extensions in this favorable direction, leading to retraction at the rear. Whether our navigation strategies are also relevant to ciliated micro-organisms [7] or group chemotaxis [265] are fascinating open questions. However, our results clearly go beyond chemotaxis in cells. With an increasing interest in developing microscopic artificial agents, a theoretical framework for decision-making in difficult-to-navigate environments is crucial. Amoeba-inspired limbless robotics may benefit from our robust strategy designs without requiring extensive hard-wired sensor-driven feedback mechanisms [108, 266, 267]. We showed that by suppressing the strength of candidate directions, the cell can use persistence to effectively navigate up a gradient by focusing on creating fewer forward-facing protrusions.

In conclusion, our work proposes a new understanding of the fundamental principles governing cell decision-making and their implications for chemotaxis in complex environments. We showed that pseudopod splitting leads to highly effective chemotaxis without a need for direct spatial sensing and memory, demonstrating aspects of mechanical intel-

ligence. The paper also highlights the potential of reinforcement learning as a powerful tool for studying and understanding the intricate interplay between cellular mechanics, sensing, and behavior without relying on black-box decision policies that obscure the internal cellular mechanism. Applications in robotics are apparent.

Appendix

A1 Gradient Sensing by a Cell with N Sensors

Here, we derive the chemotactic performance of a cell that has surface sensors and can accurately process the information these send to estimate the gradient, also referred to in the main text as an *all-knowing* cell. We assume the cell has N sensors in a plane with coordinates (x_i, y_i) for each sensor, with the cell center of mass at the origin. Each sensor measures the ligand concentration c_i with some uncertainty δc_i.

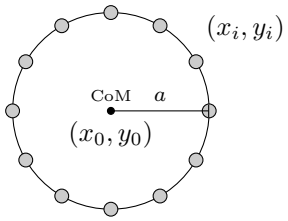


Figure 4.6: Diagram of a cell with N sensors spread on its surface.

How would the cell infer the direction of the chemical gradient in this plane using least squares fitting?

Gradient Estimation

Assume the ligand concentration c at any point (x, y) in the plane is given by:

$$c(x, y) = c_0 + g_x x + g_y y \tag{4.12}$$

where c_0 is the concentration at the origin, and g_x and g_y are the components of the gradient vector in the x and y directions, respectively.

Each sensor i located at (x_i, y_i) measures the concentration c_i with uncertainty δc_i, and the objective is to minimize the sum of squared differences between the measured concentrations and the model concentrations, defined as

$$S = \sum_{i=1}^{N} \left(c_i - (c_0 + g_x x_i + g_y y_i)\right)^2 \tag{4.13}$$

To minimize it, we take the partial derivatives of S with respect to c_0, g_x, and g_y

and set them to zero:

$$\frac{\partial S}{\partial c_0} = -2\sum_{i=1}^{N}\left(c_i - (c_0 + g_x x_i + g_y y_i)\right) = 0 \tag{4.14}$$

$$\frac{\partial S}{\partial g_x} = -2\sum_{i=1}^{N} x_i\left(c_i - (c_0 + g_x x_i + g_y y_i)\right) = 0 \tag{4.15}$$

$$\frac{\partial S}{\partial g_y} = -2\sum_{i=1}^{N} y_i\left(c_i - (c_0 + g_x x_i + g_y y_i)\right) = 0 \tag{4.16}$$

which can then be rewritten as

$$\sum_{i=1}^{N} c_i = Nc_0 + g_x\sum_{i=1}^{N} x_i + g_y\sum_{i=1}^{N} y_i \tag{4.17}$$

$$\sum_{i=1}^{N} x_i c_i = c_0\sum_{i=1}^{N} x_i + g_x\sum_{i=1}^{N} x_i^2 + g_y\sum_{i=1}^{N} x_i y_i \tag{4.18}$$

$$\sum_{i=1}^{N} y_i c_i = c_0\sum_{i=1}^{N} y_i + g_x\sum_{i=1}^{N} x_i y_i + g_y\sum_{i=1}^{N} y_i^2 \tag{4.19}$$

and since these are linear equations, we can express them in matrix form as

$$\begin{pmatrix} N & \sum x_i & \sum y_i \\ \sum x_i & \sum x_i^2 & \sum x_i y_i \\ \sum y_i & \sum x_i y_i & \sum y_i^2 \end{pmatrix}\begin{pmatrix} c_0 \\ g_x \\ g_y \end{pmatrix} = \begin{pmatrix} \sum c_i \\ \sum x_i c_i \\ \sum y_i c_i \end{pmatrix} \tag{4.20}$$

with the gradient estimated direction being $\mathbf{g} = (g_x, g_y)$.

Simplification for $c_0 = 0$

If the ligand concentration is measured relative to the center of mass's ligand concentration, effectively making $c_0 = 0$, the problem simplifies as we only need to determine the gradient components g_x and g_y. Thus, the model for the concentration at any point (x_i, y_i) changes from (4.12) to

$$c_i = g_x x_i + g_y y_i \tag{4.21}$$

with its corresponding objective function

$$S = \sum_{i=1}^{N}\left(c_i - (g_x x_i + g_y y_i)\right)^2 \tag{4.22}$$

Similarly to before, we take the partial derivates of S with respect to g_x and g_y and

set them to zero:

$$\frac{\partial S}{\partial g_x} = -2\sum_{i=1}^{N} x_i \left(c_i - (g_x x_i + g_y y_i)\right) = 0 \tag{4.23}$$

$$\frac{\partial S}{\partial g_y} = -2\sum_{i=1}^{N} y_i \left(c_i - (g_x x_i + g_y y_i)\right) = 0 \tag{4.24}$$

which we rewrite and express them in matrix form leading to

$$\begin{pmatrix} \sum_{i=1}^{N} x_i^2 & \sum_{i=1}^{N} x_i y_i \\ \sum_{i=1}^{N} x_i y_i & \sum_{i=1}^{N} y_i^2 \end{pmatrix} \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{N} x_i c_i \\ \sum_{i=1}^{N} y_i c_i \end{pmatrix} \tag{4.25}$$

Including Measurement Uncertainty

So far, we have assumed that the cell can accurately measure the concentration values on its sensors. For a more realistic derivation, we include the measurement errors δc_i in the ligand concentrations, which will modify the least squares fitting process to account for the uncertainties in the data. This involves using weighted least squares, where each measurement c_i is weighted based on its uncertainty δc_i.

In weighted least squares, each residual is weighted by the inverse of the variance of the corresponding measurement. Therefore, the weight for each measurement is $w_i = \frac{1}{\delta c_i^2}$, where c_i is the measured concentration, leading to the objective function:

$$S = \sum_{i=1}^{N} w_i \left(c_i - (g_x x_i + g_y y_i)\right)^2 . \tag{4.26}$$

Analogously to the previous derivations, we express the linear system in matrix form as

$$\begin{pmatrix} \sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2} & \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} \\ \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} & \sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} \end{pmatrix} \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{N} \frac{x_i c_i}{\delta c_i^2} \\ \sum_{i=1}^{N} \frac{y_i c_i}{\delta c_i^2} \end{pmatrix} \tag{4.27}$$

where now the components of the matrices are similar to (4.25) divided by their uncertainties.

Solving the sytem Let the previous linear system (4.27) be

$$A_w \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \mathbf{b}_w \tag{4.28}$$

where

$$A_w = \begin{pmatrix} \sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2} & \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} \\ \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} & \sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} \end{pmatrix}; \quad \mathbf{b}_w = \begin{pmatrix} \sum_{i=1}^{N} \frac{x_i c_i}{\delta c_i^2} \\ \sum_{i=1}^{N} \frac{y_i c_i}{\delta c_i^2} \end{pmatrix} \tag{4.29}$$

the inverse of A_w is given by

$$A_w^{-1} = \frac{1}{\det(A_w)} \begin{pmatrix} A_{22} & -A_{12} \\ -A_{21} & A_{11} \end{pmatrix} \tag{4.30}$$

where $\det(A_w)$ is the determinant of A_w:

$$\det(A_w) = A_{11}A_{22} - A_{12}A_{21} \tag{4.31}$$

Using the elements of A_w from (4.28):

$$A_{11} = \sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}, \quad A_{12} = A_{21} = \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}, \quad A_{22} = \sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} \tag{4.32}$$

we can express (4.30) as

$$A_w^{-1} = \frac{1}{\left(\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}\right)\left(\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}\right) - \left(\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}\right)^2} \begin{pmatrix} \sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} & -\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} \\ -\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} & \sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2} \end{pmatrix}, \tag{4.33}$$

The gradient components g_x and g_y are obtained by multiplying A_w^{-1} with $\mathbf{b}_w$:

$$\begin{pmatrix} g_x \\ g_y \end{pmatrix} = A_w^{-1}\mathbf{b}_w \tag{4.34}$$

which we use (4.33) and (4.28) to express as

$$\begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{\det(A_w)} \begin{pmatrix} \sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} & -\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} \\ -\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} & \sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2} \end{pmatrix} \begin{pmatrix} \sum_{i=1}^{N} \frac{x_i c_i}{\delta c_i^2} \\ \sum_{i=1}^{N} \frac{y_i c_i}{\delta c_i^2} \end{pmatrix} \tag{4.35}$$

with

$$\det(A_w) = \left(\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}\right)\left(\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}\right) - \left(\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}\right)^2 \tag{4.36}$$

Performing the matrix multiplication and simplifying the results leads to

$$g_x = \frac{\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} \sum_{i=1}^{N} \frac{x_i c_i}{\delta c_i^2} - \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} \sum_{i=1}^{N} \frac{y_i c_i}{\delta c_i^2}}{\left(\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}\right)\left(\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}\right) - \left(\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}\right)^2} \tag{4.37}$$

$$g_y = \frac{\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2} \sum_{i=1}^{N} \frac{y_i c_i}{\delta c_i^2} - \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2} \sum_{i=1}^{N} \frac{x_i c_i}{\delta c_i^2}}{\left(\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}\right)\left(\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}\right) - \left(\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}\right)^2} \tag{4.38}$$

These formulas provide the gradient estimation components g_x and g_y, accounting for the uncertainties in the measurements.

Uncertainity in gradient estimation

The Cramer-Rao bound (CRB) provides a lower bound on the variance of any unbiased estimator [98, 268, 269]. For our problem, we can derive the CRB to quantify the uncertainty in the gradient measurement $\delta\mathbf{g}$.

To derive the CRB, we first need to compute the Fisher information matrix (FIM). For a set of measurements $\{c_i\}$ with uncertainties $\{\delta c_i\}$ and the simplified linear profile (4.21), the FIM is given by:

$$\mathcal{I}(\mathbf{g}) = \sum_{i=1}^{N} \frac{1}{\delta c_i^2} \begin{pmatrix} x_i^2 & x_i y_i \\ x_i y_i & y_i^2 \end{pmatrix} \tag{4.39}$$

which we reformulate it as

$$\mathcal{I}(\mathbf{g}) = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix}, \tag{4.40}$$

where

$$I_{xx} = \sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}, \quad I_{xy} = \sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}, \quad I_{yy} = \sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2} \tag{4.41}$$

The CRB states that the covariance matrix of any unbiased estimator $\hat{\mathbf{g}}$ is bounded from below by the inverse of the FIM. The covariance matrix of $\hat{\mathbf{g}} = (g_x, g_y)^T$ is

$$\mathrm{Cov}(\hat{\mathbf{g}}) \geq \mathcal{I}(\mathbf{g})^{-1} \tag{4.42}$$

To find $\mathcal{I}(\mathbf{g})^{-1}$, we compute the inverse of the 2x2 matrix $\mathcal{I}(\mathbf{g})$:

$$\mathcal{I}(\mathbf{g})^{-1} = \frac{1}{I_{xx} I_{yy} - I_{xy}^2} \begin{pmatrix} I_{yy} & -I_{xy} \\ -I_{xy} & I_{xx} \end{pmatrix}, \tag{4.43}$$

Hence, the variances (uncertainties) of g_x and g_y are the diagonal elements of $\mathcal{I}(\mathbf{g})^{-1}$:

$$(\delta g_x)^2 \geq \left(\mathcal{I}(\mathbf{g})^{-1}\right)_{11} = \frac{I_{yy}}{I_{xx} I_{yy} - I_{xy}^2} \tag{4.44}$$

$$(\delta g_y)^2 \geq \left(\mathcal{I}(\mathbf{g})^{-1}\right)_{22} = \frac{I_{xx}}{I_{xx} I_{yy} - I_{xy}^2} \tag{4.45}$$

substituting (4.41) into (4.44), we obtain the standard deviation to be

$$\delta g_x = \sqrt{\frac{\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}}{\left(\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}\right)\left(\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}\right) - \left(\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}\right)^2}} \tag{4.46}$$

$$\delta g_y = \sqrt{\frac{\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}}{\left(\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2}\right)\left(\sum_{i=1}^{N} \frac{y_i^2}{\delta c_i^2}\right) - \left(\sum_{i=1}^{N} \frac{x_i y_i}{\delta c_i^2}\right)^2}} \tag{4.47}$$

These expressions provide the lower bounds on the standard deviations of the estimates of g_x and g_y, taking into account the uncertainties in the ligand concentration measurements.

Simplified formulas for radial symmetry

The resulting equations become cumbersome, but due to the symmetries of our system, we can simplify them.

Assuming the cell sensors sit on a circle of radius a such that $x_i^2 + y_i^2 = a^2$ (see Fig. 4.6), we have

$$\sum_{i=1}^{N}(x_i^2 + y_i^2) = Na^2 \tag{4.48}$$

which, due to the sensors being symmetrically distributed, we also obtain the following relations

$$\sum_{i=1}^{N} x_i^2 = \sum_{i=1}^{N} y_i^2 = \frac{Na^2}{2} \tag{4.49}$$

$$\sum_{i=1}^{N} x_i y_i = 0 \tag{4.50}$$

Furthermore, we assume that the uncertainty of measuring concentrations is the same for all sensors $\delta c_i = \delta_c$, which leads to

$$\sum_{i=1}^{N} \frac{x_i^2}{\delta c_i^2} = \frac{1}{\delta c^2}\sum_{i=1}^{N} x_i^2 = \frac{1}{\delta c^2}\frac{Na^2}{2} \tag{4.51}$$

with this, we can simplify (4.37-4.38) to

$$g_x = \frac{2}{Na^2}\sum_{i=1}^{N} x_i c_i \tag{4.52}$$

$$g_y = \frac{2}{Na^2}\sum_{i=1}^{N} y_i c_i \tag{4.53}$$

and similarly, the uncertainties (4.46-4.47) to

$$\delta g_x = \sigma_g = \sqrt{\frac{2\delta c^2}{Na^2}} \tag{4.54}$$

$$\delta g_y = \sigma_g = \sqrt{\frac{2\delta c^2}{Na^2}} \tag{4.55}$$

Chemotactic Index (CI)

Based on the derived gradient estimation, we can determine the expected chemotactic index (CI) of a cell that, at each step, decides to move toward the estimated gradient direction. The CI is defined as the mean cosine similarity of the direction of movement and the true gradient direction, which we can calculate with

$$CI = \int \int_{-\infty}^{\infty} \frac{\hat{\mathbf{g}} \cdot \mathbf{g}}{\|\hat{\mathbf{g}}\|\|\mathbf{g}\|} P(\hat{g}_x \hat{g}_y) d\hat{g}_x d\hat{g}_y, \tag{4.56}$$

where $\hat{\mathbf{g}}$ is the estimated gradient and $\mathbf{g}$ is the real gradient, where we assume the estimated gradients are Gaussian-distributed such that the probability is

$$P(\hat{g}_x, \hat{g}_y) = \frac{1}{\sqrt{2\pi}\delta g_x} \exp\left[\frac{-(\hat{g}_x - g_x)^2}{2\delta g_x^2}\right] \frac{1}{\sqrt{2\pi}\delta g_y} \exp\left[\frac{-(\hat{g}_y - g_y)^2}{2\delta g_y^2}\right], \tag{4.57}$$

Furthermore, assuming the real gradient to only be on the x-direction, the dot product simplifies to:

$$\hat{\mathbf{g}} \cdot \mathbf{g} = \hat{g}_x \cdot g_x + \hat{g}_y \cdot 0 = \hat{g}_x \cdot g_x \tag{4.58}$$

with magnitudes

$$\|\hat{\mathbf{g}}\| = \sqrt{\hat{g}_x^2 + \hat{g}_y^2}, \quad \|\mathbf{g}\| = g_x \tag{4.59}$$

by including (4.57), (4.58) and (4.59) to (4.56) we obtain

$$CI = \int \int_{-\infty}^{\infty} \frac{\hat{g}_x g_x}{\sqrt{\hat{g}_x^2 + \hat{g}_y^2}} \frac{1}{\sqrt{2\pi}\delta g_x} \exp\left(\frac{-(\hat{g}_x - g_x)^2}{2\delta g_x^2}\right) \frac{1}{\sqrt{2\pi}\delta g_y} \exp\left(\frac{-(\hat{g}_y)^2}{2\delta g_y^2}\right) d\hat{g}_x d\hat{g}_y \tag{4.60}$$

Despite its apparent complexity, we can solve this integral analogously to Endres and Wingreen [13] and obtain

$$\mathrm{CI} = \sqrt{\frac{\pi z}{2}} e^{-z} \left[I_0(z) + I_1(z)\right], \tag{4.61}$$

where $I_{0(1)}$ are first (second)-order modified Bessel functions, and, differently to the derivation using the physical limits of sensing, our z is given by

$$z = \frac{Na^2}{8} \frac{g_x^2}{\delta c^2} = \frac{Na^2 k 5\pi}{24} \mathrm{SNR} \tag{4.62}$$

where we have use the uncertainty δc expression from Berg and Purcell [10] for the error in measurement

$$\delta c = \frac{3c_0}{5\pi DT} \tag{4.63}$$

Chapter 5

On receptors location

The following chapter contains the paper:

Receptors cluster in high-curvature membrane regions for optimal spatial gradient sensing

Albert Alonso[1], *Robert G. Endres*[2] *and Julius B. Kirkegaard*[1,3]

1. Niels Bohr Institute, University of Copenhagen, 2100, Copenhagen, Denmark.
2. Department of Life Sciences and Centre for Integrative Systems Biology and Bioinformatics, Imperial College, London, United Kingdom.
3. Department of Computer Science, University of Copenhagen, 2100, Copenhagen, Denmark.

Under review in: *Physical Review Letters* (PRL)
Pre-print server: https://arxiv.org/abs/2409.09342

Author Contributions:
Conceptualisation, Methodology, Validation, Investigation, Data Management, Visualisation, Formal Analysis, Software, Writing Original Draft, Writing Review & Editing. It *harmonically* perturbs me how smooth this one turned out...

Letter reformatted to fit the style of this thesis.

Abstract

Information from cell surface receptors is crucial for processes that require signal processing and sensing of the environment. Here, we investigate the optimal placement of such receptors through a theoretical model that minimizes uncertainty in gradient estimation. Without needing to account for physical limitations and biochemical constraints, we reproduce the emergence of clusters that closely resemble those observed in real cells. In perfect spherical surfaces, optimally placed receptors spread uniformly. When perturbations break their symmetry, receptors cluster in regions of high curvature, massively reducing estimation uncertainty. This agrees with mechanistic models that minimize elastic preference discrepancies between receptors and cell membranes. We further extend our model to dynamic environments: time-dependent cell shape changes, motile receptors, and external flow perturbations. Our findings provide a simple and utilitarian explanation for receptor clustering at high-curvature regions, especially relevant at the fundamental limits of sensing.

5.1 Introduction

Cells rely on their ability to detect and respond to environmental cues for essential biological processes including chemical gradient sensing in chemotaxis, wound healing, and embryonic development. It is only logical that the spatial information gathered by the cells is highly dependent on the positioning of receptors on the cell surface, and consequently on cell shape itself. Hence, cells can actively influence sensing performance, as exemplified by yeast cells localizing receptors such as Ste2 to shmoo tips when projecting up mating pheromone gradients [270]. In many biological systems, receptors are observed to cluster, particularly in regions of high membrane curvature, avoiding the mismatch of the preferred curvatures of membrane and embedded proteins [271], or due to biochemical constraints such as attractive protein-protein interactions, facilitating chemical reactions and efficient signaling [272–274]. Receptor clustering also occurs purely by stochastic self-organization, such as driven by cell growth, receptor diffusion, and capture [275]. These studies explain the mechanistic algorithms by which clusters are formed. However, what other evolutionary advantages do receptor localization and clustering provide to cells with, especially when confronted with noisy spatial signals?

Here, we develop a simple theoretical model to study the relationship between receptor distribution, cell shape, and gradient sensing efficiency in arbitrary 3D geometries. By inducing shape perturbations, we observe that clusters emerge naturally to minimize the uncertainty of the spatial gradient estimation, without any need to explicitly invoke receptor interactions or membrane information. Crucially, the clusters that result from our model are localized in high curvature regions in agreement with both experimental studies on prokaryotic [127, 130, 276] and eukaryotic [7, 277–279] systems, as well as theoretical studies that model receptor interactions and membrane curvature sensing [280–282]. Finally, we extend the model in two key directions: First, we study how receptors move in response to dynamic changes in the cell body, such as cellular pro-

trusion, e.g. actin-driven filopodia and pseudopods [3, 128]. Second, we elucidate how receptor localization ideally responds to flow of the surrounding fluid, which produces intricate patterns that may have implications in the context of synthetic biology [283], biosensor designs [284], and robotics at the physical limits of sensing [71, 267].

5.2 Model

Consider a cell placed inside a three-dimensional diffusive environment with a linear gradient concentration profile

$$c(x, y, z) = c_0 + g_x x + g_y y + g_z z = \boldsymbol{x} \cdot \boldsymbol{g} \tag{5.1}$$

where $\boldsymbol{x} = (1, x, y, z)$ is extended positional coordinates and $\boldsymbol{g} = (c_0, g_x, g_y, g_z)$ is the background concentration and gradient values.

We assume receptors to be *perfect instruments* [10] for counting molecules, and the cellular internal mechanisms of biochemical signaling pathways to perfectly process the information from the n surface receptors to distill the best gradient estimator $\widehat{\boldsymbol{g}}$. Assuming a shallow gradient ($R|\nabla c| \ll c_0$), we can model a cell measurement as a multivariate Gaussian,

$$P(\boldsymbol{c}) = \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}|}} e^{-\frac{1}{2}(\boldsymbol{c}-\mathbf{X}\cdot\boldsymbol{g})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{c}-\mathbf{X}\cdot\boldsymbol{g})}, \tag{5.2}$$

where $\boldsymbol{\Sigma}$ is the covariance matrix between measurements $\boldsymbol{c} = (c_1, \ldots, c_n)$, and $\mathbf{X}$ is a positional matrix in $\mathbb{R}^{n\times 4}$ where each row is the extended position vector of a receptor at the surface of the cell. Assuming that the concentration fluctuations are diffusion-limited, we model the individual uncertainty of each measurement with Berg and Purcell noise [10], letting each receptor make an average measurement over a measuring time τ. This reduces the measurement variance [10, 13], but also introduces the possibility of rebinding, thus making the measurements of nearby receptors correlated. Taking all this into account we use

$$\Sigma_{i,j} = \frac{2a^3 V}{D\tau(5a + 6\Delta)} (\boldsymbol{x}_i + \boldsymbol{x}_j) \cdot \boldsymbol{g}. \tag{5.3}$$

where $\Delta = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ is the distance between receptors, D is the molecular diffusion coefficient, τ the cell measuring time, a and V the receptor effective radius and volume, respectively (see Appendix A2 for a detailed derivation). In our formulation, a determines the typical correlation distance of the receptors.

We are interested in optimizing the uncertainty of the gradient estimation as a function of the receptor locations. The Cramer-Rao bound provides a lower bound on the variance of any unbiased estimator [98, 268, 269],

$$\mathrm{Cov}(\widehat{\boldsymbol{g}}) \geq \mathcal{I}(\mathbf{X}, \widehat{\boldsymbol{g}})^{-1}, \tag{5.4}$$

where $\mathcal{I}$ is the Fisher Information Matrix, which for our multivariate Gaussian evaluates to

$$\mathcal{I}_{m,k} = \frac{\partial c^T}{\partial \widehat{g}_m} \Sigma^{-1} \frac{\partial c}{\partial \widehat{g}_k} + \frac{1}{2} \mathrm{tr}\left(\Sigma^{-1} \frac{\partial \Sigma}{\partial \widehat{g}_m} \Sigma^{-1} \frac{\partial \Sigma}{\partial \widehat{g}_k} \right). \tag{5.5}$$

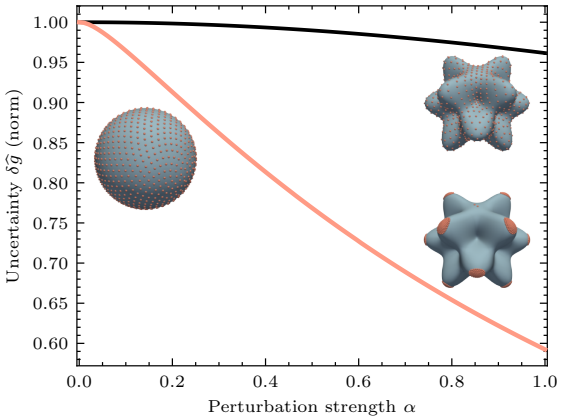


Figure 5.1: Uncertainty of gradient estimation by the cell, normalized by the initial value of $\delta\widehat{g}(X_0)$, of a spherical cell perturbed by the harmonic Y_6^4 to form protrusions, in the case of uniformly spread receptors (black) or when placed at their optimal location for each α (orange).

where we use Eq. (5.1) with the real gradient $\boldsymbol{g}$ to obtain the expected concentration c, and, using Eq. (5.3),

$$\frac{\partial \Sigma_{i,j}}{\partial \widehat{g}_k} = \frac{2a^3 V}{D\tau(5a + 6\Delta)} \left(X_{i,k} + X_{j,k}\right). \tag{5.6}$$

Our aim is to find the optimal spatial configuration $\mathbf{X}^*$ of the n surface receptors to minimize the uncertainty of the gradient estimations,

$$\delta\widehat{g}(\mathbf{X}) = \mathrm{tr}(\mathrm{Cov}(\widehat{\boldsymbol{g}}; \mathbf{X})) \tag{5.7}$$

such that the optimal spatial distribution fulfills $\mathbf{X}^* = \arg\min_{\mathbf{X}\in\mathcal{S}^n}(\delta\widehat{g})$ where $\mathcal{S}$ is the surface of the cell. We solve the optimization problem with differentiable programming and local gradient descent.

5.3 Results

5.3.1 Clusters induced by symmetry breaking

Using spherical coordinates $(r,\ \theta,\ \phi)$, we define the cell surface by $r = f(\theta, \phi)$. Our results are valid for any positive cell-shape function f, and we exemplify using spherical harmonics,

$$r(\theta, \phi) = r_0 + \alpha \cdot \mathrm{Re}\left[Y_\ell^m(\theta, \phi)\right], \tag{5.8}$$

which have previously been shown to be effective in describing cell shapes [285]. Y_ℓ^m is the spherical harmonic of degree ℓ and order m, r_0 is the base spherical radius of the cell [1], and α is a free parameter that controls the strength of the harmonic perturbations, meant to induce surface irregularities.

[1] We assume $r_0 = 1$ and set the length scale to be on cell size.

Using the Golden Spiral algorithm ("the Fibonacci sphere") [286], we initialize the receptors to be approximately uniformly distributed on the surface before optimization. In the case of the unperturbed cell ($\alpha = 0$), this provides a very evenly distributed distribution along the surface (Fig. 5.1). We note that the more perturbed the cell surface is from a sphere, the less perfectly distributed the receptors will be at the start of the optimization.

As the cell has to estimate the gradient with three independent directional components, the optimal distribution requires positioning the receptors to enable gradient estimation in any direction. On a sphere this results in uniformly spaced receptors. However, we observe that $\delta\widehat{g}$ is very insensitive to the precise localization of the receptors on a sphere, as long they are mostly spread. For symmetric cell shapes, we thus expect even small effects coming from biochemical and physical factors would dominate the minor changes to receptor efficiency.

Once the sphere is perturbed and the symmetry is broken ($\alpha > 0$), the optimal distribution is no longer uniform. Instead a clustered distribution emerges, where receptors move to gather at the tips of protrusions (Fig. 5.1). This clustering aligns with experimental observations [287, 288], suggesting that receptors aggregate in areas of higher curvature. Notably, this emerges simply by breaking the surface symmetry.

By modifying the value of α, we can smoothly perturb the spherical shape of the cell. Previous studies have considered the impact of cell shape and size on gradient estimation [2, 118, 262, 289, 290]. These studies find that elongated or protruded shapes can improve gradient estimation without accounting for the repositioning of receptors. This type of effect is reproduced in Fig. 5.1 (black curve), showing an improvement in estimation accuracy. Additionally placing the receptors in their optimal locations leads to a huge additional improvement (orange curve), thus showing the true potential of shape deformations on gradient estimation. In other words, any estimation of the improvement in accuracy resulting from deforming cell shapes will drastically underestimate the effect if it does not account for receptor localization.

5.3.2 Minimal curvature energy regions are optimal for gradient sensing

Previous work has shown that clustering occurs predominantly in high curvature regions [271, 291], corresponding to energy minima of the elastic membrane energy [292] induced by the receptors,

$$E_m = \frac{k_c}{2}(H - H_0)^2 + \bar{k}\,K. \tag{5.9}$$

Here, H_0 is the preferred mean curvature of a receptor [2], and H and K are the mean and Gaussian curvature of the membrane, the latter of which can typically be neglected for cells [271]. Assuming $H_0 < H$ everywhere [278], energy is minimized when receptors

[2] We assume our receptors to be a coarse-grained representation of small clusters of binding ligands, as there are orders of magnitude more receptors in real systems than in the ones simulated here

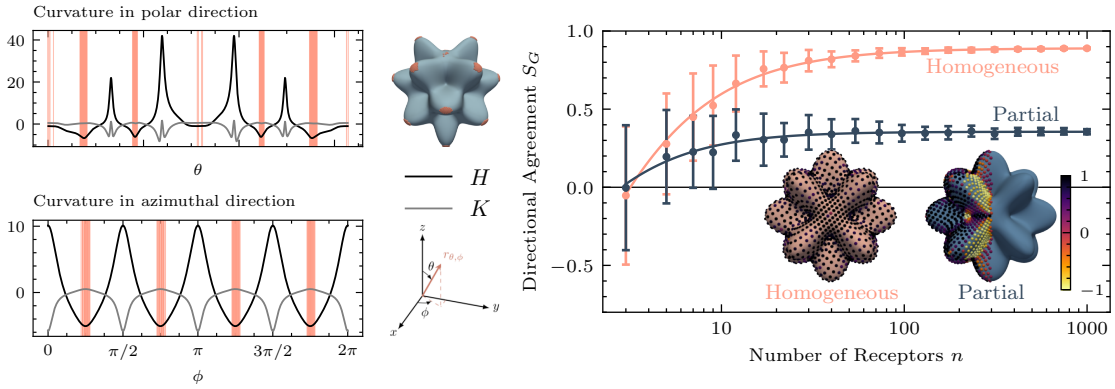


Figure 5.2: Optimal receptor regions match those of minimal elastic energy on an spherical cell perturbed by the Y_7^4 harmonic. (a) Mean curvature (H) in black and Gaussian curvature (K) in gray along the surface of the cell shown on the right, at fixed azimuthal angle $\phi = 0$ rad (top) and polar angle $\theta = 0.4\pi$ rad (bottom). A single orange vertical line is placed at each angle where a receptor is found along the surface path. (b) Direction alignment between the gradients from elastic energy and uncertainty minimization Eq. (5.10), for homogeneously spread receptors (orange) and receptors located on half the surface (blue). Error bars show the standard deviation of 100 realizations, where receptor position is uniformly sampled at the entire or half cell surface. The diagrams show the individual alignment value (cosine similarity) of each receptor indicated by the color.

localize at the highest negative mean curvature regions. This approach shows agreement with experimental observations and explains why receptors are stable at a certain locations on the membrane [277, 278, 282, 288].

Interestingly, in our study, deformation of the cell surface, which disrupts surface symmetry, results in redistribution of receptors exactly to the minimal energy regions of Eq. (5.9), as shown in Fig. 5.2a. In a curvature driven model, additional receptor-receptor interactions needs to be accounted for [271], whereas here clustering is not directly modeled; rather, it emerges through uncertainty minimization, with a cluster size set by the sensing correlation length. Clusters do smoothly merge with each other as the effective receptor size a increases (see Appendix A6). Yet, both models yield the same final localization.

Given that our resulting cluster locations match with regions of minimal energy (Fig. 5.2a), we argue that gradient estimation is optimized when receptors optimize for membrane curvature, driven by elastic membrane energy or alternative biophysical forces. Thus, we find that the mechanistic paradigm (energy minimization) that explains the physical reasons for cluster formation is intimately linked to a utilitarian evolutionary mechanism (enhanced gradient sensing), highlighting the *mechanical intelligence* [7] of receptor localization. Notably, although global information is necessary for optimizing gradient estimation, the curvature information is strictly local. Thus, the agreement between receptor motion in the two mechanisms is only valid under certain conditions. To quantify this, we calculate the alignment between receptor directions in both models

using the cosine similarity between the gradients,

$$S_G = \frac{\nabla\delta\widehat{g}\cdot\nabla E_m}{\|\nabla\delta\widehat{g}\|\,\|\nabla E_m\|}. \tag{5.10}$$

As seen in Fig. 5.2b, the alignment is strong when receptors are many and spread uniformly on the cell. In contrast, when receptors are few and unevenly spread, e.g., only half the cell is covered in them, following the local energy gradients yield suboptimal gradient estimation, demonstrated by their misaligned gradient vectors.

5.3.3 Dynamic receptors for dynamic membranes

Moving beyond optimal *static* receptor locations, we study the consequences of membrane dynamics and the ensuing response of the receptor localization. Many biological systems adapt to membrane deformations by relocating their sensors [278, 293, 294], which can enhance chemotactic efficiency [3, 295]. Thus, we allow receptors not only to move as a result of membrane deformations but also as an independent mechanism. We model receptor movement as influenced by an effective potential that mirrors the gradient of the estimation variance, $U \sim \delta\hat{g}$, Eq. (5.7), which may in turn be related to the curvature energy of the membrane and receptors. We assume the receptor motion to be over-damped and restricted to a maximum speed u. As the receptors are constrained to move on the two-dimensional star-convex cell surface, we formulate the gradient dynamics in spherical angles,

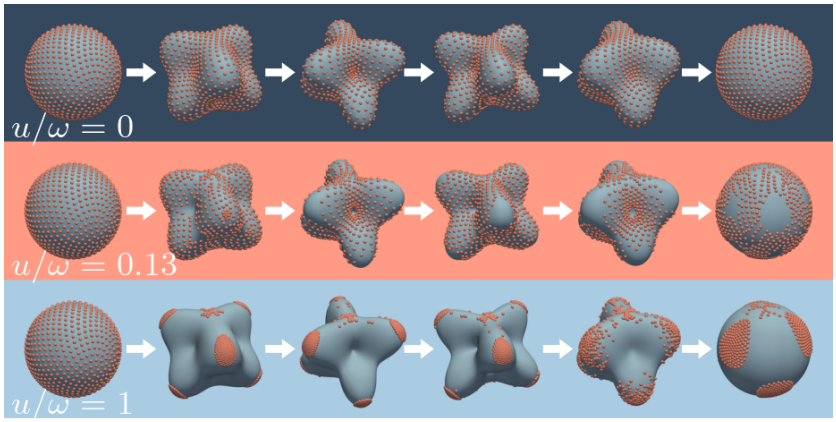
$$\partial_t\begin{pmatrix}\phi\\ \theta\end{pmatrix} = -\frac{u}{\max(\epsilon, |\mathrm{d}\boldsymbol{r}|)}\nabla_{\phi,\theta}U, \tag{5.11}$$

where the normalization from

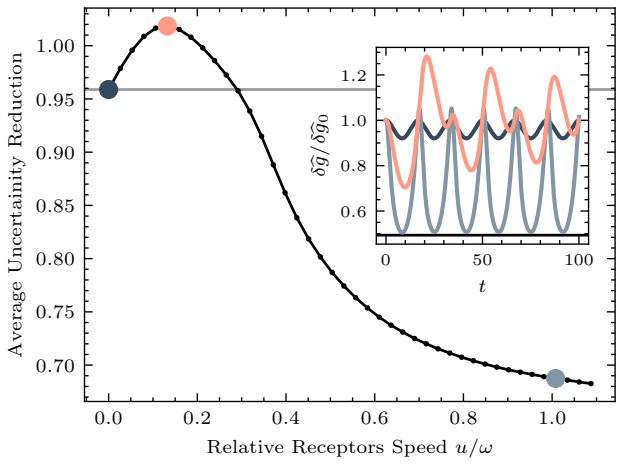
$$|\mathrm{d}\boldsymbol{r}|^2 = \sum_{q\in\{x,y,z\}}\left(\frac{\partial q}{\partial\phi}\frac{\partial U}{\partial\phi} + \frac{\partial q}{\partial\theta}\frac{\partial U}{\partial\theta}\right)^2 \tag{5.12}$$

ensures that u is the maximum speed in Cartesian coordinates, and ϵ is a small value that allows receptors to stop when the gradients are sufficiently small.

In order to study the effect of dynamical receptors, we perturb a perfect sphere to grow and retract protrusions, by defining the perturbation strength as an oscillating function $\alpha \propto \sin(\omega t)$, where ω is the speed of the membrane motion (see [118, 285] for examples of oscillatory behavior in cells). Thus, we perform simulations for increasing values of u/ω, and observe the effects of finite-speed moving receptors on gradient estimation as the membrane of the cell surface is deformed (Fig. 5.3). When receptors are immobile ($u/\omega = 0$), and thus only move due to membrane changes, the performance of the estimation is only slightly affected by the perturbations as seen in Fig. 5.3b. Paradoxically, when receptors are allowed to move at low speeds ($u/\omega = 0.13$), the average accuracy decreases compared to when the receptors are not motile. This is because the receptors move according to instantaneous membrane shape information, and are



(a)



(b)

Figure 5.3: Effects of reacting motile receptors to membrane dynamics. (a) Frames from three independent simulations of the motion form protrusions caused by the harmonic Y_4^3 perturbation, where the receptors are immobile (top), have low speed (middle) or move fast (bottom) as the membrane oscillates. (b) Average gradient estimation error on an oscillating motion where protrusions are grown and retracted (3 full oscillations). The inset shows the oscillations in error for the three cases displayed in (a) as the membrane changes shape, with the black line showing the reduction for *static* optimization.

not fast enough to reach the tip of protrusions and form clusters before those tips have disappeared again. As the speed of the receptors becomes faster, clusters form, and the performance converges towards the same optimal spatial gradient sensing as that of static shapes.

5.3.4 Fluid Flow distorts cluster morphology

The morphology of the observed clusters has until now been a conglomerate of receptors spread isometrically. This is due to the fact that the environment has been symmet-

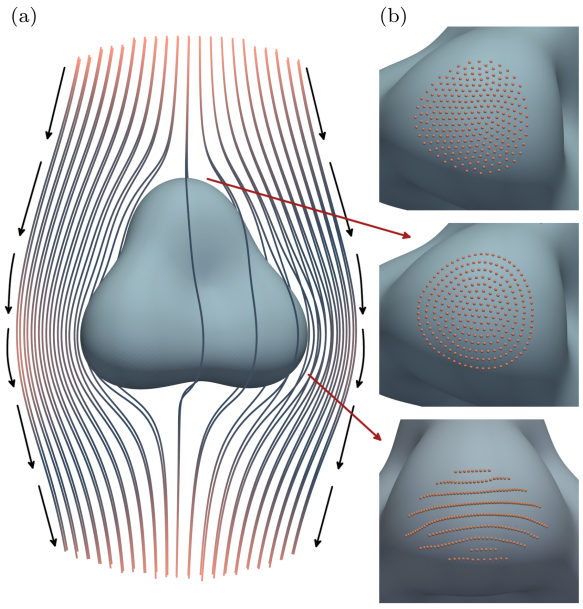


Figure 5.4: Cluster formation in the presence of flow for a cell perturbed by the Y_3^2 harmonic. (a) Streamlines of the simulated Stokes flow around the cell. (b) Resulting cluster on an environment without flow (top), cluster formation under the effect of front-facing fluid flow (middle) and cluster morphology for a side protrusion affected by flow (bottom).

ric, i.e. we have assumed the cell to be static and in a shallow gradient. However, chemotactic cells are naturally motile — motility being the typical reason to optimize gradient estimation. To that end, we study how the resulting cluster formation is affected when flow is accounted for; this flow being the result either of external motion or cell motility.

With flow, chemical molecules not only diffuse but are also advected along the flow. This changed the covariance between receptors. Under certain simplifying assumptions (see Appendix A3 for derivation), we find

$$\hat{\Sigma}_{i,j} = \frac{2a^3 V}{D\tau(5a+6\Delta)} e^{\frac{-\|\boldsymbol{v}\|\sqrt{a^2+\Delta^2}}{2D}} (\boldsymbol{x}_i e^{-\lambda} + \boldsymbol{x}_j e^{\lambda}) \cdot \boldsymbol{g}, \tag{5.13}$$

where $\lambda = \boldsymbol{v} \cdot \boldsymbol{\Delta}/2D$, and $\boldsymbol{v}$ is the flow, which we take to be the Stokes flow that results from moving the deformed cells at a constant speed. We compute this numerically using the boundary element regularized Stokeslet method [296, 297].

We observe that the cluster morphology is drastically affected by the inclusion of flow (Fig. 5.4). Interestingly, because of the higher correlation of the receptors in the direction of the fluid flow, given by Eq. 5.13, receptors in clusters are no longer isotropically spread. Instead, they converge to form lines perpendicular to the flow field.

5.4 Discussion

In this Letter, we found the optimal placement of cell-surface receptors by developing a theoretical model that minimizes uncertainty in gradient estimation. Our results show that, in spherical cells, optimal receptor location is uniform across the surface, but once symmetry is broken through shape perturbations, clusters of receptors naturally emerge in regions of high negative mean curvature. These clusters form without explicitly including curvature information, yet are consistent with the results of elastic energy minimization. This supports the idea that even though receptor distribution is governed by biophysical principles, the final distribution agrees with an evolutionary trait to maximize sensing accuracy.

We observed that when accounting for receptor localization, the effect of shape deformations on gradient estimation accuracy is massively improved, with the effect of receptor localization being more than an order of magnitude larger than the effect stemming from the shape deformation on its own. When the shape deformations are dynamic, we showed that surface receptor motility can maintain the massive improvement in estimation accuracy, provided that the receptors move *sufficiently* fast. This would indeed be the case e.g. during cell migration of social amoeba [118] and T cells [285] with observed shape deformations on time-scales of $3-4$ min, and receptor diffusivities of $\sim 0.1\mu m^2/s$ [298], as long as there are sufficient numbers of receptors to avoid long distances for diffusion ($\geq 1\mu m$). In contrast, if receptors were too slow, receptor motility can have adverse effects on estimation accuracy. Note that gradient sensing, which occurs over a time-scale of seconds [13], is much faster than the time required by the receptors to relocalize. Finally, we have shown how optimal clustering is affected by fluid flow, leading to separation of receptors in the direction of the flow to reduce receptor-receptor correlations. Taken together, our work shows that cells can strongly improve the accuracy of spatial sensing by actively regulating receptor placement and clustering.

Appendix

A1 Curvature Calculation

We can obtain the mean curvature [299] at any point on the surface of our cell, defined by the coordinates (θ, ϕ) by using

$$H(\theta,\phi) = \frac{(\nabla F)\cdot \mathbf{H}_F \cdot (\nabla F)^T - |\nabla F|^2 \Delta F}{2\,|\nabla F|^3}, \tag{5.14}$$

where F is the implicitly defined surface given by

$$F \;=\; x^2 + y^2 + z^2 - r(\theta,\phi)^2 \;=\; 0. \tag{5.15}$$

using

$$r(\theta,\varphi) = a + \alpha_t \cdot \mathrm{Re}\left[Y_\ell^m(\theta,\phi)\right], \tag{5.16}$$

for the radius, and

$$x = r\sin\theta\cos\phi \tag{5.17}$$

$$y = r\sin\theta\sin\phi \tag{5.18}$$

$$z = r\cos\theta \tag{5.19}$$

for the Cartesian coordinates, to backpropagate. ∇F, $\mathbf{H}_F$ and ΔF are the Gradient, the Hessian and the Laplacian on the cartesian coordinates, respectively.

Analogously, we can calculate the Gaussian curvature with the following expression

$$K(\theta,\phi) = \frac{\nabla F \cdot \mathbf{H}_F^* \cdot \nabla F^T}{|\nabla F|^4} \tag{5.20}$$

where $\mathbf{H}_F^*$ is the adjoint of the Hessian.

A2 Spatial Covariance of Receptors

Consider N particles that form an average density field

$$c(\boldsymbol{x}) = c_0 + \boldsymbol{g}\cdot\boldsymbol{x} \tag{5.21}$$

We mark two regions R and R' (receptors) at $\boldsymbol{x}$ and $\boldsymbol{x'}$, each spheres of radius a and volume $V = 4/3\pi a^3$. These receptors measure a time τ and report the average number of particles M, M' that were inside their volume during that time. We can define these random variables by the use of indicator functions:

$$M = \frac{1}{\tau}\int \sum_{i=1}^{N} I_i(t)\,\mathrm{d}t \tag{5.22}$$

$$M' = \frac{1}{\tau}\int \sum_{i=1}^{N} I'(t)\,\mathrm{d}t \tag{5.23}$$

Here I_i is an indicator function which is equal to one if particle i is inside receptor A, and likewise for I_i'.

On average we have e.g.

$$\langle M\rangle = \sum_i \frac{1}{\tau}\int \langle I_i(t)\rangle \,\mathrm{d}t = N\langle I_i\rangle \approx c(\boldsymbol{x})\,V, \tag{5.24}$$

with a variance

$$\begin{aligned}\langle M^2\rangle &= \frac{1}{\tau^2}\iint \sum_i\sum_j \langle I_i(t)I_j(s)\rangle \,\mathrm{d}t\,\mathrm{d}s \\ &= \sum_i \frac{1}{\tau^2}\iint \langle I_i(t)I_i(s)\rangle \,\mathrm{d}t\,\mathrm{d}s \;+\sum_{i\neq j}\frac{1}{\tau^2}\iint \langle I_i(t)\rangle\langle I_j(s)\rangle \,\mathrm{d}t\,\mathrm{d}s \\ &\approx \frac{N}{\tau^2}\iint \langle I_i(t)I_i(s)\rangle \,\mathrm{d}t\,\mathrm{d}s \;+c(\boldsymbol{x})^2\,V^2,\end{aligned} \tag{5.25}$$

where we used $N(N-1)\approx N^2$. Separating the integrals into $t<s$ and $s<t$ and exploiting symmetry, we have

$$\begin{aligned}N\iint \langle I_i(t)I_i(s)\rangle \,\mathrm{d}t\,\mathrm{d}s &= 2N\int_0^\tau\int_0^t \langle I_i(t)I_i(s)\rangle \,\mathrm{d}s\,\mathrm{d}t \\ &= 2N\int_0^\tau\int_0^t \langle I_i(s)\rangle\langle I_i(t)|I_i(s)\rangle \,\mathrm{d}s\,\mathrm{d}t \\ &= 2\,c(\boldsymbol{x})V\int_0^\tau\int_0^t \langle I_i(t)|I_i(s)\rangle \,\mathrm{d}s\,\mathrm{d}t.\end{aligned} \tag{5.26}$$

$\langle I_i(t)|I_i(s)\rangle$ is the probability that a particle that is inside the region at time s is also inside the region at time t. Assuming we are far from boundaries that set up the gradient in c, we can use the Green's function

$$\mathbb{P}[\boldsymbol{x}_i(t)|\boldsymbol{x}_i(s)] = \frac{1}{(4\pi D(t-s))^{3/2}}\exp\!\left(-\frac{|\boldsymbol{x}_i(t)-\boldsymbol{x}_i(s)|^2}{4D(t-s)}\right) \quad , \quad (t\geq s), \tag{5.27}$$

and we have

$$\langle I_i(t)|I_i(s)\rangle = \frac{3}{4\pi a^3}\iiint\limits_{V}\iiint\limits_{V} \mathbb{P}[\boldsymbol{x}_i(t)|\boldsymbol{x}_i(s)]\,\mathrm{d}V^2, \tag{5.28}$$

where we took a uniform distribution for $\boldsymbol{x}_i(s)$.

Swapping the order of the spatial and temporal integrals, the first integral we need to evaluate is

$$\int_0^\tau\int_0^t \frac{1}{(t-s)^{3/2}}\exp\!\left(-\frac{\zeta}{t-s}\right)\,\mathrm{d}s\,\mathrm{d}t = \int_0^\tau \Gamma[1/2,\zeta/t]/t\,\mathrm{d}t \approx \sqrt{\frac{\pi}{\zeta}}\tau \tag{5.29}$$

where we assumed τ large compared to the diffusive timescale, and wrote

$$\zeta = \frac{|\boldsymbol{x}_i(t)-\boldsymbol{x}_i(s)|^2}{4D}. \tag{5.30}$$

We now need to carry out the spatial integrals of Eq. (5.28). The first integral corresponds to evaluating the potential of a uniform spherical charge (evaluated inside the sphere)

$$\iiint \frac{1}{|\boldsymbol{x}_i(t) - \boldsymbol{x}_i(s)|}\,\mathrm{d}V = 2\pi a^2 - \frac{2\pi}{3}|\boldsymbol{x}_i(t)|^2, \tag{5.31}$$

from which we find

$$\iiint (2\pi a^2 - \frac{2\pi}{3}|\boldsymbol{x}_i(t)|^2)\mathrm{d}V = \frac{32\pi^2 a^5}{15}. \tag{5.32}$$

Putting the terms back together we find

$$\int_0^\tau \int_0^t \langle I_i(t)|I_i(s)\rangle\,\mathrm{d}s\,\mathrm{d}t = \frac{3}{4\pi a^3}\,\frac{1}{(4\pi D)^{3/2}}\,\sqrt{4D\pi}\,\frac{32\pi^2 a^5}{15}\,\tau = \frac{2a^2\tau}{5D}, \tag{5.33}$$

and finally,

$$\langle M^2\rangle = c(\boldsymbol{x})^2\,V^2 + c(\boldsymbol{x})V\frac{4a^2}{5D\tau}. \tag{5.34}$$

This is the celebrated Berg-Purcell measurement noise [10].

We further need the correlation between receptors, i.e.

$$\begin{aligned}\langle MM'\rangle &= \frac{1}{\tau^2}\iint \sum_i \sum_j \langle I_i(t)I'_j(s)\rangle\,\mathrm{d}t\,\mathrm{d}s \\ &= \sum_i \frac{1}{\tau^2}\iint \langle I_i(t)I'_i(s)\rangle\,\mathrm{d}t\,\mathrm{d}s\; + \sum_{i\neq j}\frac{1}{\tau^2}\iint \langle I_i(t)\rangle\langle I'_j(s)\rangle\,\mathrm{d}t\,\mathrm{d}s \\ &\approx \frac{N}{\tau^2}\iint \langle I_i(t)I_i(s)'\rangle\,\mathrm{d}t\,\mathrm{d}s\; + c(\boldsymbol{x})c(\boldsymbol{x}')\,V^2,\end{aligned} \tag{5.35}$$

Following the same approach, we divide the integral into parts that satisfy $s \leq t$,

$$N\iint \langle I_i(t)I_i(s)\rangle\,\mathrm{d}t\,\mathrm{d}s = N\int_0^\tau\int_0^t \langle I_i(t)I'_i(s)\rangle\,\mathrm{d}s\,\mathrm{d}t + N\int_0^\tau\int_0^t \langle I_i(s)I'_i(t)\rangle\,\mathrm{d}s\,\mathrm{d}t. \tag{5.36}$$

And then, e.g.

$$\begin{aligned}N\int_0^\tau\int_0^t \langle I_i(t)I'_i(s)\rangle\,\mathrm{d}s\,\mathrm{d}t &= N\int_0^\tau\int_0^t \langle I_i(t)I'_i(s)\rangle\,\mathrm{d}s\,\mathrm{d}t \\ &= N\int_0^\tau\int_0^t \langle I'_i(s)\rangle\langle I_i(t)|I'_i(s)\rangle\,\mathrm{d}s\,\mathrm{d}t \\ &= c(\boldsymbol{x}')V\int_0^\tau\int_0^t \langle I_i(t)|I'_i(s)\rangle\,\mathrm{d}s\,\mathrm{d}t.\end{aligned} \tag{5.37}$$

Here, $\langle I_i(t)|I'_i(s)\rangle$ is the probability that a particle that is inside region R' at time s is inside region R at time t.

While the formula for $\langle M^2 \rangle$ assumes $\tau \gg a^2/D$, i.e. τ large compared to the time of diffusion over the receptor, to use the same approximation we must now further assume τ to be large compared to the time scale of diffusion over the cell [see section A4 for a derivation that does not assume large τ.]. With this, we continue with the same procedure as above, but now $\boldsymbol{x}_i(t)$ is outside the integration sphere, and we get

$$\iiint \frac{1}{|\boldsymbol{x}_i(t) - \boldsymbol{x}_i(s)|} \,\mathrm{d}V = \frac{4\pi a^3}{3\,|\boldsymbol{x}_i(t)|}. \tag{5.38}$$

Denote by $\Delta = \boldsymbol{x} - \boldsymbol{x}'$ the distance between the two receptor centers. Aligning our coordinate system such that this distance is on the z-axis, we have

$$\iiint \frac{1}{\sqrt{r^2 + \Delta^2 - 2r\Delta\cos\theta}} \,\mathrm{d}V = \frac{4\pi a^3}{3\,\Delta}. \tag{5.39}$$

Thus we find

$$\int_0^\tau \int_0^t \langle I_i(t) | I_i'(s) \rangle \,\mathrm{d}s\,\mathrm{d}t = \frac{3}{4\pi a^3} \frac{1}{(4\pi D)^{3/2}} \sqrt{4D\pi}\, \frac{4\pi a^3}{3}\, \frac{4\pi a^3}{3\,\Delta}\, \tau = \frac{a^3 \tau}{3D\Delta}, \tag{5.40}$$

and finally,

$$\langle M M' \rangle = c(\boldsymbol{x}) c(\boldsymbol{x}')\, V^2 + \frac{c(\boldsymbol{x}) + c(\boldsymbol{x}')}{2} V \frac{2a^3}{3D\Delta\tau}. \tag{5.41}$$

Note that this formula is only valid for $\Delta > 2a$, as we have assumed no overlap of the receptors. When there is overlap between the receptors, the integrals must be evaluated in overlapping and non-overlapping regions separately. Clearly, though, in the range $0 < \Delta < 2a$, the formulas must monotonically interpolate our results for $\Delta = 0$ and $\Delta \geq 2a$, and there are many suitable approximations, none of which give massive numerical changes. Here, we will approximate the combination with

$$\mathrm{cov}(M, M') = \frac{c(\boldsymbol{x}) + c(\boldsymbol{x}')}{2} V \frac{4a^3}{5a + 6\Delta} \frac{1}{D\tau}, \tag{5.42}$$

which is correct both at $\Delta = 0$ and asymptotically. An alternative such as

$$\mathrm{cov}(M, M') = \frac{c(\boldsymbol{x}) + c(\boldsymbol{x}')}{2} V \frac{1}{\max(2a, \Delta)} \left(\frac{8}{5} - \frac{7}{15a} \min(2a, \Delta) \right) \frac{1}{D\tau}, \tag{5.43}$$

is correct at $\Delta = 0$ and for all $\Delta \geq 2a$, but does not give any significant numerical changes.

A3 The Effect of Flow on the Covariance

If we include a constant flow $\boldsymbol{v}$, the Eq. (5.29) becomes

$$\mathcal{I} = \int_0^\tau \int_0^t \frac{1}{(t-s)^{3/2}} \exp\!\left(-\frac{\zeta_1 + \zeta_2 (t-s)^2 - 2\zeta_3 (t-s)}{t-s} \right) \mathrm{d}s\,\mathrm{d}t \tag{5.44}$$

where

$$\zeta_1 = \frac{|\boldsymbol{x}_i(t) - \boldsymbol{x}_i(s)|^2}{4D}, \quad \zeta_2 = \frac{|\boldsymbol{v}|^2}{4D} = \frac{v^2}{4D}, \quad \zeta_3 = \frac{\boldsymbol{v} \cdot (\boldsymbol{x}_i(t) - \boldsymbol{x}_i(s))}{4D}. \tag{5.45}$$

The two new terms comes from including the drift $\boldsymbol{v}(t-s)$ in the Green's function [Eq. (5.27)] of the advection-diffusion equation. Again, we look for the asymptotics in τ, and find to leading order

$$\mathcal{I} \approx e^{-2\sqrt{\zeta_1\zeta_2}}\, e^{-2\zeta_3} \sqrt{\frac{\pi}{\zeta_1}}\tau. \tag{5.46}$$

The two new terms have a fairly clear interpretation:

The first term accounts for the fact that once there is flow, variances are generally smaller as diffusive molecules stay a shorter time within receptors as they are swept along the flow. We approximate this effect to be constant within a receptor, and use $\langle|\boldsymbol{x}_i(t) - \boldsymbol{x}_i(s)|\rangle \approx \sqrt{a^2 + \Delta^2}$ and

$$e^{-2\sqrt{\zeta_1\zeta_2}} \approx e^{-\frac{v\sqrt{a^2+\Delta^2}}{2D}}. \tag{5.47}$$

The second term accounts for the directional drift of molecules. This effect is small compared to the previous one within a receptor, but between receptors it is significant. Thus, we approximate

$$e^{-2\zeta_3} \approx e^{-\frac{\boldsymbol{v}\cdot\boldsymbol{\Delta}}{2D}}. \tag{5.48}$$

The remaining integrals are then the same as in the previous section, and we end up with

$$\mathrm{cov}(M, M') = \frac{c(\boldsymbol{x})e^{-\frac{\boldsymbol{v}\cdot\boldsymbol{\Delta}}{2D}} + c(\boldsymbol{x}')e^{\frac{\boldsymbol{v}\cdot\boldsymbol{\Delta}}{2D}}}{2}\, V\, e^{-\frac{v\sqrt{a^2+\Delta^2}}{2D}} \cdot \frac{4a^3}{5a + 6\Delta}\, \frac{1}{D\tau}. \tag{5.49}$$

A4 Relaxing the Assumption of Large Measuring Time

The assumption of large measuring time is the typical assumption in the literature of sensing, and is what leads to the simple mathematical expressions of Berg and Purcell [10]. To get similar results, we need to assume the measuring time larger than that of diffusion time over the entire cell.

Here, we explore a different set of approximations to the integrals. Starting with Eq. (5.28), we now do the spatial integrals first, i.e. evaluate $\langle I_i(t)|I_i(s)\rangle$. Before $\boldsymbol{x}_i(s)$ was sampled uniformly within the receptor. Instead, we will here approximate it by saying that at time s all particles are exactly the center $\boldsymbol{x}$. Hence,

$$\begin{aligned}
\langle I_i(t)|I_i(s)\rangle &\approx \iiint \mathbb{P}[\boldsymbol{x}_i(t)|\boldsymbol{x}_i(s)]\,\mathrm{d}V \qquad (5.50)\\
&= \frac{1}{\sqrt{4\pi}(D(t-s))^{3/2}} \int_0^a r^2 \exp\left(-\frac{r^2}{4D(t-s)}\right)\mathrm{d}r \\
&= \mathrm{erf}\left(\frac{a}{2\sqrt{D(t-s)}}\right) - \frac{a}{\sqrt{\pi D(t-s)}} \exp\left(-\frac{a^2}{4D(t-s)}\right),
\end{aligned}$$

which, after temporal integration, yields

$$\langle M^2\rangle = c(\boldsymbol{x})^2\,V^2 + c(\boldsymbol{x})\,V\left[1 - \frac{2ae^{-a^2/4D\tau}(a^2+2D\tau)}{\sqrt{\pi}(D\tau)^{3/2}} + \frac{(a^4+4a^2D\tau-4D^2\tau^2)\,\mathrm{erfc}(a/\sqrt{4D\tau})}{D^2\tau^2}\right] \tag{5.51}$$

$$= c(\boldsymbol{x})^2\,V^2 + \xi c(\boldsymbol{x})\,V.$$

The term ξ behaves like $\xi \to 0$ for $\tau \to \infty$ and $\xi \to 1$ for $\tau \to 0$, but is in general a term that is fixed to a finite value $\in [0,1]$ for our simulations.

For the covariance, we approximate $|\boldsymbol{x}_i(t) - \boldsymbol{x}_i(s)|^2 \approx \Delta^2 + |\boldsymbol{x}_i(t)|^2$ and find

$$\langle I_i(t)|I_i'(s)\rangle \approx \iiint \mathbb{P}[\boldsymbol{x}_i(t)|\boldsymbol{x}_i(s)]\,\mathrm{d}V \tag{5.52}$$

$$\approx \exp\left(-\frac{\Delta^2}{4D(t-s)}\right)\left[\mathrm{erf}\left(\frac{a}{2\sqrt{D(t-s)}}\right) - \frac{a}{\sqrt{\pi D(t-s)}}\exp\left(-\frac{a^2}{4D(t-s)}\right)\right],$$

and then approximate the first term by its average

$$\exp\left(-\frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{4D(t-s)}\right) \approx \exp\left(-\frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{4D\tau}\right) + \frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{4D\tau}\mathrm{Ei}\left(-\frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{4D\tau}\right) \approx \exp\left(-\frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{2D\tau}\right). \tag{5.53}$$

Thus we end up with

$$\langle MM'\rangle = c(\boldsymbol{x})c(\boldsymbol{x}')\,V^2 + \frac{c(\boldsymbol{x})+c(\boldsymbol{x}')}{2}V\xi\exp\left(-\frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{2D\tau}\right), \tag{5.54}$$

where ξ is the term from above. Here, the correlation distance is set by $\sqrt{2D\tau}$, which is more realistic when τ is smaller than the time to diffuse over the cell.

Using these formulae, we find numerical results very similar to those of the main text, showing that the conclusions are quite general and not artifacts of specific assumptions.

A5 Volume Conservation

In the main text, we consider perturbed spherical cells. However, for simplicity, we do not account for volume conservation, which instead we explore here. To ensure that the perturbed sphere defined by $R(\theta,\phi) = r_0 + \alpha\cdot\mathrm{Re}\,[Y_\ell^m(\theta,\phi)]$ preserves its volume, we derive the adjustment needed as a function of α.

The volume of a sphere in spherical coordinates is given by:

$$V = \int_0^{2\pi}\int_0^{\pi}\int_0^{R(\theta,\phi)} r^2\sin\theta\,dr\,d\theta\,d\phi. \tag{5.55}$$

For the perturbed sphere, $R(\theta,\phi) = r_0 + \alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]$, so:

$$V = \int_0^{2\pi}\int_0^{\pi}\int_0^{r_0+\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]} r^2\sin\theta\,dr\,d\theta\,d\phi. \tag{5.56}$$

The radial integral is:

$$\int_0^{r_0+\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]} r^2\,dr = \frac{1}{3}\left(r_0+\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^3. \tag{5.57}$$

Expanding this using the binomial theorem:

$$\left(r_0+\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^3 = r_0^3+3r_0^2\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]+3r_0\left(\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^2+\left(\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^3. \tag{5.58}$$

We note that odd powers of spherical harmonics on $d\theta$ are zero for $\ell > 0$, and hence we only care about the even powers. The radial integral becomes:

$$\int_0^{r_0+\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]} r^2\,dr \approx \frac{1}{3}r_0^3 + r_0\left(\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^2. \tag{5.59}$$

and the volume:

$$V = \int_0^{2\pi}\int_0^{\pi}\left[\frac{1}{3}r_0^3 + r_0\left(\alpha\cdot\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^2\right]\sin\theta\,d\theta\,d\phi. \tag{5.60}$$

Split this into two terms:

$$V = \frac{1}{3}r_0^3\int_0^{2\pi}\int_0^{\pi}\sin\theta\,d\theta\,d\phi \tag{5.61}$$

$$+\frac{r_0\alpha^2}{3}\int_0^{2\pi}\int_0^{\pi}\left(\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^2\sin\theta\,d\theta\,d\phi. \tag{5.62}$$

The first term evaluates to the volume of the unperturbed sphere:

$$\int_0^{2\pi}\int_0^{\pi}\sin\theta\,d\theta\,d\phi = 4\pi, \quad \text{so } \frac{1}{3}r_0^3\cdot 4\pi = \frac{4\pi}{3}r_0^3. \tag{5.63}$$

Here, for instance for $m = 4$, $\ell = 6$,

$$\mathrm{Re}[Y_6^4(\theta,\phi)] = \frac{3}{32}\sqrt{\frac{91}{2\pi}}\cos(4\phi)\sin^4(\theta)\left(11\cos^2(\theta)-1\right). \tag{5.64}$$

Integrating, we find

$$\int_0^{2\pi}\int_0^{\pi}\left(\mathrm{Re}[Y_\ell^m(\theta,\phi)]\right)^2\sin\theta\,d\theta\,d\phi = \frac{1}{2}+\frac{\delta_{m0}}{2}, \tag{5.65}$$

Thus, the volume becomes (for $m \neq 0$):

$$V \approx \frac{4\pi}{3}r_0^3 + \frac{r_0\alpha^2}{2} \tag{5.66}$$

To preserve the volume, the additional term must cancel out. Therefore:

$$\frac{4\pi}{3}\tilde{r}_0^3 = \frac{4\pi}{3}r_0^3 + \frac{r_0\alpha^2}{2}, \tag{5.67}$$

Divide by $\frac{4\pi}{3}$:

$$\tilde{r}_0^3 = r_0 \left(r_0^2 + \frac{3\alpha^2}{8\pi} \right). \tag{5.68}$$

Therefore, we set the correction factor $R_c = R(\theta, \phi, \alpha)\eta(\alpha)$ to be

$$\eta(\alpha) = \left(1 + \frac{3\alpha^2}{8\pi r_0^2} \right)^{-\frac{1}{3}}. \tag{5.69}$$

In Fig. 5.5, we plot the change in measurement uncertainty of both clustered and uniform receptor distributions as we perturb the spherical cell, and compare it with the non-adjusted for constant volume case shown from the main text.

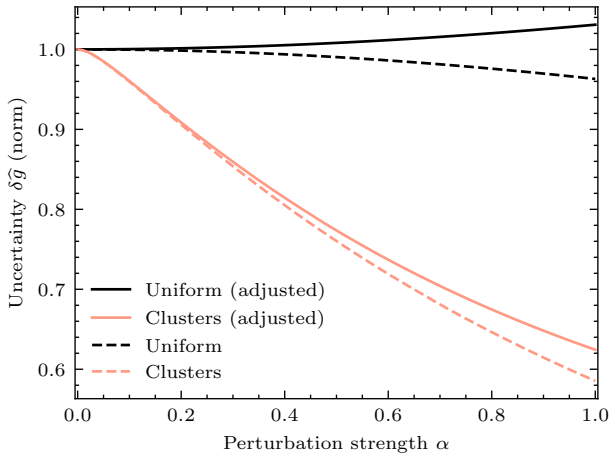


Figure 5.5: Comparison of the improvement in gradient estimation between the adjusted radius formulation (solid) and the original simpler formulation without volume conservation (dashed). The system is composed of $N = 720$ sensors for both cases and perturbation is Y_6^4.

The effect remains qualitatively the same.

A6 Effect of correlation length on cluster formulation

The correlation length, controlled here by the effective receptor size a, plays an role in determining the spatial structure of receptor clusters. Fig. 5.6 illustrates how estimation error increases with respect to the minimal separation h between receptors in a cluster. This separation is influenced by the effective sensor radius, which introduces a scaling relationship $h \sim \log a$.

A7 Optimal distribution in more complex shapes

Here, we consider the case of real cells, which are arguably less symmetric than the simple perturbed shapes we study in the main text. Notably, real cells morphologies,

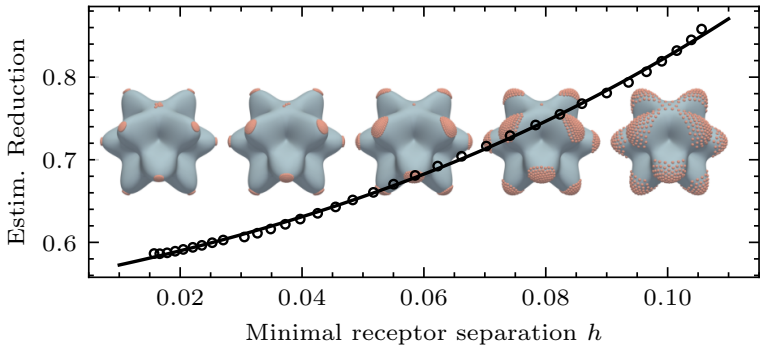
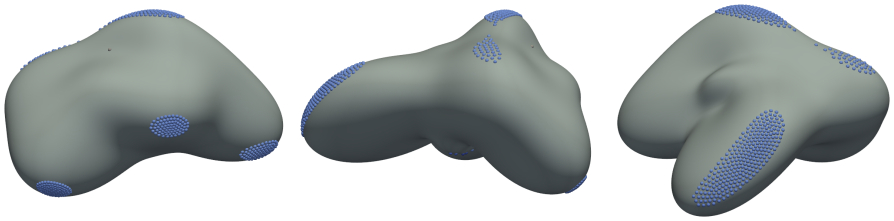


Figure 5.6: Reduction in estimation error given the minimal separation h between receptors in a cluster, modulated by the effective sensor radius which scales the separations as $h \sim \log(a)$.

such as in the case of T cells, can be represented as a linear combinations of spherical harmonics [285], so we consider the case of the cell defined by

$$\boldsymbol{r}(\theta, \phi) = \sum_{\ell=0}^{7} \sum_{m=-\ell}^{\ell} \boldsymbol{\alpha}_{\ell,m} \mathrm{Re}[Y_\ell^m(\theta, \phi)] \tag{5.70}$$

where we have use the coefficients from a cell sample from Cavanagh et al. [285].



(a) (b) (c)

Figure 5.7: Results of receptors distribution on a less symmetric and rugged shape. The images show that even when symmetry is not persevered, clustering still mostly occurs at high negative mean curvature.

The optimal distribution agrees with results in the main text, where receptors cluster on protrusions to reduce estimation uncertainty.

A8 Clustering intuition in 1D

In this section, we discuss the intuition of receptor clustering. Simply, receptors need to cover the most distance possible (in all directions), and avoid being too close to avoid rebinding events. The mathematics simplify significantly in 1D:

Assume a very simple case where we can place two receptors, (x_0, x) where x_0 is fixed and $x \in (0, 1]$, and we would like to estimate the sloped of a linear function $y(x) = mx + n$. We assume that the sampling yields two measurements $y_i = y(x) + \eta$,

with $\eta \sim \mathcal{N}(0, \Sigma)$. For simplicity, we will assume the measurements are uncorrelated and have the same noise and thus $\Sigma_{i,j} = \sigma$.

The Fisher Information Matrix for such a system is given by

$$\mathcal{I}(m,n) = \frac{1}{\sigma^2} \begin{pmatrix} x_0^2 + x^2 & x_0 + x \\ x_0 + x & 2 \end{pmatrix}. \tag{5.71}$$

from which we can compute the uncertainty in estimation in m with the Cramer-Rao bound.

$$\sigma_{\hat{m}}^2 = \left(\mathcal{I}^{-1}(m,n)\right)_{1,1}. \tag{5.72}$$

We compute $\mathcal{I}^{-1}(m,n)$:

$$\mathcal{I}^{-1}(m,n) = \frac{\sigma^2}{\det(\mathcal{I})} \begin{pmatrix} 2 & -(x_0 + x) \\ -(x_0 + x) & x_0^2 + x^2 \end{pmatrix}, \tag{5.73}$$

where the determinant becomes

$$\det(\mathcal{I}) = \left(x_0^2 + x^2\right)(2) - (x_0 + x)^2 = 2(x_0^2 + x^2) - (x_0 + x)^2. \tag{5.74}$$

Thus, the variance is

$$\sigma_{\hat{m}}^2 = \frac{2\sigma^2}{2(x_0^2 + x^2) - (x_0 + x)^2}. \tag{5.75}$$

Assuming $x_0 = 0$, we have $\sigma_{\hat{m}} = 2\sigma^2/x^2$, which is minimal when x is maximum at $x = 1$ (Fig. 5.8). In practical terms, this means that placing the receptors farther apart increases the Fisher Information and reduces the uncertainty in the slope estimate.

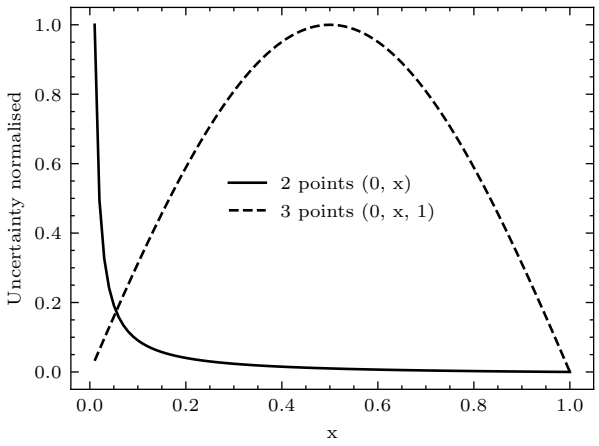


Figure 5.8: Normalized uncertainty of the estimation depending on the position of the free sampling location.

Interestingly, when we introduce a new sample, i.e. using three points (x_0, x, x_1), and repeat the steps above, we end up with

$$\sigma^2_{\hat{m}} = \frac{3\sigma^2}{3\left(x_0^2 + x^2 + x_1^2\right) - (x_0 + x + x_1)^2}, \tag{5.76}$$

Assuming $x_0 = 0$ and $x_1 = 1$, we find

$$\sigma^2_{\hat{m}} = \frac{3\sigma^2}{3\left(x^2 + 1\right) - (x + 1)^2} = \frac{3\sigma^2}{2x^2 - 2x + 2}. \tag{5.77}$$

Since $x \in (0, 1)$, we can clearly see that $\sigma_{\hat{m}}$ is minimized when the denominator is maximized, which happens at $x \to 0$ and $x \to 1$ when constrained to $x \in (0, 1)$. This indicates that while the sampled points aim to cover the entire domain and maximize their spread, the fixed boundaries cause the optimal positions to be near existing sampled points. This proximity persists if there are samples from the opposite end of the domain. Once we introduce correlation between samples, spreading becomes less optimal and thus, sampling would be more effective slightly closer to the center instead of overlapping on the edges.

Chapter 6

On adaptive transport networks

The following chapter contains the paper:

Adaptive Node Positioning in Biological Transport Networks

Albert Alonso[†,1], *Lars Erik J. Skjegstad*[†,1] *and Julius B. Kirkegaard*[1,2]

1. Niels Bohr Institute, University of Copenhagen, 2100, Copenhagen, Denmark.
2. Department of Computer Science, University of Copenhagen, 2100, Copenhagen, Denmark.
† Shared first author.

Under Review in: *Physical Review Letters* (PRL)
Pre-print server: https://arxiv.org/abs/2411.00692

Author Contributions:
Conceptualisation, Methodology, Validation, Investigation, Data Management, Visualisation, Formal Analysis, Software, Writing Original Draft.

Letter reformatted to fit the style of this thesis.

Abstract

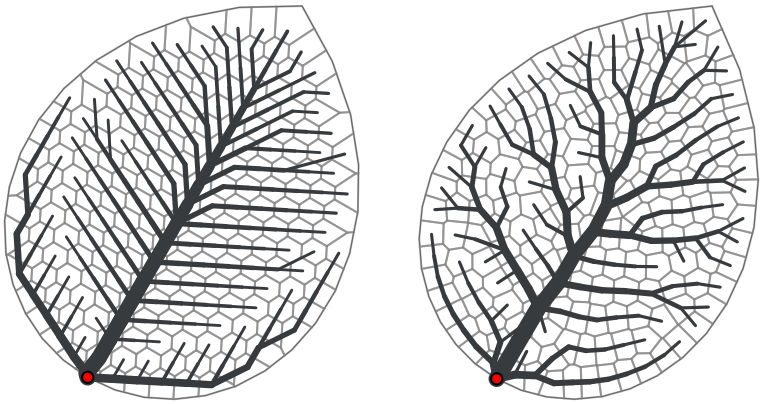
Biological transport networks are highly optimized structures that ensure power-efficient distribution of fluids across various domains, including animal vasculature and plant venation. Theoretically, these networks can be described as space-embedded graphs, and rich structures that align well with observations emerge from optimizing their hydrodynamic energy dissipation. Studies on these models typically use regular grids and focus solely on edge width optimization. Here, we present a generalization of the hydrodynamic graph model which permits additional optimization of node positioning. We achieve this by defining sink regions, accounting for the energy dissipation of delivery within these areas, and optimizing by means of differentiable physics. In the context of leaf venation patterns, our method results in organic networks that adapt to irregularities of boundaries and node misalignment, as well as overall improved efficiency. We study the dependency of the emergent network structures on the capillary delivery conductivity and identify a phase transition in which the network collapses below a critical threshold. Our findings provide insights into the early formation of biological systems and the efficient construction of transport networks.

6.1 Introduction

Transport networks are ubiquitous in nature and in living systems. The efficiency of such networks is crucial for the evolutionary fitness of organisms such as those observed in leaf venation [137, 147, 300] and blood vasculature systems [136, 143, 300], and emerges in complex systems such as river networks [301] and human transport systems [141, 142, 302]. Understanding the structure and morphogenesis of such network structures has been facilitated by studying energetically optimal solutions to static, hydrodynamic networks [145, 146, 148, 149].

The optimization of hydrodynamic transport networks is traditionally approached as an edge-optimization problem [136, 145–147], assuming systems where network nodes serve as sinks for their local area. Thus, optimizing the energy dissipation leads to optimal edge conductivities [145], while the positions of the nodes themselves are considered fixed. In principle, this approach can be used to model any bounded system if enough nodes are used. However, for a finite number of nodes confined within a bounded system, it is evident that the node placement itself influences the optimality of the fluid delivery system. In many systems, the finite number of nodes is a physical fact and must be imposed, e.g., due to a lower bound on the vein thicknesses determined by the capillary size.

Here, we consider bounded systems of finite nodes and generalize the hydrodynamic model to have well-defined optima both in edge conductivities and node positioning. Specifically, we model a bounded leaf venation network and let the boundary represent the leaf margin. Each node represents a source or sink (we set a single node at the leaf base as the source, and let all other nodes be sinks) and each edge represents a vein between two nodes. Fig. 6.1a shows the optimal solution obtained for a fixed, hexagonal



(a) (b)

Figure 6.1: Optimization and relaxation on an inclined leaf ($\varphi_0 = 60°$, $\beta = 0.4$, $N = 175$). Non-suppressed edges and delivery regions of each sink node are shown. The source is marked with a red dot. (a) Result after transport network optimization solving Eq. (6.5) on a hexagonal grid. (b) Result after relaxation, where the network optimizes $P = P_t + P_d$.

grid in a bounded system. The imposition of the domain boundary and the inability of the nodes to adapt positionally results in non-uniform areas associated with each node. As fluid dissipation will typically be proportional to area, this thus implies that the sinks become non-uniform. By removing the spatial constraint on the node positions, we find more energy-efficient solutions, which yield networks that appear organic and more consistent with networks observed in nature (Fig. 6.1b).

6.2 Methods

The power of the transport network is given by [145]

$$P_t = \sum_{e \in \text{edges}} \left(\frac{F_e^2}{C_e} + c_t \, C_e^\gamma \right) L_e, \tag{6.1}$$

where F_e is the flux associated with edge e, which has length L_e and conductivity C_e. The constant c_t defines the metabolic cost of maintaining the edge, which is equivalent to considering a system with a finite amount of resources. γ is a parameter that determines how the material cost scales with the conductivities, which we set to $\gamma = 1/2$ in this Letter. The flow is pressure-driven, such that the flux over a (directed) edge $e = i \to j$ is given by $F_e = (p_i - p_j)C_e/L_e$. The pressures are indirectly determined by the need to satisfy Kirchhoff's law

$$\sum_{e \in i} \pm_e F_e = s_i, \tag{6.2}$$

where s_i is the source/sink at node i and the sign is set by the direction of the edge. This can be solved efficiently with hardware acceleration [139]. Typically, the sink magnitudes

are assumed to be equal. However, once the nodes are permitted to move, these values must change as well. Thus, taking the source $s_0 = 1$, we define

$$s_i = -\frac{A_i}{\sum A_i} \quad \text{for} \quad i \geq 1, \tag{6.3}$$

where A_i is the area surrounding a node, implicitly defined by the associated Voronoi cell.

In order for a transport system to have stable and well-defined optima in node positioning, the above model must be expanded to account for the power dissipation within the Voronoi cells. To achieve this, we add a simple power-delivery term, P_d, to the conventional formulation for transport power [145], and thus assume that the total power can be described by the sum of two contributions $P = P_t + P_d$. With the goal of defining a self-consistent model that equally considers both transport and delivery costs, we take

$$P_d = \sum_{i \in \text{sinks}} \frac{s_i^2}{\hat{C}_i} \langle \ell \rangle_i. \tag{6.4}$$

This is analogous to the first term in the transport formulation (Eq. (6.1)), but considers the power dissipation due to the delivery of the sink fluid s_i over an average Voronoi distance $\langle \ell \rangle_i$. $\hat{C}_i$ is the delivery conductivity, which effectively models a capillary system. Physically, this term favors equally sized and isotropic sink areas. We assume that the delivery conductivity is an intrinsic property of the material, such that $\hat{C}_i = \hat{C}$ is fixed and equal for all nodes.

6.3 Results

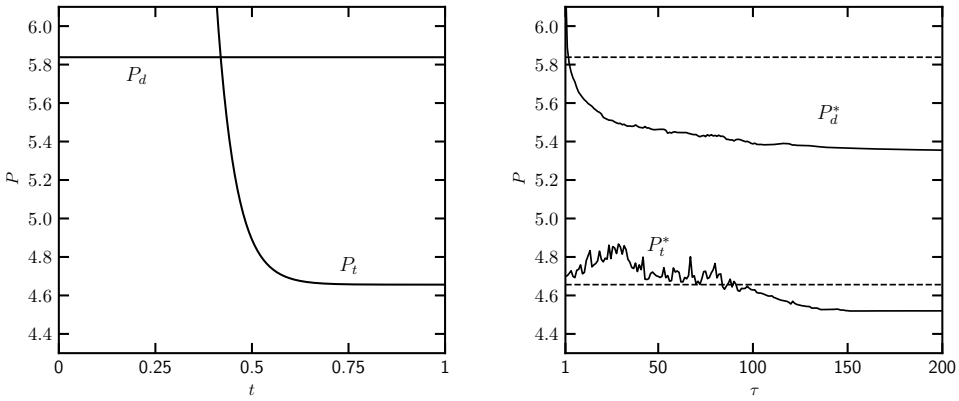
6.3.1 Optimization

Our scheme to minimize P consists of the combination of an edge conductivity optimization and a node relaxation process. We employ a fully connected graph and initialize all edge conductivities to the same value. For a given node positioning $\{\boldsymbol{x}_i\}$, we optimize the conductivities by using an adaptation model [146, 147]

$$\frac{\mathrm{d}C_e}{\mathrm{d}t} = \left(\frac{F_e^2}{C_e^{\gamma+1}} - \gamma c_t \right) C_e + c_0 e^{-\lambda t}, \tag{6.5}$$

where the term $F_e^2 / C_e^{\gamma+1}$ is the squared wall shear stress, and γc_t represents the optimal squared shear stress. Furthermore, we include a growth term characterized by the area growth rate λ, which has proven to be a robust strategy to achieve better optima [147]. To facilitate smooth convergence, we also decrease exponentially the magnitude of the growth term $c_0(\tau) \propto e^{-\nu\tau}$ during the node positioning *relaxation process*. Here, we use τ to denote the timescale of node relaxation and t for the conductivity optimization.

Crucially, even though our transport network is fully connected, optimization results in a sparse, planar graph (Fig. 6.1). While networks with static nodes can simplify



(a) (b)

Figure 6.2: Transport and delivery power terms (Eqs. (6.1) and (6.4)) during optimization. (a) Optimization of P_t by using Eq (6.5). (b) Node relaxation process. At each timestep one complete optimization of P_t is performed, resulting in an improvement of both power terms.

the incidence matrix by, for instance, employing Delaunay triangulation to account for sparse connections, we rely on the derivative information of the entire graph during the optimization to accurately determine the correct gradients for node displacement and to allow for the spontaneous creation and suppression of connections.

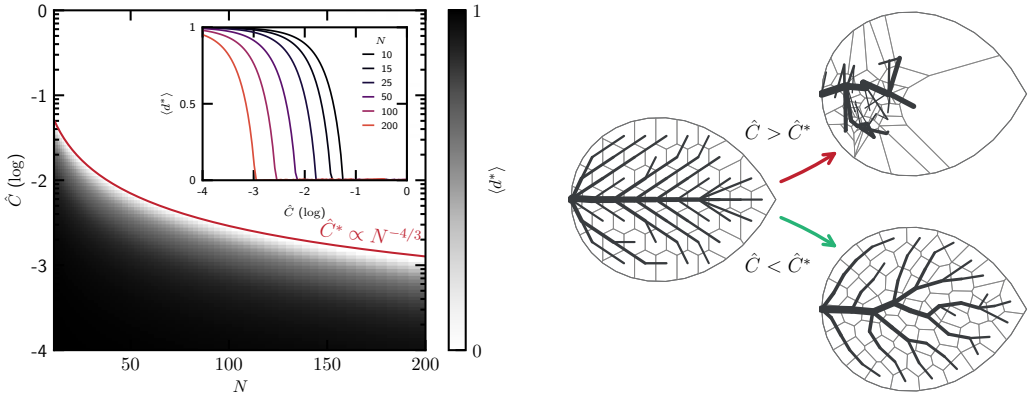
To optimize over $\{\boldsymbol{x}_i\}$, we use gradient descent computed by automatic differentiation techniques [20, 37]. To propagate the gradients across the solution of the adaptation model in Eq. (6.5), we employ a custom backpropagation method based on the implicit function theorem [21], detailed in the Appendix. Importantly, the fact that we use Voronoi cells to define the sink magnitudes s_i and the delivery distances $\langle \ell \rangle_i$ means that all terms vary continuously with node positions, yet it requires the ability to differentiate through Voronoi calculations. We achieve this with a custom differentiable implementation of Voronoi tessellation [34, 303, 304], clipped by the domain boundary [1] (see Appendix for details). Crucially, this procedure is parallelizable and amenable to hardware acceleration. As Voronoi cells are guaranteed to be convex, we can split the integrals over these cells into triangles T, e.g.,

$$\langle \ell \rangle_i = \frac{1}{A_i} \sum_{T \in \mathrm{vor}_i} \iint_T \|\boldsymbol{x}_i - \boldsymbol{x}\| \, \mathrm{d}A, \tag{6.6}$$

over which the integrals can be evaluated analytically (see Appendix). We note that our optimization schemes are local optimizers and will generally not find global optima.

During the optimization of conductivities, delivery cost remains fixed (Fig. 6.2a). On the other hand, during node placement relaxation, we see that while delivery costs are monotonically decreasing, transport costs experience a phase of stochasticity where the

[1] Code implementation is available at https://github.com/kirkegaardlab/gradnodes



(a) (b)

Figure 6.3: Network collapse as a function of effective delivery conductivity. (a) Phase diagram of (normalized) average nearest-neighbor distance for different $\hat{C}$ and network sizes in the one-dimensional system. The line shows the scaling of $\hat{C}$ at constant Ω, (analytically obtained from the scaling analysis). (inset) Individual network sizes from (a) showing the sudden but continuous collapse. (c) Resulting networks with $\hat{C}$ above and below the critical value in two dimensions.

efficiency is reduced (Fig. 6.2b). Nevertheless, the total power continuously decreases, and after some iterations both the transport power P_t and the delivery power P_d reach lower values than those achievable in the regular grid, showing that the terms are not opposing criteria and that the simultaneous optimization yields networks that show increased power efficiency in both.

6.3.2 Network stability

The delivery term P_d depends on the capillary conductivity $\hat{C}$. To understand its influence on the network dynamics, we consider a one-dimensional system, where the nodes are solely connected to their adjacent nodes. This removes the need for Voronoi calculations, where instead the average delivery distance is

$$\langle \ell \rangle_i = \frac{1}{2(x_R - x_L)} \left[(x_R - x_i)^2 + (x_i - x_L)^2 \right], \tag{6.7}$$

where x_L and x_R are the midpoints between the left and right neighbor, respectively (see Appendix). Fig. 6.3a show the mean node separation $\langle d \rangle$ in this 1D system confined to a fixed domain for increasing values of $\hat{C}$. We observe a smooth phase transition: When $\hat{C}$ is small, the energy is largely dominated by the delivery energy, which is minimal when the nodes are uniformly spread. As we increase the delivery conductivity of the system, we observe a continuous change in the network until the network collapses (Fig. 6.3a).

To understand the origin of this transition, we consider the contributions of each power term. If the energy needed to transfer fluid from a sink node to the surrounding region is significantly greater than the energy required for transportation (small $\hat{C}$),

the cost incurred by reducing the connections between nodes – and thereby minimizing transport energy – is too high to justify compromising the size discrepancies of the regions. In contrast, when the delivery conductivity is sufficiently high to allow effectively energy-free fluid transfer from the sink to the region, the optimal solution is achieved by reducing the transport cost, i.e., moving all the nodes to the source, as a single node can then deliver fluid to the entire domain cheaply. This leads us to study the transition using the dimensionless power ratio

$$\Omega = \frac{P_t}{P_d}. \tag{6.8}$$

Scaling analysis of the one-dimensional system reveals that $\mathcal{O}(\Omega) = N^{4/3}$ (see Appendix). Since $\Omega \propto \hat{C}$, we observe that the abrupt collapse of the system in Fig. 6.3a aligns with the outcomes of the scaling analysis. This analysis can be extended to the two-dimensional formulation, resulting in $\mathcal{O}(\Omega) = N^{5/3}$, which predicts that *valid* relaxed networks only emerge below the critical $\hat{C}^*$, as shown in Fig. 6.3b. Although more complex in two dimensions, the phase transition still occurs.

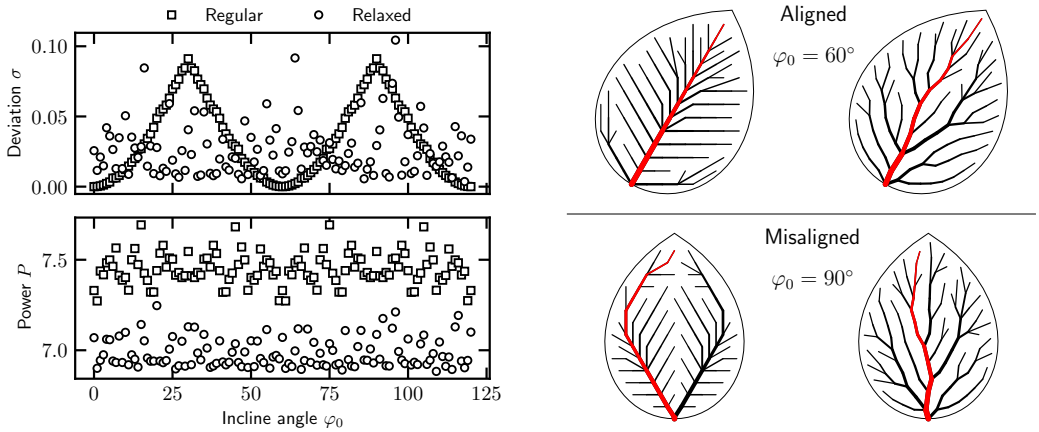
We note that the above scaling laws are derived for a fixed domain space. If instead, we consider a 2D domain that increases with network size, e.g., $\mathcal{O}(L) = \sqrt{N}$ (i.e., a growing leaf), we find the resulting scaling to be $\mathcal{O}(\Omega) = N^{-1/3}$. Thus, for fixed domains, the more densely packed the network is, the lower the capillary conductivity must be in order for the formulation to hold. If, however, the domain expands, the system can maintain the same effective conductivity during growth – as expected of an intrinsic property. For systems with a scaling domain, delivery costs become negligible at large network sizes, where instead the transport costs dominate completely.

6.3.3 Adaptability

One constraint of regular lattice graphs is that their optimal solutions are dependent on the alignment of the main axis of the leaf with the grid. To find optimal networks for fixed grids, careful construction and alignment of the network boundary is therefore required (Fig. 6.1a). Typically, this alignment is chosen so that the main axis follows the shortest path between neighboring nodes, resulting in network shapes where the leaf main branch is perfectly parallel with the main axis (Fig. 6.1a). Another issue is the aforementioned non-uniformity of the sink areas along the leaf boundary. Thus, for fixed grids, careful construction and alignment of the network boundary is required. To illustrate this, we construct a leaf shape that can be rotated relative to the grid by using a simplified version of Gielis' superellipse equation [305], as it describes a broad range of leaf shapes accurately [306]:

$$r(\varphi; \varphi_0, \beta) = \left(\left| \cos\left(\tfrac{\varphi-\varphi_0}{4}\right) \right| + \left| \sin\left(\tfrac{\varphi-\varphi_0}{4}\right) \right| \right)^{-1/\beta}. \tag{6.9}$$

Here φ_0 is the inclination of the major axis of the leaf, and β is a shape parameter.



(a) (b)

Figure 6.4: Model adaptability on differently aligned leaves. (a) The deviation σ (top), as well as power P (bottom) are 60°-periodic for the regular grid, and non-periodic for the relaxation model. (b) Resulting leaves for maximum alignment (top) and misalignment (bottom) are fundamentally different for the regular grid (left), but similar for the relaxation model (right).

We quantify the discrepancies by rotating the leaf and measuring the power P and the average deviation of the main branch with the grid

$$\sigma = 1 - \frac{\langle c_i \cos(\theta_i - \varphi_0)\rangle}{\langle c_i \rangle}, \tag{6.10}$$

where θ_i and c_i are the angle and conductivity of the i'th edge of the main branch, respectively. The main branch is found by greedily following the edges with the highest conductivity, from the source to a leaf node. On the hexagonal grid, angles $\varphi_0 \equiv 0 \,(\mathrm{mod}\, 60°)$ correspond to maximum alignment of the leaf with the grid, whereas $\varphi_0 \equiv 30° \,(\mathrm{mod}\, 60°)$ correspond to maximum misalignment.

Fig. 6.4 shows the average main branch deviations for the outputs of the regular grid and the relaxation model. For the regular grid, we see that the deviation is completely dependent on angle, with a minimum of $\sigma = 0$ at the maximum values of leaf alignment, as well as peaks at maximum misalignment. The slightly non-smooth changes in deviation are caused by the discrete differences in the initializations of the node positions due to boundary clipping. For the deviations in the relaxation model, there is no apparent dependency on angle, which indicates that the model's independence of initial boundary alignment, with the noisiness being the result of local optimization. While zero deviation is never reached for the relaxation model (due to the stochastic nature of the optimization), the average deviation $\langle\sigma\rangle = 0.026$ is nonetheless lower than $\langle\sigma\rangle = 0.033$ for the regular grid.

The resulting power from the same data can be seen in Fig. 6.4a. The mean power is significantly lower for the relaxation model. We note that at maximum alignment we still see a lower power in the relaxation model, even though the regular grid output has lower

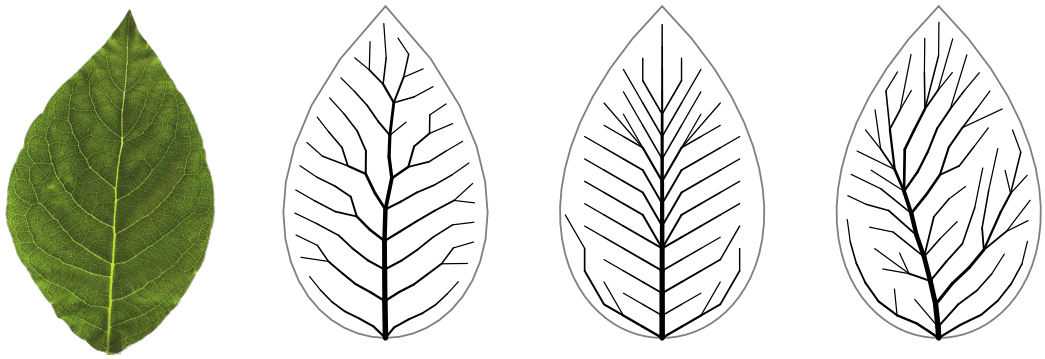
deviation, which can be attributed to the fact that power is not solely determined by the structure of the main branch, but by the network in its entirety. Figs. 6.4b illustrate the output of the models for maximum alignment and misalignment, respectively. The regular-grid model yields two extremes where the main branch either completely follows the main axis (Fig. 6.4b, top-left), or splits into two similarly sized branches (Fig. 6.4b, bottom-left). In contrast, all outputs from the relaxation model preserve the central main branch (Figs. 6.4b, right).

6.3.4 Vein curvature

A characteristic feature of the model networks is the emergence of smoothly curving veins, leading to venation patterns that appear more organic compared to those produced by a regular grid. Previous work relied on complementary initialization techniques in order to effectively mimic biological stochasticity, for example by using disordered tessellation grids [147]. Such a method works by enforcing a repulsive potential between sink nodes that leads to evenly spaced positions in the domain. This is approximately equivalent to independently optimizing P_d in our formulation.

In Fig. 6.5 we compare the resulting morphologies of a leaf that mimics the domain of *L. xylosteum* (Fig. 6.5a). This may be compared to venation patterns in our formulation (Fig. 6.5b), and a regular grid (Fig. 6.5c), respectively. In Fig. 6.5d we show the result of sequential optimization, i.e., optimization of P_d and subsequent optimization of P_t. We observe sub-optimal solutions where the venation patterns do not match those of the real leaf. Instead, we find that the optimization of P_t identifies approximately straight lines in the optimal node positions, and uses these to form the main branch.

Our observations indicate that incorporating coupled energy costs in the power formulation leads to solutions influenced by domain boundaries, closely resembling those



(a) (b) (c) (d)

Figure 6.5: Comparison between a real leaf and model venation patterns ($\beta = 0.25, N = 100$). (a) Sample of *L. xylosteum* [139]. (b-d) Model output from the (b) relaxation model, (c) regular grid, and (d) sequential optimization.

found *in vivo* (Fig. 6.5b). This is reproduced to a lesser extent when performing sequential optimization (Fig. 6.5d), and is completely absent when relying on regular grids (Fig. 6.5c).

6.4 Discussion

In this Letter, we have addressed the limitations of traditional transport network optimization models by allowing the optimization over the node positions themselves. We have shown this to be a well-defined problem when incorporating the cost of resource delivery into the energy formulation, with resulting configurations that show adaptability to domain boundaries and misalignment. The study has been enabled by exploiting a fully differentiable process including the Voronoi tessellation of the domain and the steady state of the conductivity adaptation model. Our results demonstrate the advantages of node localization both internally and in adapting to external domain boundaries and allow the emergence of natural organic networks. This approach could find applications in the efficient design of human-engineered networks [72]. We identified a phase transition as a function of the capillary conductivity $\hat{C}$, showing network collapse above a critical value, reminiscent of the well-known phase transition in γ for static networks [145]. Scaling analysis further reveals that physical networks remain stable during network growth only if the domain expands along with the network. While our goal has been a minimal extension of the hydrodynamic network model, we note that a main insight is the addition of *some* delivery power term and not necessarily precisely the one of Eq. (6.4). For instance, similar phase transition behavior emerges from considering, e.g., $\langle \ell^2 \rangle$ (diffusion-limited costs). Our approach is limited by the local behavior of gradient descent and the presence of many local optima. Resulting patterns are thus sensitive to initial conditions even though the energy dissipation is similar between them. This problem similarly prevails in edge optimization alone [145]. Our findings pose an interesting question for further research: how can appropriate *local* feedback models [146, 147] be formulated that optimize $P_t + P_d$? Finally, we note that in this Letter we do not consider fluctuations in the sink magnitudes or similar phenomena that can result in reticulate networks [148, 149, 307], the relevance of which becomes evident for larger network sizes. This can be incorporated by choosing a suitable parameterization that allows for area-weighted sinks [139].

Appendix

A1 Spatially continuous sinks: integrals

In this section, we provide a detailed analysis of the area computation of the region where the sink nodes deliver to and explore the integration of distance metrics in a triangle within the polygon. Take a *star convex* polygon (such as those that result from Voronoi tesselations), and consider a triangle ABC in that polygon:

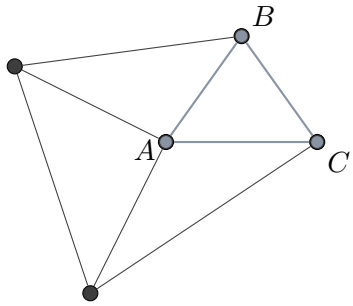


Figure 6.6: Diagram of a triangle of a Voronoi cell. A is the sink node and B and C are vertices resulting from the tessaration.

The area of such a triangle is trivial to calculate, and thus the total area is:

$$A = \sum_{T \in \text{triangles}} A_T, \tag{6.11}$$

where A_T is trivially computed with

$$A_T = \frac{1}{2} \left| (\boldsymbol{x}_A - \boldsymbol{x}_B) \times (\boldsymbol{x}_A - \boldsymbol{x}_C) \right|. \tag{6.12}$$

Likewise, we need the average distance from A to all points in the polygon. This becomes:

$$\langle L \rangle = \frac{1}{A} \sum_{T \in \text{triangles}} \int_{A_T} ||\boldsymbol{x}_A - \boldsymbol{x}|| \, \mathrm{d}A. \tag{6.13}$$

This integral is not easy, but can be done by using polar coordinates. We write

$$\boldsymbol{x}(r, \theta) = \boldsymbol{x}_A + r \begin{pmatrix} \cos\theta \\ \sin\theta, \end{pmatrix} \tag{6.14}$$

and the integral becomes

$$\mathcal{I} = \int_{A_T} ||\boldsymbol{x}_A - \boldsymbol{x}|| \, \mathrm{d}A = \int_{\theta_1}^{\theta_2} \mathrm{d}\theta \int_0^{R(\theta)} \mathrm{d}r \, r^2 = \frac{1}{3} \int_{\theta_1}^{\theta_2} R(\theta)^3 \, \mathrm{d}\theta, \tag{6.15}$$

where $R(\theta)$ is the distance from A to the line BC with angle θ,

$$R(\theta) = \frac{x_a(y_b - y_c) + x_b(y_c - y_a) + x_c(y_a - y_b)}{(y_c - y_b)\cos\theta + (x_b - x_c)\sin\theta}, \tag{6.16}$$

thus, our integral becomes

$$\mathcal{I} = \frac{1}{3}\left[x_a(y_b - y_c) + x_b(y_c - y_a) + x_c(y_a - y_b)\right]^3 \int_{\theta_1}^{\theta_2} \frac{1}{(\delta y \cos\theta - \delta x \sin\theta)^3}\,\mathrm{d}\theta,$$

where $\delta x = x_c - x_b$ and $\delta y = y_c - y_b$. This integral can be evaluated using Weierstrass substitution to give

$$\begin{aligned}\int \frac{1}{(\delta y \cos\theta - \delta x \sin\theta)^3}\,\mathrm{d}\theta =& \frac{1}{\delta^3}\tanh^{-1}\left(\frac{\delta x + \delta_y \tan(\theta/2)}{\delta}\right) \\ &+ \frac{\delta x \cos\theta + \delta y \sin\theta}{2\delta^2(\delta y \cos\theta - \delta x \sin\theta)^2},\end{aligned} \tag{6.17}$$

where we defined $\delta = \sqrt{\delta x^2 + \delta y^2}$. Note that

$$\tan(\theta/2) = \frac{\sin\theta}{1 + \cos\theta}, \tag{6.18}$$

so this can all be written in terms of $\cos\theta$ and $\sin\theta$ (i.e. no need to actually calculate θ). Thus, the formula is complete by specifying

$$\cos\theta_1 = \frac{x_B - x_A}{||\boldsymbol{x}_B - \boldsymbol{x}_A||}, \qquad \sin\theta_1 = \frac{y_B - y_A}{||\boldsymbol{x}_B - \boldsymbol{x}_A||}, \tag{6.19}$$

$$\cos\theta_2 = \frac{x_C - x_A}{||\boldsymbol{x}_C - \boldsymbol{x}_A||}, \qquad \sin\theta_2 = \frac{y_C - y_A}{||\boldsymbol{x}_C - \boldsymbol{x}_A||}. \tag{6.20}$$

As Weierstrass substitution has problems at $\theta = \pi$, the integral is only valid when the triangle is aligned with node A west of nodes B, C — but this is always achievable by a simple rotation.

A2 Solution for the 1-Dimensional case

Here, we show the derivations and explanations to find the optimal distribution of $\boldsymbol{x}$ in the simplified case of a one-dimensional (1D) leaf. We begin by considering a system with a single source $s_0 = 1$ at $x_0 = 0$, and n delivery nodes positioned at locations $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\}$, which get constrained to $x \in [0, L]$.

Similar to the 2D case, the sink values s_i represent the rate at which flow is absorbed or removed at each node x_i. The sink value is calculated based on the distances between the neighbouring nodes, normalized by the system length L. An exception to this is the first and last nodes, where the distance to the boundary is used instead.

$$s_i = -\frac{1}{2L}(x_{i+1} - x_{i-1}), \tag{6.21}$$

with special attention to correct at x_0 and x_n.

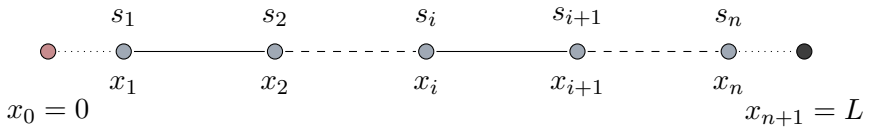


Figure 6.7: Diagram of the 1D transport networks with moving nodes (blue) and a fixed source (red).

Next, we compute the flow values F_i of the passing flow through each node. Since we are in 1D, we assume that the nodes are only connected to the nodes on their sides. Hence, the flow expression for each edge simplifies to

$$F_{i+1} = F_i - s_i = F_1 - \sum_{j \leq i} s_j, \tag{6.22}$$

given that $F_1 = 1$.

The conductivity at each node denoted C_i, is a function of the flow F_i. We can find the optimal conductivity for this system by minimizing the transport power P_t given by

$$P_t = \sum_{i \in \text{transport}} L_i \left(\frac{F_i^2}{C_i} + c_t C_i^\gamma \right), \tag{6.23}$$

by setting $\frac{\partial P}{\partial C} = 0$, we find the conductivity at node x_i to be:

$$C_i = \left(\frac{F_i^2}{c_t \gamma} \right)^{\frac{1}{1+\gamma}}. \tag{6.24}$$

This formula captures the non-linear dependence of conductivity on the square of the flow, modulated by the parameters c_t and γ.

The total power P of the system consists of two components: the transport power P_t and the delivery power P_d. The transport power is given by:

$$P_\text{t} = \sum_{i \in \text{transport}} L_i \left(\frac{F_i^2}{C_i} + c_t C_i^\gamma \right), \tag{6.25}$$

where L_i represents the length of the transport segment corresponding to node x_i, and the two terms inside the summation represent the power due to flow and the power related to the conductivity, respectively.

The delivery power, on the other hand, is the power used to deliver flow from the sink nodes to their surrounding areas, and it depends on the sink values s_i and the average delivery distance $\langle \ell_i \rangle$. The delivery power is expressed as:

$$P_{\text{delivery}} = \sum_{i \in \text{delivery}} \langle \ell_i \rangle \frac{s_i^2}{\hat{C}}, \tag{6.26}$$

where $\hat{C}$ represents the delivery conductivity, and $\langle \ell_i \rangle$ is the average distance over which the flow must be delivered.

The average delivery distance $\langle \ell_i \rangle$ for a node at position x_i is computed using the positions of the left and right domain edges, denoted x_L and x_R, respectively. The formula for the average delivery distance is the weighted average:

$$\langle \ell_i \rangle = \frac{(x_R - x_i)}{x_R - x_L} \cdot \frac{(x_R - x_i)}{2} + \frac{(xi - x_L)}{(x_R - x_L)} \cdot \frac{(x_i - x_L)}{2} = \frac{1}{2(x_R - x_L)} \left[(x_R - x_i)^2 + (x_i - x_L)^2 \right] \tag{6.27}$$

This formula accounts for the spatial distribution of the flow and the relative positions of the nodes within the system.

A3 Scaling for different network sizes

A3.1 One-dimensional network

We analyze the scaling of the total power P given by

$$P = P_t + P_d, \tag{6.28}$$

with respect to the number of elements N on the 1 dimensional case using the expressions above. For this, we assume the domain to be fixed with N.

Starting by the transport term, we have

$$P_{\mathrm{t}} = \sum_{i \in \mathrm{transport}} L_i \left(\frac{F^2}{C_i} + c_t C_i^{\gamma} \right), \tag{6.29}$$

where $\mathcal{O}(F_i^2) \approx 1/N^2$, and thus $\mathcal{O}(C_i) = (\frac{1}{N^2})^{2/3}$ according to Eq. (6.24). We also assume c_t constant and $\gamma = 0.5$ as in the main text. Taking into account $L_i \propto \frac{L}{N}$, and splitting the terms we find:

$$\mathcal{O}(P_t) = \mathcal{O}(\Sigma)\ \mathcal{O}(L)\ \mathcal{O}(F^2)\ \mathcal{O}(C^{-1}) + \mathcal{O}(\Sigma)\ \mathcal{O}(L)\ \mathcal{O}(C^{1/2}) = N^{-\frac{2}{3}} = N^{-\frac{2}{3}} + N^{-\frac{2}{3}}. \tag{6.30}$$

Hence, the transport network scales as $\mathcal{O}(P_t) = N^{-2/3}$.

In the case of the delivery term, given by

$$P_d = \sum_{i \in \mathrm{delivery}} \langle \ell_i \rangle \frac{s_i^2}{\hat{C}}, \tag{6.31}$$

where $\langle \ell_i \rangle \propto \frac{L}{N}$, $s_i^2 \propto \frac{1}{N^2}$ and $\hat{C}$ is constant, giving:

$$\mathcal{O}(P_d) = \mathcal{O}(\Sigma)\ \mathcal{O}(\langle \ell \rangle)\ \mathcal{O}(s^2) = N^{-2}. \tag{6.32}$$

The ratio between the transport and cost terms to the delivery term scales as:

$$\mathcal{O}(\Omega) = \frac{\mathcal{O}(P_d)}{\mathcal{O}(P_t)} = \frac{N^{-2}}{N^{-\frac{2}{3}}} = N^{-\frac{4}{3}} \tag{6.33}$$

Therefore, the ratio scales as $\mathcal{O}(\Omega) = N^{-4/3}$.

A3.2 Two-dimensional network

In the case of two dimensions, such as the ones on the networks of the main paper, we need to approximate the scaling of the edges. This is due to the optimized conductivities making the graph sparse, resulting in a linear scaling of the edges.

Similarly to A3.1, we want to understand the scaling of the terms of the power by analyizing how its components scale with N.

Here, the scaling of the flow $\mathcal{O}(F^2)$ is unknown because it depends on the network structure and therefore γ. Empirically $\mathcal{O}(F^2) \approx 1/N$ if $\gamma = 1/2$ and $\mathcal{O}(F^2) \approx 1/\sqrt{N}$ if $\gamma = 1/4$. This is approximate because in reality F^2 does not scale uniformly: some edges have $\mathcal{O}(1)$ and some have $\mathcal{O}(1/N)$.

Here we assume $\gamma = 1/2$ and take

$$\mathcal{O}(F^2) = N^{-1} \tag{6.34}$$

The remaining terms have known scaling with N:

$$C = \left(\frac{F^2}{c_t \gamma}\right)^{\frac{1}{1+\gamma}} \Rightarrow \mathcal{O}(C) = \mathcal{O}(c_t)^{-\frac{2}{3}} N^{-\frac{2}{3}} \tag{6.35}$$

$$\mathcal{O}(L) = N^{-\frac{1}{2}} \tag{6.36}$$

$$\mathcal{O}(\langle \ell \rangle) = N^{-\frac{1}{2}} \tag{6.37}$$

$$\mathcal{O}(\sum) = N \tag{6.38}$$

$$\mathcal{O}(s^2) = N^{-2} \tag{6.39}$$

So we find

$$\mathcal{O}(P_t) = \mathcal{O}(\Sigma)\ \mathcal{O}(L)\ \mathcal{O}(F^2)\ \mathcal{O}(C^{-1}) + \mathcal{O}(\Sigma)\ \mathcal{O}(L)\ \mathcal{O}(C^{1/2}) = N^{\frac{1}{6}} + N^{\frac{1}{6}} = N^{\frac{1}{6}} \tag{6.40}$$

and analogously

$$\mathcal{O}(P_d) = \mathcal{O}(\Sigma)\ \mathcal{O}(\langle \ell \rangle)\ \mathcal{O}(s^2) = N^{-3/2}. \tag{6.41}$$

Therefore, the resulting power ratio scales as

$$\mathcal{O}(\Omega) = \frac{\mathcal{O}(P_d)}{\mathcal{O}(P_t)} = \frac{N^{-3/2}}{N^{1/6}} = N^{-10/6} = N^{-5/3} \tag{6.42}$$

which changes the scaling with regards to the one dimensional case.

A4 Differentiable Voronoi

In this section, we show an overview of the clipped Voronoi tessellation implementation used. Notably, our approach to make it differentiable relies on a pre-calculation using SciPy, and leveraging that information to efficiency construct the resulting diagram.

Algorithm 1: Differentiable Voronoi Tessellation Clipped by a Boundary

Input: Set of points $x = \{(x_i, y_i)\}$, convex boundary polygon boundary
Output: Clipped Voronoi nodes and regions
Pre-calculation (Non-differentiable, CPU):
Extend x:

$$x_{\text{ext}} = x \cup (\text{boundary} + 100 \cdot (\text{boundary} - \text{mean}(x)))$$

Compute Voronoi regions, Delaunay triangulation, and mark regions needing clipping.
Differentiable Voronoi (GPU-friendly):
foreach *triangle* $\{(a_x, a_y), (b_x, b_y), (c_x, c_y)\}$ *in Delaunay* **do**
  Compute Voronoi node (u_x, u_y):

$$D = 2((a_x - c_x)(b_y - c_y) - (b_x - c_x)(a_y - c_y))$$
$$u_x = \frac{\sum(p_x^2 + p_y^2) \cdot (\Delta_y)}{D}, \quad u_y = \frac{\sum(p_x^2 + p_y^2) \cdot (\Delta_x)}{D}$$

Clipping Regions:
foreach *Voronoi region needing clipping* **do**
  foreach *edge* (A, B) *of the region* **do**
    Find intersections with boundary edges (C, D):

$$u_a = \frac{(D - C) \times (A - C)}{(B - A) \times (D - C)}$$

    if $0 \leq u_a \leq 1$ **then**
      Compute intersection: $(x, y) = A + u_a \cdot (B - A)$
  Sort clipped vertices by angle θ relative to the centroid:

$$\theta = -\arctan 2(y - y_{\text{centroid}}, x - x_{\text{centroid}})$$

Return clipped Voronoi nodes and regions.

Since this implementation is written with GPU in mind, most of the loops are instead vector operations, and calculation occur on its majority in parallel. For calculating the gradients we rely on jax [37] automatic differentiation rules.

A5 Implicit Function Theorem

During the optimization of the nodes positions, we perform a full optimization of the conductivities of the system. To do so, we use the dynamical model defined in the main text as

$$\frac{\mathrm{d}C_e}{\mathrm{d}t} = \left(\frac{F_e^2}{C_e^{\gamma+1}} - \gamma c_t\right) C_e + c_0 e^{-\lambda t}, \tag{6.43}$$

until it has reached the steady state $\boldsymbol{C}^*$. The steady state solution is equivalent to finding the fixed point of the equation $\frac{\mathrm{d}C_e}{\mathrm{d}t} = 0$. From now on, we are interested in solving for the implicit function $f(C^*, x) = 0$.

The result $\boldsymbol{C}^*$ is dependent on the positions of the nodes $\boldsymbol{x}$ in a non-trivial manner. Namely, $\boldsymbol{C}^*$ depends on $\boldsymbol{F}_e$, which is a function of $\boldsymbol{L}_e$ and $\boldsymbol{s}$, all of which dependent on $\boldsymbol{x}$. We are interested in finding $\frac{dC}{dx}$, and even though automatic differentiation would allow to apply the chain rule through the iterative solver, the memory and computational cost of calculating derivatives and storing them would make it computationally challenging. Luckily, since we are iterating over an implicit function, we can make use of the implicit function theorem [21, 308], which allows for backpropagation through a fixed-point iteration without requiring saving each iterative step into memory.

For simplicity of presentation, here we will consider that the power only depends on the optimal conductivities, i.e.

$$\frac{dP}{dx} = \frac{dP}{dC^*} \cdot \frac{dC^*}{dx}. \tag{6.44}$$

In order to compute $\frac{dC^*}{dx}$ (the gradients of interest here), we use its implicit form $f(C, x) = 0$ and obtain

$$\frac{df}{dx} = \frac{\partial f}{\partial C} \cdot \frac{dC^*}{dx} + \frac{\partial f}{\partial x} = 0. \tag{6.45}$$

Letting $A = \frac{\partial f}{\partial C}$ (the Jacobian of $\boldsymbol{f}$ with respect to $\boldsymbol{C}$) and $B = \frac{\partial f}{\partial x}$ (the Jacobian of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$), we isolate $\frac{\partial C^*}{\partial x}$:

$$\frac{dC^*}{dx} = -A^{-1}B. \tag{6.46}$$

This Jacobian is memory intensive (here we are ignoring the dependencies on the solver of $\boldsymbol{F}$). However, we can rephrase this as a fixed-point problem using the adjoint vector w given by

$$w^T = v^T A. \tag{6.47}$$

where $v = \frac{\partial f}{\partial C^*}$. This means that we can solve for w iteratively if we express it in implicit form such that $w = f(w) = w^T - v^t A$ where we are around $f(w) = 0$, and have

$$\frac{df}{dx} = w^T B. \tag{6.48}$$

which is equivalent to finding $\frac{dC}{dx}$ in the regime where $f(C, x) = f(C^*, x) = 0$. Thus, by applying the implicit function theorem, we can compute the gradient with respect to parameters in the adaptation model for conductivity optimization, ensuring efficiency by avoiding the storage and backpropagation through each individual iteration step.

Our implementation of the fixed point approach can be found at https://github.com/kirkegaardlab/gradnodes.

Chapter 7

Final Remarks

The fundamental principles of physics, integrated with the computational tools of computer science and the intricate complexities of biological systems, create a surprisingly exciting playground for scientific exploration. In this thesis, I have explored this interdisciplinary approach employing differentiable programming techniques to study the behaviours of microorganisms, enhance data acquisition methodologies, and investigate how transport networks can adapt to their domain.

Firstly, I took on the lasting challenge of tracking overlapping slender organisms in microscopy images. Something quantitative behavioural studies were in need of. By developing a deep learning solution, the method was able to assign identities to organisms during occlusions and detect thousands of *C. elegans* simultaneously. The creation of a synthetic dataset using force resistive theory was crucial, as it accurately mimicked the motions of *in vitro* swimming nematodes at low Reynolds numbers. This not only provided a robust training ground for the deep-learning detection model but also demonstrated the effectiveness of combining physical modelling with machine learning to overcome practical limitations in biological imaging.

The second and third projects explored decision making during chemotaxis, focussing on how physical limits shape optimal strategies by constraining environmental sensing. Although chemotaxis is a well-studied topic, integration with newer and more powerful tools, such as deep reinforcement learning (DRL), is still anecdotal. These projects aimed to bridge this gap and explore questions where optimality is not clearly known.

The first challenge addressed the transition between temporal and spatial sensing strategies. The findings of which revealed a continuous transition in optimal strategies and identified a regime in which both temporal and spatial information sources are efficiently integrated. This continuum challenges the traditional dichotomy of chemotactic strategies and suggests that microorganisms can adapt their sensing mechanisms based on environmental conditions and limitations.

Further, modelling spatial sensing in amoeba as a finite resource contest between protrusions offered insight into how cells could overcome physical limitations in sensing. This mechanism demonstrated that cells could leverage persistence in motion to enhance their chemotactic capabilities, aligning closely with experimental observations.

The simplicity and effectiveness of this model underscore the potential for minimalistic designs in biological systems to achieve complex behaviours through resource allocation and mechanical interactions. Similarly to the previous project, optimality is not well defined due to the complex dynamics of the system. Thus, by employing DRL, we discovered how optimal pseudopod strategies depend on environmental information and how cells adapt their strategies to enhance performance accordingly.

The fourth project investigated the optimal placement of cell-surface receptors for spatial gradient sensing. Specifically, it studied their spatial organisation to reduce the uncertainty in gradient estimation. The results showed that optimally placed receptors tend to cluster in surface protrusions once symmetry is broken. This clustering not only improves the accuracy of the detection, but also aligns with previous mechanical explanations for receptor distribution. The convergence of mechanical and functional explanations suggests an evolutionary link in which physical mechanisms are harnessed for functional advantages, highlighting the interplay between form and function in biological systems.

The final project ventured into the optimisation of transport networks by exploring the optimal distribution of nodes when the cost of resource delivery is included in the energy formulation. By removing the constraint of nodes being placed on a static grid, the research used differentiable physics to determine their optimal placement, a task that is challenging with conventional methods. The resulting networks showed adaptations to boundaries and misalignments, indicating a higher level of efficiency and responsiveness to environmental constraints. Although slightly tangential to the main theme of microorganism behaviour, this project exemplifies the versatility of differentiable programming in solving complex optimisation problems in biological contexts.

Collectively, this thesis highlights the benefits of integrating differentiable programming with physical models to gain a deeper understanding of complex biological systems. The methodologies developed offer a foundation for future applications, demonstrating how computational techniques can bridge gaps between theoretical models and empirical observations. By addressing challenges in tracking, sensing, and optimisation, the research presented in this thesis may contribute to a more comprehensive understanding of cell behaviour and the fundamental principles governing biological systems at microscopic scales.

7.1 Future Directions

The findings of this thesis open up several avenues for further research. In the context of organism tracking, future work could focus on extending the deep learning model to handle a wider variety of microorganisms with different morphological and motility characteristics. Incorporating more complex environmental conditions and interactions into the synthetic datasets could enhance the model's robustness and applicability to real-world scenarios. Similarly, coupling physics-based simulations with generative models can only but extend the capabilities of deep-learning-based models in difficult-to-label tasks.

In exploring chemotactic decision making, subsequent studies might investigate how external factors such as fluctuating environmental gradients or the presence of obstacles influence the integration of temporal and spatial sensing strategies. Furthermore, examining the genetic and molecular bases of resource allocation in protrusion dynamics could provide deeper insights into the mechanisms behind spatial sensing in amoebas and other cells.

The optimal placement of cell-surface receptors presents opportunities to explore the evolutionary pressures that shape receptor distributions. Experimental validation of the theoretical models using advanced imaging techniques could substantiate the proposed link between mechanical structures and functional advantages. Similarly, if in agreement, the experimental evidence would validate whether the cells actually estimate the magnitude and direction of the gradient, analogous to the least squares assumption. Furthermore, investigating receptor dynamics in response to changing environmental conditions could reveal adaptive mechanisms employed by cells. On a more practical note, it would be valuable to investigate whether micro robots benefit from reduced uncertainty in estimation, as suggested by the results of our study.

The optimisation of transport networks could be expanded to model more complex biological systems, such as vascular networks in tissues or neural connectivity patterns. Incorporating additional factors such as network robustness, redundancy, and energy efficiency could lead to more comprehensive models that better reflect the intricacies of biological transport systems.

7.2 Closing thoughts

This thesis has shown the power of interdisciplinary approaches in modelling the complexities of biological systems. By integrating differentiable programming with physical and biological models, this work has addressed long-standing challenges with fresh perspectives and practical solutions. The fact that the projects presented here have not only been accepted for publication, but also positively peer-reviewed (an outcome that remains pleasantly unexpected) showcases the value of computational methods grounded in physical principles for studying biological systems. The research I have conducted over the last three years should hopefully demonstrate that progress in biological sciences is not confined to traditional approaches; but it thrives in the deliberate integration of innovative and interdisciplinary methodologies.

Bibliography

[1] Albert Alonso and Julius B. Kirkegaard. Fast detection of slender bodies in high density microscopy data. *Nature Communications Biology*, 6(1):1–12, 2023. doi: 10.1038/s42003-023-05098-1.

[2] Albert Alonso and Julius B Kirkegaard. Learning optimal integration of spatial and temporal information in noisy chemotaxis. *PNAS Nexus*, 3(7), 2024. doi: 10.1093/pnasnexus/pgae235.

[3] Albert Alonso, Julius B. Kirkegaard, and Robert G. Endres. Persistent pseudopod splitting is an effective chemotaxis strategy in shallow gradients, 2024. (preprint).

[4] Albert Alonso, Robert G. Endres, and Julius B. Kirkegaard. Receptors cluster in high-curvature membrane regions for optimal spatial gradient sensing, 2024.

[5] Albert Alonso, Lars Erik J. Skjegstad, and Julius B. Kirkegaard. Adaptive Node Positioning in Biological Transport Networks, 2024. (preprint – shared first author).

[6] Tuan Pham, Albert Alonso, and Karel Proesmans. Irreversibility in Non-reciprocal Chaotic Systems, 2024. (preprint).

[7] Kirsty Y. Wan and Gáspár Jékely. Origins of eukaryotic excitability. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 376(1820):20190758, January 2021. doi: 10.1098/rstb.2019.0758.

[8] B. F. Skinner. *The Behavior of Organisms: An Experimental Analysis*. B. F. Skinner Foundation, December 2019. ISBN 978-0-9964539-0-5.

[9] Paul Weiss. Cellular Dynamics. *Reviews of Modern Physics*, 31(1):11–20, January 1959. doi: 10.1103/RevModPhys.31.11.

[10] H. C. Berg and E. M. Purcell. Physics of chemoreception. *Biophysical Journal*, 20(2):193–219, November 1977. ISSN 0006-3495. doi: 10.1016/S0006-3495(77)85544-6.

[11] E. M. Purcell. Life at low Reynolds number. *American Journal of Physics*, 45(1):3–11, January 1977. ISSN 0002-9505. doi: 10.1119/1.10903.

[12] David B. Dusenbery. Minimum size limit for useful locomotion by free-swimmingmicrobes. *Proceedings of the National Academy of Sciences*, 94(20):10949–10954, September 1997. doi: 10.1073/pnas.94.20.10949.

[13] Robert G. Endres and Ned S. Wingreen. Accuracy of direct gradient sensing by single cells. *Proceedings of the National Academy of Sciences*, 105(41):15749–15754, October 2008. doi: 10.1073/pnas.0804688105.

[14] Mathieu Blondel and Vincent Roulet. The Elements of Differentiable Programming, March 2024.

[15] Mohammed AlQuraishi and Peter K. Sorger. Differentiable biology: using deep learning for biophysics-based and data-driven modeling of molecular mechanisms. *Nature Methods*, 18(10): 1169–1180, October 2021. ISSN 1548-7105. doi: 10.1038/s41592-021-01283-4.

[16] Wolfram Research Inc. Mathematica, Version 14.1, 2024.

[17] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondej ertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, tpán Rouka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. SymPy: symbolic computing in Python. *PeerJ Computer Science*, 3:e103, January 2017. ISSN 2376-5992. doi: 10.7717/peerj-cs.103.

[18] James F. Epperson. *An Introduction to Numerical Methods and Analysis*. John Wiley & Sons, October 2013. ISBN 978-1-118-36759-9.

[19] Simone Marras, James F. Kelly, Margarida Moragues, Andreas Müller, Michal A. Kopera, Mariano Vázquez, Francis X. Giraldo, Guillaume Houzeaux, and Oriol Jorba. A Review of Element-Based Galerkin Methods for Numerical Weather Prediction: Finite Elements, Spectral Elements, and Discontinuous Galerkin. *Archives of Computational Methods in Engineering*, 23(4):673–722, December 2016. ISSN 1886-1784. doi: 10.1007/s11831-015-9152-1.

[20] L. B. Rall. The Arithmetic of Differentiation. *Mathematics Magazine*, 59(5):275–282, December 1986. ISSN 0025-570X. doi: 10.1080/0025570X.1986.11977261.

[21] Bradley M. Bell and James V. Burke. Algorithmic Differentiation of Implicit Functions and Optimal Values. In Christian H. Bischof, H. Martin Bücker, Paul Hovland, Uwe Naumann, and Jean Utke, editors, *Advances in Automatic Differentiation*, pages 67–77, Berlin, Heidelberg, 2008. ISBN 978-3-540-68942-3. doi: 10.1007/978-3-540-68942-3_7.

[22] J. M. Child. THE MANUSCRIPTS OF LEIBNIZ ON HIS DISCOVERY OF THE DIFFERENTIAL CALCULUS. PART II (Continued). *The Monist*, 27(3):411–454, 1917. ISSN 0026-9662.

[23] Atlm Günes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *J. Mach. Learn. Res.*, 18(1):5595–5637, January 2017. ISSN 1532-4435.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[25] Dion Häfner, René Løwe Jacobsen, Carsten Eden, Mads R. B. Kristensen, Markus Jochum, Roman Nuterman, and Brian Vinter. Veros v0.1 a fast and versatile ocean simulator in pure Python. *Geoscientific Model Development*, 11(8):3299–3312, August 2018. ISSN 1991-959X. doi: 10.5194/gmd-11-3299-2018.

[26] Giuseppe Carleo, Kenny Choo, Damian Hofmann, James E. T. Smith, Tom Westerhout, Fabien Alet, Emily J. Davis, Stavros Efthymiou, Ivan Glasser, Sheng-Hsuan Lin, Marta Mauri, Guglielmo Mazzola, Christian B. Mendl, Evert van Nieuwenburg, Ossian O'Reilly, Hugo Théveniaut, Giacomo Torlai, Filippo Vicentini, and Alexander Wietek. NetKet: a machine learning toolkit for many-body quantum systems. *SoftwareX*, page 100311, 2019. doi: 10.1016/j.softx.2019.100311.

[27] Elena Orlova, Aleksei Ustimenko, Ruoxi Jiang, Peter Y. Lu, and Rebecca Willett. Deep Stochastic Mechanics. In *Proceedings of the 41st International Conference on Machine Learning*, pages 38779–38814, July 2024.

[28] Weiluo Ren, Weizhong Fu, Xiaojie Wu, and Ji Chen. Towards the ground state of molecules via diffusion Monte Carlo on neural networks. *Nature Communications*, 14(1):1860, April 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-37609-3.

[29] Zhe Li, Zixiang Lu, Ruichen Li, Xuelan Wen, Xiang Li, Liwei Wang, Ji Chen, and Weiluo Ren. Symmetry enforced solution of the many-body Schrödinger equation with deep neural network, 2024.

[30] Pao-Hsiung Chiu, Jian Cheng Wong, Chinchun Ooi, My Ha Dao, and Yew-Soon Ong. CAN-PINN: A fast physics-informed neural network based on coupled-automaticnumerical differentiation method. *Computer Methods in Applied Mechanics and Engineering*, 395:114909, May 2022. ISSN 0045-7825. doi: 10.1016/j.cma.2022.114909.

[31] Changkyu Song and Abdeslam Boularias. Identifying Mechanical Models of Unknown Objects with Differentiable Physics Simulations. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, pages 749–760, July 2020.

[32] Davide Piras and Alessio Spurio Mancini. CosmoPower-JAX: high-dimensional Bayesian inference with differentiable cosmological emulators. *The Open Journal of Astrophysics*, 6:20, July 2023. doi: 10.21105/astro.2305.06347.

[33] Samuel Schoenholz and Ekin Dogus Cubuk. JAX MD: A Framework for Differentiable Physics. In *Advances in Neural Information Processing Systems*, volume 33, pages 11428–11441, 2020.

[34] Sergei Shumilin, Alexander Ryabov, Serguei Barannikov, Evgeny Burnaev, and Vladimir Vanovskii. A Method for Auto-Differentiation of the Voronoi Tessellation, May 2024.

[35] Shi-Xin Zhang. Automatic differentiable Monte Carlo: Theory and application. *Physical Review Research*, 5(3), 2023. doi: 10.1103/PhysRevResearch.5.033041.

[36] Albert Alonso. *Evaluation of Google TPUs for High Performance Physics Calculations*. PhD thesis, University of Copenhagen, 2021.

[37] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018.

[38] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library, December 2019.

[39] Dougal Maclaurin, David Duvenaud, and Ryan P Adams. Autograd: Effortless gradients in numpy. In *ICML 2015 AutoML workshop*, volume 238, page 5, 2015.

[40] Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. DiffTaichi: Differentiable Programming for Physical Simulation, February 2020.

[41] Maxwell Aifer, Kaelan Donatella, Max Hunter Gordon, Samuel Duffield, Thomas Ahle, Daniel Simpson, Gavin E. Crooks, and Patrick J. Coles. Thermodynamic Linear Algebra, June 2024.

[42] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[43] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, January 1989. ISSN 0893-6080. doi: 10.1016/0893-6080(89)90020-8.

[44] Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, and Gulshan Kumar. A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning. *Archives of Computational Methods in Engineering*, 27(4):1071–1092, September 2020. ISSN 1886-1784. doi: 10.1007/s11831-019-09344-w.

[45] Zhichao Liu, Luhong Jin, Jincheng Chen, Qiuyu Fang, Sergey Ablameyko, Zhaozheng Yin, and Yingke Xu. A survey on applications of deep learning in microscopy image analysis. *Computers in Biology and Medicine*, 134:104523, July 2021. ISSN 0010-4825. doi: 10.1016/j.compbiomed.2021.104523.

[46] Erick Moen, Dylan Bannon, Takamasa Kudo, William Graf, Markus Covert, and David Van Valen. Deep learning for cellular image analysis. *Nature methods*, 16(12):1233–1246, 2019.

[47] Mireia Cordero, Namiko Mitarai, and Liselotte Jauffred. Motility mediates satellite formation in confined biofilms. *The ISME Journal*, 17(11):1819–1827, November 2023. ISSN 1751-7370. doi: 10.1038/s41396-023-01494-x.

[48] Liang Li, Edward C Cox, and Henrik Flyvbjerg. Dicty dynamics: Dictyostelium motility as persistent random motion. *Physical biology*, 8(4):046006, 2011.

[49] Kristen F. Swaney, Chuan-Hsiang Huang, and Peter N. Devreotes. Eukaryotic Chemotaxis: A Network of Signaling Pathways Controls Motility, Directional Sensing, and Polarity. *Annual Review of Biophysics*, 39(Volume 39, 2010):265–289, June 2010. ISSN 1936-122X, 1936-1238. doi: 10.1146/annurev.biophys.093008.131228.

[50] Lillian K. Fritz-Laylin. The evolution of animal cell motility. *Current Biology*, 30(10):R477–R482, May 2020. ISSN 0960-9822. doi: 10.1016/j.cub.2020.03.026.

[51] Howard C. Berg. Motile Behavior of Bacteria. *Physics Today*, 53(1):24–29, January 2000. ISSN 0031-9228. doi: 10.1063/1.882934.

[52] Jirapat Likitlersuang, Greg Stephens, Konstantine Palanski, and William S Ryu. C. elegans tracking and behavioral measurement. *JoVE*, page e4094, 2012. doi: doi:10.3791/4094.

[53] Wei Geng, Pamela Cosman, Charles C Berry, Zhaoyang Feng, and William R Schafer. Automatic tracking, feature extraction and classification of C. elegans phenotypes. *IEEE transactions on biomedical engineering*, 51(10):1811–1820, 2004.

[54] Michele Perni, Pavan K Challa, Julius B Kirkegaard, Ryan Limbocker, Mandy Koopman, Maarten C Hardenberg, Pietro Sormanni, Thomas Müller, Kadi L Saar, Lianne WY Roode, and others. Massively parallel C. elegans tracking provides multi-dimensional fingerprints for phenotypic discovery. *Journal of neuroscience methods*, 306:57–67, 2018.

[55] Steven D Buckingham, Frederick A Partridge, and David B Sattelle. Automated, high-throughput, motility analysis in Caenorhabditis elegans and parasitic nematodes: Applications in the search for new anthelmintics. *International Journal for Parasitology: Drugs and Drug Resistance*, 4(3):226–232, 2014.

[56] Xiaomin Zheng, Jie Zhou, Lei Wang, Meiting Wang, Wenshuai Wu, Jiajie Chen, Junle Qu, Bruce Zhi Gao, and Yonghong Shao. Current challenges and solutions of super-resolution structured illumination microscopy. *APL Photonics*, 6(2):020901, February 2021. ISSN 2378-0967. doi: 10.1063/5.0038065.

[57] Fuyong Xing, Yuanpu Xie, Hai Su, Fujun Liu, and Lin Yang. Deep learning in microscopy image analysis: A survey. *IEEE transactions on neural networks and learning systems*, 29(10):4550–4568, 2017.

[58] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

[59] Thomas Alscher, Kenny Erleben, and Sune Darkner. Collision-constrained deformable image registration framework for discontinuity management. *PLOS ONE*, 18(8):e0290243, August 2023. ISSN 1932-6203. doi: 10.1371/journal.pone.0290243.

[60] Ebraheem Fontaine, Joel Burdick, and Alan Barr. Automated Tracking of Multiple C. Elegans. In *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 3716–3719, August 2006. doi: 10.1109/IEMBS.2006.260657.

[61] Kathleen Bates, Kim N. Le, and Hang Lu. Deep learning for robust and flexible tracking in behavioral studies for C. elegans. *PLOS Computational Biology*, 18(4):e1009942, April 2022. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1009942.

[62] Talmo D. Pereira, Nathaniel Tabris, Arie Matsliah, David M. Turner, Junyu Li, Shruthi Ravindranath, Eleni S. Papadoyannis, Edna Normand, David S. Deutsch, Z. Yan Wang, Grace C. McKenzie-Smith, Catalin C. Mitelut, Marielisa Diez Castro, John DUva, Mikhail Kislin, Dan H. Sanes, Sarah D. Kocher, Samuel S.-H. Wang, Annegret L. Falkner, Joshua W. Shaevitz, and Mala Murthy. SLEAP: A deep learning system for multi-animal pose tracking. *Nature Methods*, 19(4): 486–495, April 2022. ISSN 1548-7105. doi: 10.1038/s41592-022-01426-1.

[63] Nikzad Babaii Rizvandi, Aleksandra Piurica, Filip Rooms, and Wilfried Philips. Skeleton analysis of population images for detection of isolated and overlapped nematode C. elegans. In *2008 16th European signal processing conference*, pages 1–5, 2008.

[64] B. Moghaddam and A. Pentland. Probabilistic visual learning for object detection. In *Proceedings of IEEE International Conference on Computer Vision*, pages 786–793, June 1995. doi: 10.1109/ICCV.1995.466858.

[65] Chastine Fatichah, Diana Purwitasari, Victor Hariadi, and Faried Effendy. Overlapping White Blood Cell Segmentation And Counting On Microscopic Blood Cell Images. *International Journal on Smart Sensing and Intelligent Systems*, 7(3):1271–1286, January 2022. ISSN 1178-5608. doi: 10.21307/ijssis-2017-705.

[66] Hansang Lee and Junmo Kim. Segmentation of overlapping cervical cells in microscopic images with superpixel partitioning and cell-wise contour refinement. In *2016 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, pages 1367–1373, 2016. doi: 10.1109/CVPRW.2016.172.

[67] Jie Song, Liang Xiao, and Zhichao Lian. Contour-Seed Pairs Learning-Based Framework for Simultaneously Detecting and Segmenting Various Overlapping Cells/Nuclei in Microscopy Images. *IEEE Transactions on Image Processing*, 27(12):5759–5774, December 2018. ISSN 1941-0042. doi: 10.1109/TIP.2018.2857001.

[68] Julius B Kirkegaard. Spontaneous breaking of symmetry in overlapping cell instance segmentation using diffusion models. *Biology Methods and Protocols*, page bpae084, November 2024. ISSN 2396-8923. doi: 10.1093/biomethods/bpae084.

[69] J. Gray and G. J. Hancock. The Propulsion of Sea-Urchin Spermatozoa. *Journal of Experimental Biology*, 32(4):802–814, December 1955. ISSN 0022-0949. doi: 10.1242/jeb.32.4.802.

[70] Antonio C. Costa, Tosif Ahamed, David Jordan, and Greg J. Stephens. A Markovian dynamics for *Caenorhabditis elegans* behavior across scales. *Proceedings of the National Academy of Sciences*, 121(32):e2318805121, August 2024. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.2318805121.

[71] Yanmin Zhou, Zheng Yan, Ye Yang, Zhipeng Wang, Ping Lu, Philip F. Yuan, and Bin He. Bioinspired sensors and applications in intelligent robots: a review. *Robotic Intelligence and Automation*, 44(2):215–228, January 2024. ISSN 2754-6977. doi: 10.1108/RIA-07-2023-0088.

[72] Robert W. Barber and David R. Emerson. Optimal design of microfluidic networks using biologically inspired principles. *Microfluidics and Nanofluidics*, 4(3):179–191, March 2008. ISSN 1613-4982, 1613-4990. doi: 10.1007/s10404-007-0163-6.

[73] Lawren Sack and Christine Scoffoni. Leaf venation: structure, function, development, evolution, ecology and applications in the past, present and future. *New phytologist*, 198(4):983–1000, 2013.

[74] J. Arjan G. M. de Visser and Joachim Krug. Empirical fitness landscapes and the predictability of evolution. *Nature Reviews Genetics*, 15(7):480–490, July 2014. ISSN 1471-0064. doi: 10.1038/nrg3744.

[75] Prateek Jain and Purushottam Kar. Non-convex Optimization for Machine Learning. *Foundations and Trendső in Machine Learning*, 10(3-4):142–363, December 2017. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000058.

[76] Kenji Kawaguchi. Deep Learning without Poor Local Minima. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

[77] Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive Methods for Nonconvex Optimization. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[78] Charles Darwin. *On the origin of species by means of natural selection.* Murray, London, 1859.

[79] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction.* The MIT Press, 2 edition, 2018.

[80] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, 4:237–285, May 1996. ISSN 1076-9757. doi: 10.1613/jair.301.

[81] James Gillespie, Iñaki Rañó, Nazmul Siddique, José Santos, and Mehdi Khamassi. Using reinforcement learning to attenuate for stochasticity in robot navigation controllers. In *2019 IEEE symposium series on computational intelligence (SSCI)*, pages 705–713, 2019. doi: 10.1109/SSCI44817.2019.9002834.

[82] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with Deep Reinforcement Learning, December 2013.

[83] Simona Colabrese, Kristian Gustavsson, Antonio Celani, and Luca Biferale. Flow Navigation by Smart Microswimmers via Reinforcement Learning. *Physical Review Letters*, 118(15):158004, April 2017. doi: 10.1103/PhysRevLett.118.158004.

[84] Eckart Zitzler. Evolutionary algorithms for multiobjective optimization: methods and applications. *Berichte aus der Informatik*, 030, 1999. doi: 10.3929/ethz-a-003856832.

[85] Probir Kumar Ray and Ajay Mahajan. A genetic algorithm-based approach to calculate the optimal configuration of ultrasonic sensors in a 3D position estimation system. *Robotics and Autonomous Systems*, 41(4):165–177, December 2002. ISSN 0921-8890. doi: 10.1016/S0921-8890(02)00292-0.

[86] Benedikt Hartl, Maximilian Hübl, Gerhard Kahl, and Andreas Zöttl. Microswimmers learning chemotaxis with genetic algorithms. *Proceedings of the National Academy of Sciences*, 118(19): e2019683118, May 2021. doi: 10.1073/pnas.2019683118.

[87] Hui Bai, Ran Cheng, and Yaochu Jin. Evolutionary Reinforcement Learning: A Survey. *Intelligent Computing*, 2:0025, January 2023. doi: 10.34133/icomputing.0025.

[88] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, January 2016. ISSN 1476-4687. doi: 10.1038/nature16961.

[89] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature*, 588 (7839):604–609, December 2020. ISSN 1476-4687. doi: 10.1038/s41586-020-03051-4.

[90] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin ídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596 (7873):583–589, August 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03819-2.

[91] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 3319–3328, Sydney, NSW, Australia, August 2017.

[92] Michael Eisenbach. Bacterial Chemotaxis. In *Encyclopedia of Life Sciences*. John Wiley & Sons, 2011. ISBN 978-0-470-01590-2. doi: 10.1002/9780470015902.a0001251.pub3.

[93] Benjamin M. Friedrich and Frank Jülicher. Chemotaxis of sperm cells. *Proceedings of the National Academy of Sciences*, 104(33):13256–13261, August 2007. doi: 10.1073/pnas.0703530104.

[94] Herbert Levine and Wouter-Jan Rappel. The physics of eukaryotic chemotaxis. *Physics Today*, 66 (2):24–30, February 2013. ISSN 0031-9228. doi: 10.1063/PT.3.1884.

[95] Douglas R. Brumley, Francesco Carrara, Andrew M. Hein, Yutaka Yawata, Simon A. Levin, and Roman Stocker. Bacteria push the limits of chemotactic precision to navigate dynamic chemical gradients. *Proceedings of the National Academy of Sciences*, 116(22):10792–10797, May 2019. doi: 10.1073/pnas.1816621116.

[96] Julius B Kirkegaard and Raymond E Goldstein. The role of tumbling frequency and persistence in optimal run-and-tumble chemotaxis. *IMA Journal of Applied Mathematics*, 83(4):700–719, July 2018. ISSN 0272-4960. doi: 10.1093/imamat/hxy013.

[97] William Bialek and Sima Setayeshgar. Physical limits to biochemical signaling. *Proceedings of the National Academy of Sciences*, 102(29):10040–10045, July 2005. doi: 10.1073/pnas.0504321102.

[98] Gerardo Aquino, Ned S. Wingreen, and Robert G. Endres. Know the Single-Receptor Sensing Limit? Think Again. *Journal of Statistical Physics*, 162(5):1353–1364, March 2016. ISSN 1572-9613. doi: 10.1007/s10955-015-1412-9.

[99] Antonio Celani and Massimo Vergassola. Bacterial strategies for chemotaxis response. *Proceedings of the National Academy of Sciences*, 107(4):1391–1396, January 2010. doi: 10.1073/pnas.0909673107.

[100] Robert G. Endres and Ned S. Wingreen. Accuracy of direct gradient sensing by cell-surface receptors. *Progress in Biophysics and Molecular Biology*, 100(1):33–39, September 2009. ISSN 0079-6107. doi: 10.1016/j.pbiomolbio.2009.06.002.

[101] Shuangyu Bi and Victor Sourjik. Stimulus sensing and signal processing in bacterial chemotaxis. *Current Opinion in Microbiology*, 45:22–29, October 2018. ISSN 1369-5274. doi: 10.1016/j.mib.2018.02.002.

[102] Masahiro Ueda and Tatsuo Shibata. Stochastic Signal Processing and Transduction in Chemotactic Response of Eukaryotic Cells. *Biophysical Journal*, 93(1):11–20, July 2007. ISSN 0006-3495. doi: 10.1529/biophysj.106.100263.

[103] J. E. Avron, O. Gat, and O. Kenneth. Optimal Swimming at Low Reynolds Numbers. *Physical Review Letters*, 93(18):186001, October 2004. doi: 10.1103/PhysRevLett.93.186001.

[104] Andrew Mugler, Andre Levchenko, and Ilya Nemenman. Limits to the precision of gradient sensing with spatial communication and temporal integration. *Proceedings of the National Academy of Sciences*, 113(6):E689–E695, February 2016. doi: 10.1073/pnas.1509597112.

[105] Peter J.M. van Haastert and Marten Postma. Biased Random Walk by Stochastic Fluctuations of Chemoattractant-Receptor Interactions at the Lower Limit of Detection. *Biophysical Journal*, 93 (5):1787–1796, September 2007. ISSN 0006-3495. doi: 10.1529/biophysj.107.104356.

[106] Rui Zhen Tan and Keng-Hwee Chiam. A computational model for how cells choose temporal or spatial sensing during chemotaxis. *PLOS Computational Biology*, 14(3):e1005966, March 2018. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1005966.

[107] David B. Dusenbery. Spatial Sensing of Stimulus Gradients Can Be Superior to Temporal Sensing for Free-Swimming Bacteria. *Biophysical Journal*, 74(5):2272–2277, May 1998. ISSN 0006-3495. doi: 10.1016/S0006-3495(98)77936-6.

[108] Julian Rode, Maja Novak, and Benjamin M. Friedrich. Information Theory of Chemotactic Agents Using Both Spatial and Temporal Gradient Sensing. *PRX Life*, 2(2):023012, June 2024. doi: 10.1103/PRXLife.2.023012.

[109] Claus Metzner. On the efficiency of chemotactic pursuit - Comparing blind search with temporal and spatial gradient sensing. *Scientific Reports*, 9(1):14091, October 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-50514-4.

[110] Charles DeLisi and Federico Marchetti. A theory of measurement error and its implications for spatial and temporal gradient sensing during chemotaxis. *Cell Biophysics*, 5(4):237–253, December 1983. ISSN 1559-0283. doi: 10.1007/BF02788623.

[111] Elleard F. W. Heffern, Holly Huelskamp, Sonya Bahar, and R. Fredrik Inglis. Phase transitions in biology: from bird flocks to population dynamics. *Proceedings of the Royal Society B: Biological Sciences*, 288(1961):20211111, October 2021. doi: 10.1098/rspb.2021.1111.

[112] Peter J. M. Van Haastert. Chemotaxis: insights from the extending pseudopod. *Journal of Cell Science*, 123(18):3031–3037, September 2010. ISSN 0021-9533. doi: 10.1242/jcs.071118.

[113] Lawrence Yolland, Mubarik Burki, Stefania Marcotti, Andrei Luchici, Fiona N. Kenny, John Robert Davis, Eduardo Serna-Morales, Jan Müller, Michael Sixt, Andrew Davidson, Will Wood, Linus J. Schumacher, Robert G. Endres, Mark Miodownik, and Brian M. Stramer. Persistent and polarized global actin flow is essential for directionality during cell migration. *Nature Cell Biology*, 21(11):1370–1381, November 2019. ISSN 1476-4679. doi: 10.1038/s41556-019-0411-5.

[114] Peter J. M. van Haastert. Unified control of amoeboid pseudopod extension in multiple organisms by branched F-actin in the front and parallel F-actin/myosin in the cortex. *PLOS ONE*, 15(12): e0243442, December 2020. ISSN 1932-6203. doi: 10.1371/journal.pone.0243442.

[115] Nicolas Ecker and Karsten Kruse. Excitable actin dynamics and amoeboid cell migration. *PLOS ONE*, 16(2):e0246311, February 2021. ISSN 1932-6203. doi: 10.1371/journal.pone.0246311.

[116] Simona Sorrentino, Jose Javier Conesa, Ana Cuervo, Roberto Melero, Bruno Martins, Estrella Fernandez-Gimenez, Federico P. de Isidro-Gomez, Jimenez de la Morena, Jan-Dirk Studt, Carlos Oscar S. Sorzano, Matthias Eibauer, Jose Maria Carazo, and Ohad Medalia. Structural analysis of receptors and actin polarity in platelet protrusions. *Proceedings of the National Academy of Sciences*, 118(37):e2105004118, September 2021. doi: 10.1073/pnas.2105004118.

[117] Markus Affolter and Cornelis J. Weijer. Signaling to Cytoskeletal Dynamics during Chemotaxis. *Developmental Cell*, 9(1):19–34, July 2005. ISSN 1534-5807. doi: 10.1016/j.devcel.2005.06.003.

[118] Luke Tweedy, Börn Meier, Jürgen Stephan, Doris Heinrich, and Robert G. Endres. Distinct cell shapes determine accurate chemotaxis. *Scientific Reports*, 3(1):2606, September 2013. ISSN 2045-2322. doi: 10.1038/srep02606.

[119] Hans Meinhardt. Orientation of chemotactic cells and growth cones: models and mechanisms. *Journal of Cell Science*, 112(17):2867–2874, September 1999. ISSN 0021-9533. doi: 10.1242/jcs.112.17.2867.

[120] Matthew P. Neilson, Douwe M. Veltman, Peter J. M. van Haastert, Steven D. Webb, John A. Mackenzie, and Robert H. Insall. Chemotaxis: A Feedback-Based Computational Model Robustly Predicts Multiple Aspects of Real Cell Behaviour. *PLOS Biology*, 9(5):e1000618, May 2011. ISSN 1545-7885. doi: 10.1371/journal.pbio.1000618.

[121] Natalie Andrew and Robert H. Insall. Chemotaxis in shallow gradients is mediated independently of PtdIns 3-kinase by biased choices between random protrusions. *Nature Cell Biology*, 9(2): 193–200, February 2007. ISSN 1476-4679. doi: 10.1038/ncb1536.

[122] Leonard Bosgraaf and Peter J. M. Van Haastert. Navigation of Chemotactic Cells by Parallel Signaling to Pseudopod Persistence and Orientation. *PLOS ONE*, 4(8):e6842, August 2009. ISSN 1932-6203. doi: 10.1371/journal.pone.0006842.

[123] Darren Pais, Patrick M. Hogan, Thomas Schlegel, Nigel R. Franks, Naomi E. Leonard, and James A. R. Marshall. A Mechanism for Value-Sensitive Decision-Making. *PLOS ONE*, 8(9):e73216, September 2013. ISSN 1932-6203. doi: 10.1371/journal.pone.0073216.

[124] Junbang Liang and Ming C. Lin. Differentiable physics simulation. In *ICLR 2020 workshop on integration of deep neural models and differential equations*, 2019.

[125] Filipe de Avila Belbute-Peres, Kevin Smith, Kelsey Allen, Josh Tenenbaum, and J. Zico Kolter. End-to-End Differentiable Physics for Learning and Control. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[126] P. M. Janssens and P. J. Van Haastert. Molecular basis of transmembrane signal transduction in Dictyostelium discoideum. *Microbiological Reviews*, 51(4):396–418, December 1987. ISSN 0146-0749. doi: 10.1128/mr.51.4.396-418.1987.

[127] David Kentner, Sebastian Thiem, Markus Hildenbeutel, and Victor Sourjik. Determinants of chemoreceptor cluster formation in Escherichia coli. *Molecular Microbiology*, 61(2):407–417, 2006. ISSN 1365-2958. doi: 10.1111/j.1365-2958.2006.05250.x.

[128] M. Florencia Sánchez and Robert Tampé. Ligand-independent receptor clustering modulates transmembrane signaling: a new paradigm. *Trends in Biochemical Sciences*, 48(2):156–171, February 2023. ISSN 0968-0004. doi: 10.1016/j.tibs.2022.08.002.

[129] Kazunari Kaizu, Wiet deăRonde, Joris Paijmans, Koichi Takahashi, Filipe Tostevin, and Pieter Rein tenăWolde. The Berg-Purcell Limit Revisited. *Biophysical Journal*, 106(4):976–985, February 2014. ISSN 0006-3495. doi: 10.1016/j.bpj.2013.12.030.

[130] Moriah Koler, Eliran Peretz, Chetan Aditya, Thomas S. Shimizu, and Ady Vaknin. Long-term positioning and polar preference of chemoreceptor clusters in E. coli. *Nature Communications*, 9 (1):4444, October 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-06835-5.

[131] Julien Varennes, Bumsoo Han, and Andrew Mugler. Collective Chemotaxis through Noisy Multicellular Gradient Sensing. *Biophysical Journal*, 111(3):640–649, August 2016. ISSN 0006-3495. doi: 10.1016/j.bpj.2016.06.040.

[132] He-Peng Zhang, Avraham Beer, E-L Florin, and Harry L Swinney. Collective motion and density fluctuations in bacterial colonies. *Proceedings of the National Academy of Sciences*, 107(31): 13626–13630, 2010.

[133] Christina Oettmeier, Jonghyun Lee, and Hans-Günther Döbereiner. Form follows function: ultrastructure of different morphotypes of *Physarum polycephalum*. *Journal of Physics D: Applied Physics*, 51(13):134006, April 2018. ISSN 0022-3727, 1361-6463. doi: 10.1088/1361-6463/aab147.

[134] Lisa Schick, Mirna Kramar, and Karen Alim. Dynamic Cost Allocation Allows Network-Forming Forager to Switch Between Search Strategies. *PRX Life*, 2(3):033005, August 2024. doi: 10.1103/PRXLife.2.033005.

[135] James N. Wilking, Vasily Zaburdaev, Michael De Volder, Richard Losick, Michael P. Brenner, and David A. Weitz. Liquid transport facilitated by channels in Bacillus subtilis biofilms. *Proceedings of the National Academy of Sciences*, 110(3):848–852, January 2013. doi: 10.1073/pnas.1216376110.

[136] Julius B. Kirkegaard and Kim Sneppen. Optimal Transport Flows for Distributed Production Networks. *Physical Review Letters*, 124(20):208101, May 2020. doi: 10.1103/PhysRevLett.124.208101.

[137] Katherine A. McCulloh, John S. Sperry, and Frederick R. Adler. Water transport in plants obeys Murray's law. *Nature*, 421(6926):939–942, February 2003. ISSN 1476-4687. doi: 10.1038/nature01444.

[138] Peter R. Corridon. In vitro investigation of the impact of pulsatile blood flow on the vascular architecture of decellularized porcine kidneys. *Scientific Reports*, 11(1):16965, August 2021. ISSN 2045-2322. doi: 10.1038/s41598-021-95924-5.

[139] Lars Erik J. Skjegstad and Julius B. Kirkegaard. Modeling Full-Scale Leaf Venation Networks, October 2024.

[140] Eleni Katifori. The transport network of a leaf. *Comptes Rendus. Physique*, 19(4):244–252, 2018.

[141] Alessandro Lonardi, Enrico Facca, Mario Putti, and Caterina De Bacco. Designing optimal networks for multicommodity transport problem. *Physical Review Research*, 3(4):043010, 2021.

[142] Daniela Leite and Caterina De Bacco. Similarity and economy of scale in urban transportation networks and optimal transport-based infrastructures. *Nature Communications*, 15(1):7981, September 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-52313-6.

[143] Cecil D Murray. The physiological principle of minimum work: I. The vascular system and the cost of blood volume. *Proceedings of the National Academy of Sciences*, 12(3):207–214, 1926.

[144] Charles A Price, Sarah-Jane C Knox, and Tim J Brodribb. The influence of branch order on optimal leaf vein geometries: Murrays law and area preserving branching. *PloS one*, 8(12):e85420, 2013.

[145] Steffen Bohn and Marcelo O. Magnasco. Structure, Scaling, and Phase Transition in the Optimal Transport Network. *Physical Review Letters*, 98(8):088702, February 2007. doi: 10.1103/PhysRevLett.98.088702.

[146] Dan Hu and David Cai. Adaptation and Optimization of Biological Transport Networks. *Physical Review Letters*, 111(13):138701, September 2013. doi: 10.1103/PhysRevLett.111.138701.

[147] Henrik Ronellenfitsch and Eleni Katifori. Global Optimization, Local Adaptation, and the Role of Growth in Distribution Networks. *Physical Review Letters*, 117(13):138301, September 2016. doi: 10.1103/PhysRevLett.117.138301.

[148] Eleni Katifori, Gergely J. Szöllsi, and Marcelo O. Magnasco. Damage and Fluctuations Induce Loops in Optimal Transport Networks. *Physical Review Letters*, 104(4):048704, January 2010. doi: 10.1103/PhysRevLett.104.048704.

[149] Francis Corson. Fluctuations and Redundancy in Optimal Transport Networks. *Physical Review Letters*, 104(4):048703, January 2010. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.104.048703.

[150] Nikhil R Pal and Sankar K Pal. A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294, 1993.

[151] Dinesh D Patil and Sonal G Deore. Medical image segmentation: a review. *International Journal of Computer Science and Mobile Computing*, 2(1):22–27, 2013.

[152] Dzung L Pham, Chenyang Xu, and Jerry L Prince. A survey of current methods in medical image segmentation. *Annual review of biomedical engineering*, 2(3):315–337, 2000.

[153] Anne E Carpenter, Thouis R Jones, Michael R Lamprecht, Colin Clarke, In Han Kang, Ola Friman, David A Guertin, Joo Han Chang, Robert A Lindquist, Jason Moffat, and others. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome biology*, 7(10): 1–11, 2006.

[154] Rainer Pepperkok and Jan Ellenberg. High-throughput fluorescence microscopy for systems biology. *Nature reviews Molecular cell biology*, 7(9):690–696, 2006.

[155] Juan C Caicedo, Sam Cooper, Florian Heigwer, Scott Warchal, Peng Qiu, Csaba Molnar, Aliaksei S Vasilevich, Joseph D Barry, Harmanjit Singh Bansal, Oren Kraus, and others. Data-analysis strategies for image-based cell profiling. *Nature methods*, 14(9):849–863, 2017.

[156] David A Van Valen, Takamasa Kudo, Keara M Lane, Derek N Macklin, Nicolas T Quach, Mialy M DeFelice, Inbal Maayan, Yu Tanouchi, Euan A Ashley, and Markus W Covert. Deep learning automates the quantitative analysis of individual cells in live-cell imaging experiments. *PLoS computational biology*, 12(11):e1005177, 2016.

[157] Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, and others. U-Net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.

[158] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241, 2015.

[159] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.

[160] Noah F Greenwald, Geneva Miller, Erick Moen, Alex Kong, Adam Kagel, Thomas Dougherty, Christine Camacho Fullaway, Brianna J McIntosh, Ke Xuan Leow, Morgan Sarah Schwartz, and others. Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature biotechnology*, 40(4):555–565, 2022.

[161] Reka Hollandi, Abel Szkalisity, Timea Toth, Ervin Tasnadi, Csaba Molnar, Botond Mathe, Istvan Grexa, Jozsef Molnar, Arpad Balind, Mate Gorbe, and others. nucleAIzer: a parameter-free deep learning framework for nucleus segmentation using image style transfer. *Cell Systems*, 10(5): 453–458, 2020.

[162] Arif Ahmed Sekh, Ida S. Opstad, Gustav Godtliebsen, Åsa Birna Birgisdottir, Balpreet Singh Ahluwalia, Krishna Agarwal, and Dilip K. Prasad. Physics-based machine learning for subcellular segmentation in living cells. *Nature Machine Intelligence*, 3(12):1071–1080, December 2021. ISSN 2522-5839. doi: 10.1038/s42256-021-00420-0.

[163] Paul Lang, Karen Yeow, Anthony Nichols, and Alexander Scheer. Cellular imaging in drug discovery. *Nature Reviews Drug Discovery*, 5(4):343–356, 2006.

[164] Mitko Veta, Paul J van Diest, and Josien PW Pluim. Cutting out the middleman: measuring nuclear area in histopathology slides without segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 632–639, 2016.

[165] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017.

[166] Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from nonsmall cell lung cancer histopathology images using deep learning. *Nature medicine*, 24(10):1559–1567, 2018.

[167] Mahdieh Poostchi, Kamolrat Silamut, Richard J Maude, Stefan Jaeger, and George Thoma. Image analysis and machine learning for detecting malaria. *Translational Research*, 194:36–55, 2018.

[168] Vibor Laketa. Microscopy in infectious disease researchImaging across scales. *Journal of molecular biology*, 430(17):2612–2625, 2018.

[169] Gordon J Berman. Measuring behavior across scales. *BMC biology*, 16(1):1–11, 2018.

[170] John W Krakauer, Asif A Ghazanfar, Alex Gomez-Marin, Malcolm A MacIver, and David Poeppel. Neuroscience needs behavior: correcting a reductionist bias. *Neuron*, 93(3):480–490, 2017.

[171] Gopal P Sarma, Chee Wai Lee, Tom Portegys, Vahid Ghayoomie, Travis Jacobs, Bradly Alicea, Matteo Cantarelli, Michael Currie, Richard C Gerkin, Shane Gingell, and others. OpenWorm: overview and recent advances in integrative biological simulation of Caenorhabditis elegans. *Philosophical Transactions of the Royal Society B*, 373(1758):20170382, 2018.

[172] Kelsey M Hallinen, Ross Dempsey, Monika Scholz, Xinwei Yu, Ashley Linder, Francesco Randi, Anuj K Sharma, Joshua W Shaevitz, and Andrew M Leifer. Decoding locomotion from population neural activity in moving C. elegans. *Elife*, 10:e66135, 2021.

[173] Linda Turner, Liam Ping, Marianna Neubauer, and Howard C Berg. Visualizing flagella while tracking bacteria. *Biophysical journal*, 111(3):630–639, 2016.

[174] Marco Polin, Idan Tuval, Knut Drescher, Jerry P Gollub, and Raymond E Goldstein. Chlamydomonas swims with two gears in a eukaryotic version of run-and-tumble locomotion. *Science*, 325 (5939):487–490, 2009.

[175] Andrew Adamatzky. Neuroscience without neurons. *AIP Conference Proceedings*, 2425(1):390001, 2022. doi: 10.1063/5.0082008.

[176] David Kokel and Randall T Peterson. Using the zebrafish photomotor response for psychotropic drug screening. In *Methods in cell biology*, volume 105, pages 517–524. Academic Press, 2011.

[177] Linda P O'Reilly, Cliff J Luke, David H Perlmutter, Gary A Silverman, and Stephen C Pak. C. elegans in high-throughput drug discovery. *Advanced drug delivery reviews*, 69:247–253, 2014.

[178] Demetrio Raldua and Benjamin Pina. In vivo zebrafish assays for analyzing drug toxicity. *Expert opinion on drug metabolism & toxicology*, 10(5):685–697, 2014.

[179] Adam Michael Stewart, Robert Gerlai, and Allan V Kalueff. Developing highER-throughput zebrafish screens for in-vivo CNS drug discovery. *Frontiers in behavioral neuroscience*, 9:14, 2015.

[180] Michele Perni, Céline Galvagnion, Alexander Maltsev, Georg Meisl, Martin BD Müller, Pavan K Challa, Julius B Kirkegaard, Patrick Flagmeier, Samuel IA Cohen, Roberta Cascella, and others. A natural product inhibits the initiation of -synuclein aggregation and suppresses its toxicity. *Proceedings of the National Academy of Sciences*, 114(6):E1009–E1017, 2017.

[181] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[182] Nicolas Roussel, Christine A Morton, Fern P Finger, and Badrinath Roysam. A computational model for C. elegans locomotory behavior: application to multiworm tracking. *IEEE transactions on biomedical engineering*, 54(10):1786–1797, 2007.

[183] Veikko F Geyer, Frank Jülicher, Jonathon Howard, and Benjamin M Friedrich. Cell-body rocking is a dominant mechanism for flagellar synchronization in a swimming alga. *Proceedings of the National Academy of Sciences*, 110(45):18058–18063, 2013.

[184] Kirsty Y Wan, Kyriacos C Leptos, and Raymond E Goldstein. Lag, lock, sync, slip: the many phases of coupled flagella. *Journal of the Royal Society Interface*, 11(94):20131160, 2014.

[185] Carolina Wählby, Lee Kamentsky, Zihan H. Liu, Tammy Riklin-Raviv, Annie L. Conery, Eyleen J. O'Rourke, Katherine L. Sokolnicki, Orane Visvikis, Vebjorn Ljosa, Javier E. Irazoqui, Polina Golland, Gary Ruvkun, Frederick M. Ausubel, and Anne E. Carpenter. An image analysis toolbox for high-throughput C. elegans assays. *Nature Methods*, 9(7):714–716, July 2012. ISSN 1548-7105. doi: 10.1038/nmeth.1984.

[186] Pascal Laube, Matthias O Franz, and Georg Umlauf. Deep learning parametrization for B-spline curve approximation. In *2018 International Conference on 3D Vision (3DV)*, pages 691–699, 2018.

[187] Jun Gao, Chengcheng Tang, Vignesh Ganapathi-Subramanian, Jiahui Huang, Hao Su, and Leonidas J Guibas. Deepspline: Data-driven reconstruction of parametric curves and surfaces. *arXiv preprint arXiv:1901.03781*, 2019.

[188] Soham Mandal and Virginie Uhlmann. Splinedist: Automated cell segmentation with spline curves. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1082–1086, 2021.

[189] Greg J Stephens, Bethany Johnson-Kerner, William Bialek, and William S Ryu. Dimensionality and dynamics in the behavior of C. elegans. *PLoS computational biology*, 4(4):e1000028, 2008.

[190] André EX Brown, Eviatar I Yemini, Laura J Grundy, Tadas Jucikas, and William R Schafer. A dictionary of behavioral motifs reveals clusters of genes affecting Caenorhabditis elegans locomotion. *Proceedings of the National Academy of Sciences*, 110(2):791–796, 2013.

[191] Tosif Ahamed, Antonio C Costa, and Greg J Stephens. Capturing the continuous complexity of behaviour in Caenorhabditis elegans. *Nature Physics*, 17(2):275–283, 2021.

[192] Noel R Krieg, Joseph P Tomelty, and J Scott Wells Jr. Inhibitio of Flagellar Coordination in Spirillum volutans. *Journal of Bacteriology*, 94(5):1431–1436, 1967.

[193] Dhruv K Vig and Charles W Wolgemuth. Swimming dynamics of the Lyme disease spirochete. *Physical review letters*, 109(21):218104, 2012.

[194] David J. Hampson. The Spirochete Brachyspira pilosicoli, Enteric Pathogen of Animals and Humans. *Clinical Microbiology Reviews*, 31(1):e00087–17, November 2017. doi: 10.1128/CMR.00087-17.

[195] Navish Wadhwa and Howard C Berg. Bacterial motility: machinery and mechanisms. *Nature Reviews Microbiology*, 20(3):161–173, 2022.

[196] David M Woolley, Rachel F Crockett, William DI Groom, and Stuart G Revell. A study of synchronisation between the flagella of bull spermatozoa, with related observations. *Journal of Experimental Biology*, 212(14):2215–2223, 2009.

[197] Trine B. Haugen, Steven A. Hicks, Jorunn M. Andersen, Oliwia Witczak, Hugo L. Hammer, Rune Borgli, Pål Halvorsen, and Michael Riegler. VISEM: A Multimodal Video Dataset of Human Spermatozoa. In *Proceedings of the 10th ACM Multimedia Systems Conference*, MMSys '19, pages 261–266, New York, NY, USA, 2019. doi: 10.1145/3304109.3325814.

[198] Linda Turner, William S Ryu, and Howard C Berg. Real-time imaging of fluorescent flagellar filaments. *Journal of bacteriology*, 182(10):2793–2801, 2000.

[199] Laurence G Wilson, Lucy M Carter, and Sarah E Reece. High-speed holographic microscopy of malaria parasites reveals ambidextrous flagellar waveforms. *Proceedings of the National Academy of Sciences*, 110(47):18769–18774, 2013.

[200] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[201] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[202] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[203] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, and others. Recent advances in convolutional neural networks. *Pattern recognition*, 77:354–377, 2018.

[204] Piali Sengupta and Aravinthan DT Samuel. Caenorhabditis elegans: a model system for systems neuroscience. *Current opinion in neurobiology*, 19(6):637–643, 2009.

[205] Maria Markaki and Nektarios Tavernarakis. Caenorhabditis elegans as a model system for human diseases. *Current opinion in biotechnology*, 63:118–125, 2020.

[206] Jesse M Gray, Joseph J Hill, and Cornelia I Bargmann. A circuit for navigation in Caenorhabditis elegans. *Proceedings of the National Academy of Sciences*, 102(9):3184–3191, 2005.

[207] Catherine H Rankin, Christine DO Beck, and Catherine M Chiba. Caenorhabditis elegans: a new model system for the study of learning and memory. *Behavioural brain research*, 37(1):89–92, 1990.

[208] Michael R Klass. A method for the isolation of longevity mutants in the nematode Caenorhabditis elegans and initial results. *Mechanisms of ageing and development*, 22(3-4):279–286, 1983.

[209] CH Opperman and S Chang. Effects of Aldicarb and Fenamiphos on Acetycholinesterase and Motility of Caenorhabditis elegans. *Journal of Nematology*, 23(1):20, 1991.

[210] James F Morley, Heather R Brignull, Jill J Weyers, and Richard I Morimoto. The threshold for polyglutamine-expansion protein aggregation and cellular toxicity is dynamic and influenced by aging in Caenorhabditis elegans. *Proceedings of the National Academy of Sciences*, 99(16): 10417–10422, 2002.

[211] Jesse M Gray, David S Karow, Hang Lu, Andy J Chang, Jennifer S Chang, Ronald E Ellis, Michael A Marletta, and Cornelia I Bargmann. Oxygen sensation and social feeding mediated by a C. elegans guanylate cyclase homologue. *Nature*, 430(6997):317–322, 2004.

[212] Linjiao Luo, Christopher V Gabel, Heon-Ick Ha, Yun Zhang, and Aravinthan DT Samuel. Olfactory behavior of swimming C. elegans analyzed by measuring motile responses to temporal variations of odorants. *Journal of neurophysiology*, 99(5):2617–2625, 2008.

[213] Terence I. Moy, Annie L. Conery, Jonah Larkins-Ford, Gang Wu, Ralph Mazitschek, Gabriele Casadei, Kim Lewis, Anne E. Carpenter, and Frederick M. Ausubel. High Throughput Screen for Novel Antimicrobials using a Whole Animal Infection Model. *ACS chemical biology*, 4(7):527–533, July 2009. ISSN 1554-8929. doi: 10.1021/cb900084v.

[214] Annelie Persson, Einav Gross, Patrick Laurent, Karl Emanuel Busch, Hugo Bretes, and Mario De Bono. Natural variation in a neural globin tunes oxygen sensing in wild Caenorhabditis elegans. *Nature*, 458(7241):1030–1033, 2009.

[215] Raphael Sznitman, Manaswi Gupta, Gregory D Hager, Paulo E Arratia, and Josue Sznitman. Multi-environment model estimation for motility analysis of Caenorhabditis elegans. *PLoS One*, 5(7):e11631, 2010.

[216] Jeong-Hoon Hahm, Sunhee Kim, Race DiLoreto, Cheng Shi, Seung-Jae V Lee, Coleen T Murphy, and Hong Gil Nam. C. elegans maximum velocity correlates with healthspan and is maintained in worms with an insulin receptor mutation. *Nature communications*, 6(1):1–7, 2015.

[217] Sheng Fong, Emelyne Teo, Li Fang Ng, Ce-Belle Chen, Lakshmi Narayanan Lakshmanan, Sau Yee Tsoi, Philip Keith Moore, Takao Inoue, Barry Halliwell, and Jan Gruber. Energy crisis precedes global metabolic failure in a novel Caenorhabditis elegans Alzheimer Disease model. *Scientific reports*, 6(1):1–9, 2016.

[218] Amy L Lee, Hailey M Ung, L Paul Sands, and Elise A Kikis. A new Caenorhabditis elegans model of human huntingtin 513 aggregation and toxicity in body wall muscles. *PloS one*, 12(3):e0173644, 2017.

[219] Michele Perni, Patrick Flagmeier, Ryan Limbocker, Roberta Cascella, Francesco A Aprile, Celine Galvagnion, Gabriella T Heller, Georg Meisl, Serene W Chen, Janet R Kumita, and others. Multistep inhibition of -synuclein aggregation and toxicity in vitro and in vivo by trodusquemine. *ACS chemical biology*, 13(8):2308–2319, 2018.

[220] D Dipon Ghosh, Dongyeop Lee, Xin Jin, H Robert Horvitz, and Michael N Nitabach. C. elegans discriminates colors to guide foraging. *Science*, 371(6533):1059–1063, 2021.

[221] Nicolas Roussel, Jeff Sprenger, Susan J Tappan, and Jack R Glaser. Robust tracking and quantification of C. elegans body shape and locomotion through coiling, entanglement, and omega bends. In *Worm*, volume 3, page e982437, 2014.

[222] Sijie Jason Wang and Zhao-Wen Wang. Track-a-worm, an open-source system for quantitative assessment of C. elegans locomotory and bending behavior. *PloS one*, 8(7):e69653, 2013.

[223] Zhaoyang Feng, Christopher J Cronin, John H Wittig, Paul W Sternberg, and William R Schafer. An imaging system for standardized quantitative analysis of C. elegans behavior. *BMC bioinformatics*, 5(1):1–6, 2004.

[224] Ebraheem Fontaine, Alan H Barr, and Joel W Burdick. Tracking of multiple worms and fish for biological studies. In *ICCV Workshop on Dynamical Vision*, 2007.

[225] Avelino Javer, Michael Currie, Chee Wai Lee, Jim Hokanson, Kezhi Li, Céline N. Martineau, Eviatar Yemini, Laura J. Grundy, Chris Li, QueeLim Chng, William R. Schafer, Ellen A. A. Nollen, Rex Kerr, and André E. X. Brown. An open-source platform for analyzing and sharing worm-behavior data. *Nature Methods*, 15(9):645–646, September 2018. ISSN 1548-7105. doi: 10.1038/s41592-018-0112-1.

[226] Daniel Ramot, Brandon E Johnson, Tommie L Berry Jr, Lucinda Carnell, and Miriam B Goodman. The Parallel Worm Tracker: a platform for measuring average speed and drug-induced paralysis in nematodes. *PloS one*, 3(5):e2208, 2008.

[227] Nicholas A Swierczek, Andrew C Giles, Catharine H Rankin, and Rex A Kerr. High-throughput behavioral analysis in C. elegans. *Nature methods*, 8(7):592–598, 2011.

[228] Shoubhik Chandan Banerjee, Khursheed Ahmad Khan, and Rati Sharma. Deep-Worm-Tracker: Deep Learning Methods for Accurate Detection and Tracking for Behavioral Studies in C. elegans, August 2022.

[229] Anthony D Fouad, Matthew A Churgin, Julia Hayden, Joyce Xu, Jeong-Inn Park, Alice Liu, Christopher Teng, Hongjing Sun, Mateo Parrado, Peter Bowlin, and others. High-throughput imaging of Caenorhabditis elegans aging using collective activity monitoring. *bioRxiv*, 2021.

[230] Laetitia Hebert, Tosif Ahamed, Antonio C Costa, Liam OShaughnessy, and Greg J Stephens. WormPose: Image synthesis and convolutional networks for pose estimation in C. elegans. *PLoS computational biology*, 17(4):e1008914, 2021.

[231] Eric E. Keaveny and André E. X. Brown. Predicting path from undulations for C. elegans using linear and nonlinear resistive force theory. *Physical Biology*, 14(2):025001, March 2017. ISSN 1478-3975. doi: 10.1088/1478-3975/aa5ce6.

[232] John C Crocker and David G Grier. Methods of digital video microscopy for colloidal studies. *Journal of colloid and interface science*, 179(1):298–310, 1996.

[233] Amirhossein Kazerouni, Ehsan Khodapanah Aghdam, Moein Heidari, Reza Azad, Mohsen Fayyaz, Ilker Hacihaliloglu, and Dorit Merhof. Diffusion Models for Medical Image Analysis: A Comprehensive Survey, November 2022.

[234] Alfonso Pérez-Escudero, Julián Vicente-Page, Robert C. Hinz, Sara Arganda, and Gonzalo G. de Polavieja. idTracker: tracking individuals in a group by automatic identification of unmarked animals. *Nature Methods*, 11(7):743–748, July 2014. ISSN 1548-7105. doi: 10.1038/nmeth.2994.

[235] Miguel A Matilla, Félix Velando, David Martín-Mora, Elizabet Monteagudo-Cascales, and Tino Krell. A catalogue of signal molecules that interact with sensor kinases, chemoreceptors and transcriptional regulators. *FEMS Microbiology Reviews*, 46(1):fuab043, January 2022. ISSN 0168-6445. doi: 10.1093/femsre/fuab043.

[236] Miguel A Matilla and Tino Krell. The effect of bacterial chemotaxis on host infection and pathogenicity. *FEMS Microbiology Reviews*, 42(1):fux052, January 2018. ISSN 0168-6445. doi: 10.1093/femsre/fux052.

[237] Stacey S. Willard and Peter N. Devreotes. Signaling pathways mediating chemotaxis in the social amoeba, Dictyostelium discoideum. *European Journal of Cell Biology*, 85(9):897–904, September 2006. ISSN 0171-9335. doi: 10.1016/j.ejcb.2006.06.003.

[238] Shah Jumaat Mohd Yussof, Effat Omar, Dinker R. Pai, and Suneet Sood. Cellular events and biomarkers of wound healing. *Indian Journal of Plastic Surgery*, 45(2):220–228, May 2012. ISSN 0970-0358, 1998-376X. doi: 10.4103/0970-0358.101282.

[239] Julie L. Christensen, Douglas E. Wright, Amy J. Wagers, and Irving L. Weissman. Circulation and Chemotaxis of Fetal Hematopoietic Stem Cells. *PLOS Biology*, 2(3):e75, March 2004. ISSN 1545-7885. doi: 10.1371/journal.pbio.0020075.

[240] Kent W. Hunter, Nigel PS Crawford, and Jude Alsarraj. Mechanisms of metastasis. *Breast Cancer Research*, 10(1):S2, December 2008. ISSN 1465-542X. doi: 10.1186/bcr1988.

[241] Victor Sourjik and Ned S Wingreen. Responding to chemical gradients: bacterial chemotaxis. *Current Opinion in Cell Biology*, 24(2):262–268, April 2012. ISSN 0955-0674. doi: 10.1016/j.ceb.2011.11.008.

[242] Richa Karmakar, Man-Ho Tang, Haicen Yue, Daniel Lombardo, Aravind Karanam, Brian A. Camley, Alex Groisman, and Wouter-Jan Rappel. Cellular memory in eukaryotic chemotaxis depends on the background chemoattractant concentration. *Physical Review E*, 103(1):012402, January 2021. doi: 10.1103/PhysRevE.103.012402.

[243] Akihiko Nakajima, Shuji Ishihara, Daisuke Imoto, and Satoshi Sawai. Rectified directional sensing in long-range cell migration. *Nature Communications*, 5(1):5367, November 2014. ISSN 2041-1723. doi: 10.1038/ncomms6367.

[244] Satpreet H. Singh, Floris van Breugel, Rajesh P. N. Rao, and Bingni W. Brunton. Emergent behaviour and neural dynamics in artificial agents tracking odour plumes. *Nature Machine Intelligence*, 5(1):58–70, January 2023. ISSN 2522-5839. doi: 10.1038/s42256-022-00599-w.

[245] Rahul O. Ramakrishnan and Benjamin M. Friedrich. Learning run-and-tumble chemotaxis with support vector machines. *Europhysics Letters*, 142(4):47001, May 2023. ISSN 0295-5075. doi: 10.1209/0295-5075/acd0d3.

[246] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, August 2017.

[247] Marcos F. Velho Rodrigues, Maciej Lisicki, and Eric Lauga. The bank of swimming organisms at the micron scale (BOSO-Micro). *PLOS ONE*, 16(6):e0252291, June 2021. ISSN 1932-6203. doi: 10.1371/journal.pone.0252291.

[248] Yevgeniy V. Kalinin, Lili Jiang, Yuhai Tu, and Mingming Wu. Logarithmic Sensing in Escherichia coli Bacterial Chemotaxis. *Biophysical Journal*, 96(6):2439–2448, March 2009. ISSN 0006-3495. doi: 10.1016/j.bpj.2008.10.027.

[249] Luis Alvarez, Benjamin M. Friedrich, Gerhard Gompper, and U. Benjamin Kaupp. The computational sperm cell. *Trends in Cell Biology*, 24(3):198–207, March 2014. ISSN 0962-8924, 1879-3088. doi: 10.1016/j.tcb.2013.10.004.

[250] Milena D. Lazova, Tanvir Ahmed, Domenico Bellomo, Roman Stocker, and Thomas S. Shimizu. Response rescaling in bacterial chemotaxis. *Proceedings of the National Academy of Sciences*, 108 (33):13870–13875, August 2011. doi: 10.1073/pnas.1108608108.

[251] Yuichi Iino and Kazushi Yoshida. Parallel Use of Two Behavioral Mechanisms for Chemotaxis in Caenorhabditis elegans. *Journal of Neuroscience*, 29(17), April 2009. ISSN 0270-6474, 1529-2401. doi: 10.1523/JNEUROSCI.3633-08.2009.

[252] Meisam Zaferani and Alireza Abbaspourrad. Biphasic Chemokinesis of Mammalian Sperm. *Physical Review Letters*, 130(24):248401, June 2023. doi: 10.1103/PhysRevLett.130.248401.

[253] Dani L. Bodor, Wolfram Pönisch, Robert G. Endres, and Ewa K. Paluch. Of Cell Shapes and Motion: The Physical Basis of Animal Cell Migration. *Developmental Cell*, 52(5):550–562, March 2020. ISSN 1534-5807. doi: 10.1016/j.devcel.2020.02.013.

[254] Marianne Grognot, Jong Woo Nam, Lauren E. Elson, and Katja M. Taute. Physiological adaptation in flagellar architecture improves Vibrio alginolyticus chemotaxis in complex environments. *Proceedings of the National Academy of Sciences*, 120(34):e2301873120, August 2023. doi: 10.1073/pnas.2301873120.

[255] Kirsty Y. Wan. Active oscillations in microscale navigation. *Animal Cognition*, 26(6):1837–1850, November 2023. ISSN 1435-9456. doi: 10.1007/s10071-023-01819-5.

[256] Alex Mogilner and Angelika Manhart. Intracellular Fluid Mechanics: Coupling Cytoplasmic Flow with Active Cytoskeletal Gel. *Annual Review of Fluid Mechanics*, 50(Volume 50, 2018):347–370, January 2018. ISSN 0066-4189, 1545-4479. doi: 10.1146/annurev-fluid-010816-060238.

[257] Diana Clausznitzer, Gabriele Micali, Silke Neumann, Victor Sourjik, and Robert G. Endres. Predicting Chemical Environments of Bacteria from Receptor Signaling. *PLOS Computational Biology*, 10(10):e1003870, October 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003870.

[258] Andrew J. Davidson, Clelia Amato, Peter A. Thomason, and Robert H. Insall. WASP family proteins and formins compete in pseudopod- and bleb-based migration. *Journal of Cell Biology*, 217(2):701–714, November 2017. ISSN 0021-9525. doi: 10.1083/jcb.201705160.

[259] Miri Adler and Uri Alon. Fold-change detection in biological systems. *Current Opinion in Systems Biology*, 8:81–89, April 2018. ISSN 2452-3100. doi: 10.1016/j.coisb.2017.12.005.

[260] Lea Goentoro and Marc W. Kirschner. Evidence that fold-change, and not absolute level, of beta-catenin dictates Wnt signaling. *Molecular Cell*, 36(5):872–884, December 2009. ISSN 1097-4164. doi: 10.1016/j.molcel.2009.11.017.

[261] Gerardo Aquino, Luke Tweedy, Doris Heinrich, and Robert G. Endres. Memory improves precision of cell sensing in fluctuating environments. *Scientific Reports*, 4(1):5688, July 2014. ISSN 2045-2322. doi: 10.1038/srep05688.

[262] Kento Nakamura and Tetsuya J. Kobayashi. Gradient sensing limit of a cell when controlling the elongating direction, May 2024.

[263] G A Nelson, T M Roberts, and S Ward. Caenorhabditis elegans spermatozoan locomotion: amoeboid movement with almost no actin. *Journal of Cell Biology*, 92(1):121–131, January 1982. ISSN 0021-9525. doi: 10.1083/jcb.92.1.121.

[264] Karen Alim. Fluid flows shaping organism morphology. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 373(1747):20170112, April 2018. doi: 10.1098/rstb.2017.0112.

[265] Louis González and Andrew Mugler. Collective effects in flow-driven cell migration. *Physical Review E*, 108(5):054406, November 2023. doi: 10.1103/PhysRevE.108.054406.

[266] Takuya Umedachi, Ryo Idei, Toshiyuki Nakagaki, Ryo Kobayashi, and Akio Ishiguro. Fluid-Filled Soft-Bodied Amoeboid Robot Inspired by Plasmodium of True Slime Mold. *Advanced Robotics*, 26(7):693–707, January 2012. ISSN 0169-1864. doi: 10.1163/156855312X626316.

[267] Chengyao Deng, Jiahao Dong, Yifei Guo, Xudong Sun, Zhongru Song, and Zhenkun Li. Amoeboid soft robot based on multi-material composite 3D printing technology. *Journal of Magnetism and Magnetic Materials*, 588:171390, December 2023. ISSN 0304-8853. doi: 10.1016/j.jmmm.2023.171390.

[268] Bo Hu, Wen Chen, Wouter-Jan Rappel, and Herbert Levine. How geometry and internal bias affect the accuracy of eukaryotic gradient sensing. *Physical Review E*, 83(2):021917, February 2011. doi: 10.1103/PhysRevE.83.021917.

[269] Austin Hopkins and Brian A. Camley. Chemotaxis in uncertain environments: Hedging bets with multiple receptor types. *Physical Review Research*, 2(4):043146, October 2020. doi: 10.1103/PhysRevResearch.2.043146.

[270] Travis I. Moore, Hiromasa Tanaka, Hyung Joon Kim, Noo Li Jeon, and Tau-Mu Yi. Yeast G-proteins mediate directional sensing and polarization behaviors in response to changes in pheromone gradient direction. *Molecular Biology of the Cell*, 24(4):521–534, February 2013. ISSN 1059-1524. doi: 10.1091/mbc.E12-10-0739.

[271] Robert G. Endres. Polar Chemoreceptor Clustering by Coupled Trimers of Dimers. *Biophysical Journal*, 96(2):453–463, January 2009. ISSN 0006-3495. doi: 10.1016/j.bpj.2008.10.021.

[272] Run-Zhi Lai, Josiah M. B. Manson, Arjan F. Bormans, Roger R. Draheim, Ngoc T. Nguyen, and Michael D. Manson. Cooperative Signaling among Bacterial Chemoreceptors. *Biochemistry*, 44 (43):14298–14307, November 2005. ISSN 0006-2960, 1520-4995. doi: 10.1021/bi050567y.

[273] Rosemary S. McAndrew, E. Ann Ellis, Michael Manson, and Andreas Holzenburg. TEM Anaysis of Chemoreceptor Arrays in Native Membranes of *E. coli*. *Microscopy and Microanalysis*, 10(S02): 416–417, August 2004. ISSN 1431-9276, 1435-8115. doi: 10.1017/S1431927604886239.

[274] Peter Ames, Claudia A. Studdert, Rebecca H. Reiser, and John S. Parkinson. Collaborative signaling by mixed chemoreceptor teams in Escherichia coli. *Proceedings of the National Academy of Sciences*, 99(10):7060–7065, May 2002. doi: 10.1073/pnas.092071899.

[275] Sebastian Thiem and Victor Sourjik. Stochastic assembly of chemoreceptor clusters in Escherichia coli. *Molecular Microbiology*, 68(5):1228–1236, 2008. ISSN 1365-2958. doi: 10.1111/j.1365-2958.2008.06227.x.

[276] Christopher W. Jones and Judith P. Armitage. Positioning of bacterial chemoreceptors. *Trends in Microbiology*, 23(5):247–256, May 2015. ISSN 0966-842X, 1878-4380. doi: 10.1016/j.tim.2015.03.004.

[277] En Cai, Casey Beppler, John Eichorst, Kyle Marchuk, Scott W. Eastman, and Matthew F. Krummel. T cells use distinct topographical and membrane receptor scanning strategies that individually coalesce during receptor recognition. *Proceedings of the National Academy of Sciences of the United States of America*, 119(32):e2203247119, August 2022. ISSN 0027-8424. doi: 10.1073/pnas.2203247119.

[278] Heidi Koldsø, David Shorthouse, Jean Hélie, and Mark S. P. Sansom. Lipid Clustering Correlates with Membrane Curvature as Revealed by Molecular Simulations of Complex Lipid Bilayers. *PLOS Computational Biology*, 10(10):e1003911, October 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003911.

[279] Shilong Yang, Xinwen Miao, Steven Arnold, Boxuan Li, Alan T. Ly, Huan Wang, Matthew Wang, Xiangfu Guo, Medha M. Pathak, Wenting Zhao, Charles D. Cox, and Zheng Shi. Membrane curvature governs the distribution of Piezo1 in live cells. *Nature Communications*, 13(1):7467, December 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-35034-6.

[280] Shao-Zhen Lin, Jacques Prost, and Jean-François Rupprecht. Curvature-induced clustering of cell adhesion proteins. *Physical Review E*, 109(5):054406, May 2024. doi: 10.1103/PhysRevE.109.054406.

[281] Hui Wang, Ned S. Wingreen, and Ranjan Mukhopadhyay. Self-Organized Periodicity of Protein Clusters in Growing Bacteria. *Physical Review Letters*, 101(21):218101, November 2008. doi: 10.1103/PhysRevLett.101.218101.

[282] Will Draper and Jan Liphardt. Origins of chemoreceptor curvature sorting in Escherichia coli. *Nature Communications*, 8(1):14838, March 2017. ISSN 2041-1723. doi: 10.1038/ncomms14838.

[283] Shuangyu Bi, Abiola M. Pollard, Yiling Yang, Fan Jin, and Victor Sourjik. Engineering Hybrid Chemotaxis Receptors in Bacteria. *ACS Synthetic Biology*, 5(9):989–1001, September 2016. ISSN 2161-5063, 2161-5063. doi: 10.1021/acssynbio.6b00053.

[284] Cristina Chircov, Alexandra Ctlina Bîrc, Alexandru Mihai Grumezescu, and Ecaterina Andronescu. Biosensors-on-Chip: An Up-to-Date Review. *Molecules*, 25(24):6013, December 2020. ISSN 1420-3049. doi: 10.3390/molecules25246013.

[285] Henry Cavanagh, Daryan Kempe, Jessica K. Mazalo, Maté Biro, and Robert G. Endres. T cell morphodynamics reveal periodic shape oscillations in three-dimensional migration. *Journal of the Royal Society, Interface*, 19(190):20220081, May 2022. ISSN 1742-5662. doi: 10.1098/rsif.2022.0081.

[286] Doug P. Hardin, Timothy Michaels, and Edward B. Saff. A Comparison of Popular Point Configurations on S^2. *Dolomites Research Notes on Approximation*, 9(DRNA Volume 9.1):16–49, November 2016. ISSN 2035-6803.

[287] Dimitrios Stamou, Kadla Rosholm, Alexander Damalas, Nikos Hatzakis, Volker Wirth, Karen Martinez, Natascha Leijnse, Lene Oddershede, Poul Martin Bendix, and Soren Pedersen. Membrane Curvature Regulates the Localization of G Protein Coupled Receptors and Ras Isoforms. *Biophysical Journal*, 108(2):95a–96a, January 2015. ISSN 0006-3495. doi: 10.1016/j.bpj.2014.11.551.

[288] Kadla R. Rosholm, Natascha Leijnse, Anna Mantsiou, Vadym Tkach, Søren L. Pedersen, Volker F. Wirth, Lene B. Oddershede, Knud J. Jensen, Karen L. Martinez, Nikos S. Hatzakis, Poul Martin Bendix, Andrew Callan-Jones, and Dimitrios Stamou. Membrane curvature regulates ligand-specific membrane sorting of GPCRs in living cells. *Nature Chemical Biology*, 13(7):724–729, July 2017. ISSN 1552-4469. doi: 10.1038/nchembio.2372.

[289] Daiqiu Mou and Yuansheng Cao. Optimal cell shape for accurate chemical gradient sensing in eukaryote chemotaxis. *Physical Review Letters*, 2024.

[290] Akinori Baba, Tetsuya Hiraiwa, and Tatsuo Shibata. Directional sensing of deformed cells under faint gradients. *Physical Review E*, 86(6):060901, December 2012. doi: 10.1103/PhysRevE.86.060901.

[291] Thomas Duke and Ian Graham. Equilibrium mechanisms of receptor clustering. *Progress in Biophysics and Molecular Biology*, 100(1):18–24, September 2009. ISSN 0079-6107. doi: 10.1016/j.pbiomolbio.2009.08.003.

[292] W. Helfrich. Elastic Properties of Lipid Bilayers: Theory and Possible Experiments. *Zeitschrift für Naturforschung C*, 28(11-12):693–703, December 1973. ISSN 1865-7125. doi: 10.1515/znc-1973-11-1209.

[293] Chiara Bernard, Angelo Rosario Carotenuto, Nicola Maria Pugno, Luca Deseri, and Massimiliano Fraldi. The interplay between membrane viscosity and ligand-binding receptor kinetics in lipid bilayers. *Meccanica*, 59(8):1231–1253, August 2024. ISSN 1572-9648. doi: 10.1007/s11012-024-01779-1.

[294] Andrew A. Potanin, V. Vladyslav, Olga S. Belokoneva, and Frederick W. Wiegel. Kinetics of ligand binding to a cluster of membrane-associated receptors. *European Biophysics Journal*, 23 (3):197–205, May 1994. ISSN 1432-1017. doi: 10.1007/BF01007611.

[295] Ming-Jie Wang, Yulia Artemenko, Wen-Jie Cai, Pablo A. Iglesias, and Peter N. Devreotes. The Directional Response of Chemotactic Cells Depends on a Balance between Cytoskeletal Architecture and the External Gradient. *Cell Reports*, 9(3):1110–1121, November 2014. ISSN 2211-1247. doi: 10.1016/j.celrep.2014.09.047.

[296] Ricardo Cortez. The Method of Regularized Stokeslets. *SIAM Journal on Scientific Computing*, 23(4):1204–1225, January 2001. ISSN 1064-8275, 1095-7197. doi: 10.1137/s106482750038146x.

[297] D. J. Smith. A boundary element regularized Stokeslet method applied to cilia- and flagella-driven flow. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 465 (2112):3605–3626, September 2009. doi: 10.1098/rspa.2009.0295.

[298] H. Nguyen, P. Dayan, and G. J. Goodhill. How receptor diffusion influences gradient sensing. *Journal of The Royal Society Interface*, 12(102):20141097, January 2015. doi: 10.1098/rsif.2014.1097.

[299] Ron Goldman. Curvature formulas for implicit curves and surfaces. *Computer Aided Geometric Design*, 22(7):632–658, October 2005. ISSN 01678396. doi: 10.1016/j.cagd.2005.06.005.

[300] Eleni Katifori and Marcelo O Magnasco. Quantifying loopy network architectures. *PloS one*, 7 (6):e37994, 2012.

[301] Adam Konkol, Jon Schwenk, Eleni Katifori, and John Burnham Shaw. Interplay of river and tidal forcings promotes loops in coastal channel networks. *Geophysical Research Letters*, 49(10): e2022GL098284, 2022.

[302] Alessandro Lonardi and Caterina De Bacco. Bilevel optimization for traffic mitigation in optimal transport networks. *Physical Review Letters*, 131(26):267401, 2023.

[303] Logan Numerow, Yue Li, Stelian Coros, and Bernhard Thomaszewski. Differentiable Voronoi Diagrams for Simulation of Cell-Based Mechanical Systems, April 2024.

[304] Fan Feng, Shiying Xiong, Ziyue Liu, Zangyueyang Xian, Yuqing Zhou, Hiroki Kobayashi, Atsushi Kawamoto, Tsuyoshi Nomura, and Bo Zhu. Cellular topology optimization on differentiable Voronoi diagrams. *International Journal for Numerical Methods in Engineering*, 124(1):282–304, 2023. ISSN 1097-0207. doi: 10.1002/nme.7121.

[305] Johan Gielis. A generic geometric transformation that unifies a wide range of natural and abstract shapes. *American Journal of Botany*, 90(3):333–338, 2003. ISSN 1537-2197. doi: 10.3732/ajb.90.3.333.

[306] Peijian Shi, David A. Ratkowsky, Yang Li, Lifang Zhang, Shuyan Lin, and Johan Gielis. A General Leaf Area Geometric Formula Exists for PlantsEvidence from the Simplified Gielis Equation. *Forests*, 9(11):714, November 2018. ISSN 1999-4907. doi: 10.3390/f9110714.

[307] Radost Waszkiewicz, John Burnham Shaw, Maciej Lisicki, and Piotr Szymczak. Goldilocks fluctuations: dynamic constraints on loop formation in scale-free transport networks. *Physical Review Letters*, 132(13):137401, 2024.

[308] Dimitri Bertsekas. *Nonlinear Programming*. Athena Scientific, September 2016. ISBN 978-1-886529-05-2.